Statistical Computing HW2

Wei-Chen Chang r12227118

Due: 2024-04-03

Q1.

Find two importance function f_1 and f_2 that are supported on $(1, \infty)$ and are "close" to

$$g(x) = \frac{x^2}{\sqrt{2\pi}} e^{-x^2/2},\ x > 1$$

Which of your two importance functions should produce the smaller variance in estimating

$$\int_1^\infty \frac{x^2}{\sqrt{2\pi}} e^{-x^2/2} dx$$

by importance sampling? Explain.

Ans:

Here I choose truncated exponential distribution for $f_1(x)$, a truncated $N(1,1)$ on $[1, \infty)$ for $f_2(x)$. Their pdf were shown as below:

$$\begin{aligned}
f_1(x) &= \lambda x^{-\lambda x} / \int_1^\infty \lambda x^{-\lambda x} dx \\
&= \lambda x^{-\lambda(x-1)},\ (\lambda = 2); \\
f_2(x) &= \frac{1}{\sqrt{2\pi}} \exp(\frac{-(x-1)^2}{2}) / \int_1^\infty \frac{1}{\sqrt{2\pi}} \exp(\frac{-(x-1)^2}{2}) dx \\
&= \frac{2}{\sqrt{2\pi}} \exp(\frac{-(x-1)^2}{2}).
\end{aligned}$$

As a remark, note that $\frac{1}{\sqrt{2\pi}} \exp(\frac{-(x-1)^2}{2})$ is symmetric at 1, thus its integral on $[1, \infty)$ equals to $\frac{1}{2}$. The shape of f_1, f_2 compared to $g(x)$ can be seen in Fig.1.

```r
g <- function(x){
  return(x^2/sqrt(2*pi)*exp(-(x^2/2)))
}
f1 <- function(x,lambda=1){ #truncated exponential
  return(ifelse(x>=1, lambda*exp(-lambda*(x-1)),0))
}
f1_invcdf <- function(x,lambda=1){ #truncated exponential
  return(1-(log(1-x)/lambda))
}
```

```
# f2 truncated normal
f2 <- function(x){
  return(ifelse(x>=1 ,sqrt(2/pi)*exp(-(x-1)^2/2),0))
}

curve(g(x),from=1,to=8, lwd=2, col ="darkcyan", ylim = c(0,2),
      main = "target and importance functions", ylab = "density")
curve(f1(x, lambda=2),from=1,to=8, lwd=2, lty=2, col ="red", add= TRUE)
curve(f2(x),from=1,to=8, lwd=2, lty=3, col ="darkorange", add= TRUE)
legend("topright",c("g(x)", "f1(x)", "f2(x)"),
       col= c("darkcyan", "red", "darkorange"),
       lty = 1:3, lwd=c(2,2,2))
```

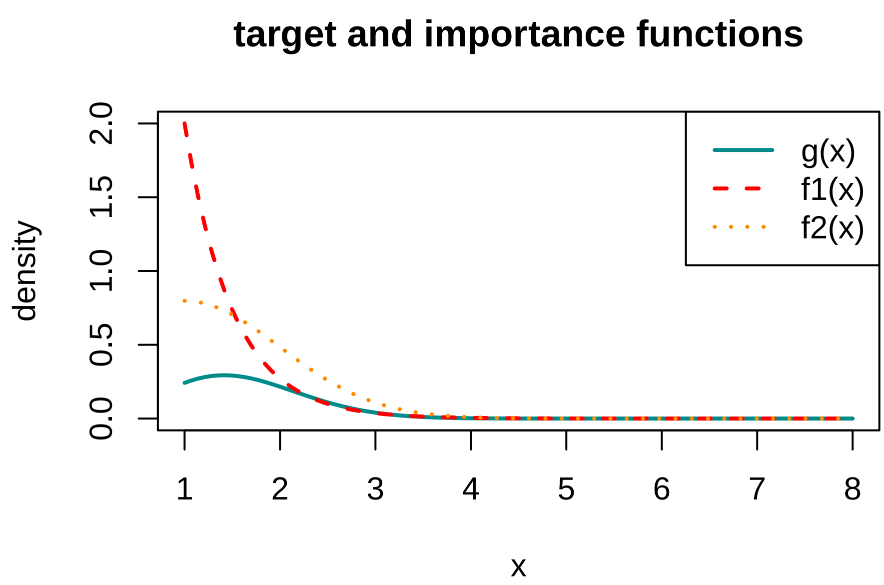


Figure 1: Target and Importance Functions

Following the importance sampling, we have the relation below.

$$\theta = \int_1^\infty g(x)dx = \int_1^\infty \frac{g(x)}{f_i(x)} f_i(x)dx = E_{f_i}[\frac{g(x)}{f_i(x)}],\ (i = 1, 2)$$

And the MC estimate would be:

$$\hat{\theta}_j = \frac{1}{n}\Sigma_{i=1}^n \frac{g(X_i)}{f_j(X_i)},\ X_i \sim f_j,\ (j = 1, 2)$$

For $f_1(x)$, we derive its quantile function $F_1^{-1}(x) = 1 - \frac{\log(1-x)}{\lambda}$ to apply the inverse transform method to generate random samples.
For $f_2(x)$, note that $f_2(x) = |Z| + 1$, where $Z \sim N(0, 1)$. Thus generate random samples from standard normal distribution then apply the transformation would done the sampling.

```r
set.seed(322)
N=1000
u <- runif(N)
f1_rsamp <- f1_invcdf(u, lambda=2)
theta1 <- