STA380 Practice Problems for Quiz 3

These problems are not to be handed in, but they are for extra practice for students to be prepared for the quiz.

1. Consider the following probability density function:

$$f_Y(y) = \begin{cases} \frac{1}{8} & 0 < y \leq 2, \\ \frac{y}{8} & 2 < y \leq 4, \\ 0 & \text{otherwise.} \end{cases}$$

   Use the simple Monte Carlo estimator to **estimate** the expected value. Write the algorithm and the corresponding `R` codes. Use $n = 10^4$. Compare it to the true value.

2. Consider the Weibull distribution:

$$f_X(x) = \begin{cases} \frac{\alpha}{\lambda}\left(\frac{x}{\lambda}\right)^{\alpha-1} e^{-(x/\lambda)^\alpha} & x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

   Where α represents the shape parameter and λ represents the scale parameter. Let $X \sim Weibull(\alpha = 2, \lambda = 3)$. Use the **hit or miss** approach to compute $\mathbb{P}(0 < X < 3)$.

   (a) Write out the algorithm.

   (b) Using $n = 10^4$, code the algorithm in `R` using `rweibull()`. Compare this value with `pweibull()`. *Hint: please use and read* `help(rweibull)` *and* `help(pweibull)` *to understand how this function works.)*

   (c) Using $n = 10^4$, code the algorithm in `R` **without** using `rweibull()`. Compare this value with `pweibull()`. *Hint: in the previous set of practice problems, we generated values from the weibull distrbution using the acceptance-rejection method.*

Below are the solutions.

1. Note that,

$$\mathbb{E}[Y] = \int_0^2 \frac{1}{8}dy + \int_2^4 \frac{y}{8}dy = \frac{1}{4} + \int_2^4 \frac{y}{8}dy$$

   We can evaluate the first integral for the estimator since it's entirely free of y. Now:

$$\int_2^4 \frac{y}{8}dy = (4-2)\int_2^4 \frac{y}{(4-2)8}dy = 2\mathbb{E}_U[Y/8]$$

   Where $U \sim Uniform(2,4)$. Using the SLLN, our estimator will be:

$$\widehat{\mathbb{E}[Y]} = \frac{1}{4} + 2\frac{1}{n}\sum_{i=1}^{n}\frac{u_i}{8}$$

   The algorithm is as follows:

   (a) Generate $u_1, \ldots, u_m \sim Uniform(2,4)$.
   (b) Compute $\widehat{\mathbb{E}[g(Y)]} = \frac{1}{m}\sum_{i=1}^{m}\frac{u_i}{8}$.
   (c) Deliver $\hat{\theta} = \frac{1}{4} + 2\widehat{\mathbb{E}[g(Y)]}$.

   The code for estimation is:

```
n <- 10^4
u <- runif(n, 2, 4)
est <- 1/4 + 2*mean(u/8)
```

   The rest is optional exploration; it shows how to integrate in R instead of by hand (although on a test, if I **do not** ask you to code in R, please don't write this instead of actually evaluating an integral.)

```
help(integrate)

fy <- function(x) {
  ifelse(x <= 0, 0,
         ifelse(x < 2, 1/8,
                ifelse(x < 4, x/8, 0)))
}

true_val <- integrate(fy, lower = 0, upper = 4)
# below gives an error
abs(est - true_val)
# use below if you ever want to diagnose how to extract the numerical value for comparison
str(true_val)
# we see the value of the integral is hidden within $value

abs(est - true_val$value)

# this is if you want to do automated testing -- for sessions where
# you are working on multiple functions,
# you always want to be testing if all values are reasonable.
library(testthat)
test_that("Ensuring mc estimation of mean is sensible", {
  expect_equal(est, true_val$value, tol = 1e-3)
})
```

2. (a)
   - Generate a random sample $X_1, \ldots, X_n$ from the $Weibull(\alpha = 2, \lambda = 3)$.
   - For each observation X_i, compute:

$$I(0 \leq X_i \leq 3) = \begin{cases} 1 & 0 < X_i \leq 3, \\ 0 & \text{otherwise.} \end{cases}$$

- Compute $\widehat{\mathbb{P}}(0 \leq X_i \leq 3) = \frac{1}{n}\sum_{i=1}^{n} I(0 \leq X_i \leq 3)$.

(b)

```
n <- 10^4
x <- rweibull(n, shape = 2, scale = 3)
prob_est <- sum(x < 3)/n
prob_est

prob_exa <- pweibull(3, shape = 2, scale = 3)

test_that("Ensuring mc estimation of mean is sensible", {
  expect_equal(prob_est, prob_exa, tol = 1e-2)
})
# if you increase n, you can also decrease the above tolerance.
```

(c) In this example, without using Weibull we use the acceptance-rejection method:

```
n <- 10^4
accepted <- numeric(n)
u_accepted <- numeric(n)
i <- 0
iteration <- 0
while(i < n){
  y <- rexp(n = 1, rate = 1/3) # candidate from g
  u <- runif(1) # u ~ uniform(0, 1)
  ftgt <- (1/3) * y * exp(-(y/3)^2 + y/3) # f(x)/cg(x)

  if(u < ftgt){
    i <- i+1
    accepted[i] <- y
    u_accepted[i] <- u
  }
  iteration <- iteration + 1
}
# below are the samples we are going to use
accepted
prob_est2 <- sum(accepted < 3)/n
prob_est2

test_that("Ensuring mc estimation of mean is sensible", {
  expect_equal(prob_est2, prob_exa, tol = 1e-2)
})
```