STA4273 HW 2

Alex Jones

2023-02-13

6.1

Question 6.1 asks to compute a Monte Carlo estimate of $\int_0^{\pi/3} sin(t)dt$

```
m <- 10000
x <- runif(m, 0, pi/3)
theta_hat <- (pi/3)*(mean(sin(x)))
theta <- -cos(pi/3) - (-1)
```

With $\theta = 0.5$ and $\hat{\theta} = 0.495228$, our estimate is 0.95% off from the true value.

6.3

Question 6.3 asks to compute Monte Carlo estimate of $\int_0^{0.5} e^{-x}dx$ by first using the Uniform distribution (where the estimate is denoted $\hat{\theta}$) and then using the exponential distribution ($\hat{\theta^\star}$). Afterwards, we compare the variances to see which is a better estimator.

```
theta_hat_sampler <- replicate(1000, expr = {
  x <- runif(m, 0, 0.5)
  theta_hat <- 0.5*mean(exp(-x))
})

theta_star_sampler <- replicate(1000, expr = {
  g <- rexp(m)
  theta_star <- mean(g <= 0.5)
})
var(theta_hat_sampler)
```

```
## [1] 3.346701e-07
```

```
var(theta_star_sampler)
```

```
## [1] 2.349377e-05
```

It appears that $\hat{\theta}$ has a smaller variance than $\hat{\theta^\star}$. Verifying this mathematically:

$Var(\hat{\theta}) = \frac{1}{4m}(\int_0^{0.5} 2e^{-2u}du - (\int_0^{0.5} 2e^{-u}du)^2) \approx \frac{0.013}{4m}$
$Var(\hat{\theta^\star}) = \frac{(1-e^{-0.5})(e^{-0.5})}{m} \approx \frac{0.2386}{m}$
$\frac{Var(\hat{\theta})}{Var(\hat{\theta^\star})} = \frac{0.013}{4*0.2386} < 1$
The ratio is less than one, thus $\hat{\theta}$ is a more efficient estimator.

6.4

Question 6.4 asks to write a function to compute a Monte Carlo estimate of the Beta(3, 3) cdf, and use the function to estimate F(x) for x = 0.1, 0.2,. . . , 0.9.

```
estimates <- c()
true_prob <- c()
values <- seq(0.1,0.9, 0.1)
cdf_computer <- function(n, p) {
  x <- runif(n, min = 0, max = p)
  ### the function inside mean() is a simplified form of the beta(3,3) pdf
  p*mean((x^2*(1-x)^2*30))
}

for (i in values) {
  estimates <- append(estimates, cdf_computer(100000, i))
  true_prob <- append(true_prob, pbeta(i, 3,3))
}
percent_error <- 100*(estimates - true_prob)/true_prob
round(rbind(values, true_prob, estimates, percent_error), 4)
```

```
##                 [,1]   [,2]   [,3]   [,4]   [,5]    [,6]    [,7]    [,8]
## values        0.1000 0.2000 0.3000 0.4000 0.5000  0.6000  0.7000  0.8000
## true_prob     0.0086 0.0579 0.1631 0.3174 0.5000  0.6826  0.8369  0.9421
## estimates     0.0086 0.0580 0.1638 0.3177 0.5002  0.6825  0.8352  0.9418
## percent_error 0.0507 0.0810 0.4395 0.0706 0.0393 -0.0058 -0.2035 -0.0338
##                  [,9]
## values         0.9000
## true_prob      0.9914
## estimates      0.9905
## percent_error -0.0981
```

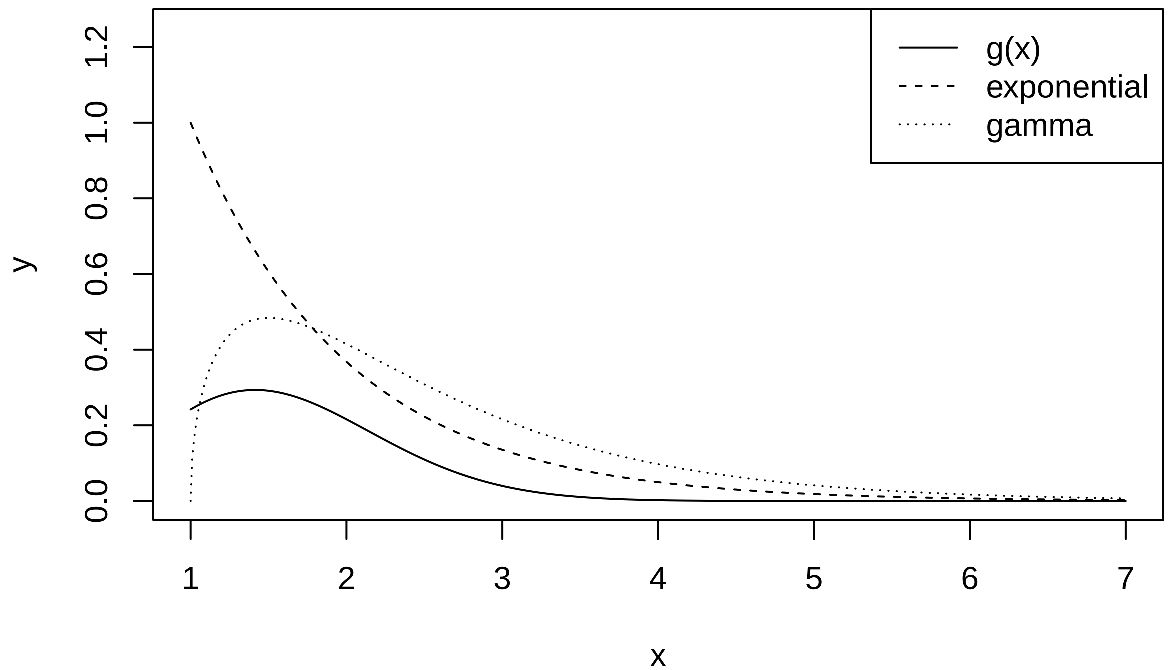
Our estimates seem reasonably close to the true probability.

6.13

6.13 asks to find two functions to estimate $\int_1^\infty \frac{x^2}{\sqrt{2\pi}} e^{-x^2} dx$, then to compare the functions to find the better estimator.

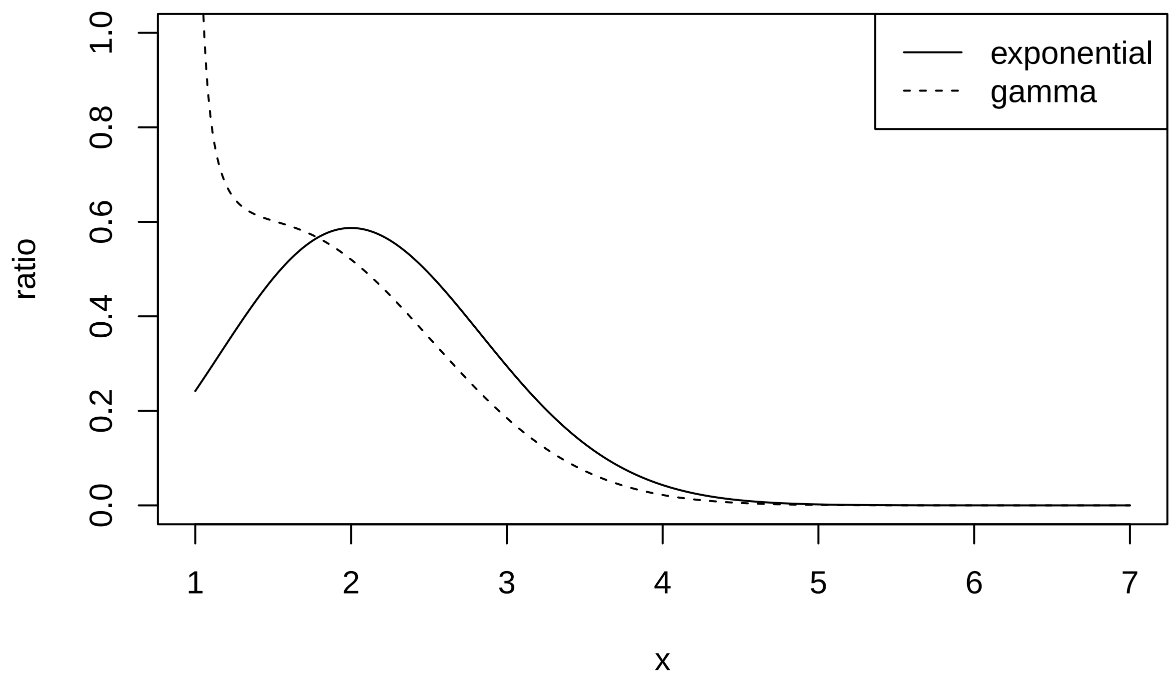
The two functions we will look at are the exponential distribution, and the gamma distribution (both shifted by 1 in order to match the support). Comparing those to the given function:

```
x <- seq(1,7,0.01)
y <- x^2 * exp(-x^2/2)/sqrt(2*pi)
yexp <- dexp(x-1)
ygam <- dgamma(x-1, 3/2, 1)
plot(x,y, type = "l", ylim = c(0,1.25))
lines(x, yexp, lty = 2)
lines(x, ygam, lty = 3)
legend("topright", legend = c("g(x)","exponential","gamma"), lty = 1:3)
```



Then creating functions to compare the ratios to minimize errors:

```
expratio <- y/yexp
gamratio <- y/ygam
plot(x, expratio, type = "l", ylim = c(0,1), ylab = "ratio")
lines(x, gamratio, lty = 2)
legend("topright", legend = c("exponential", "gamma"), lty = c(1,2))
```



It seems the modified exponential function would be the better fit, since the importance function seems more "constant".

6.14

6.14 asks to find the estimate of the integral referenced in 6.13
Using the modified exponential distribution to sample:

```
x <- rexp(10000) + 1
est <- mean((x^2 * exp(-x^2/2))/sqrt(2*pi)/dexp(x-1))
```

Our estimate for the integral is 0.4024391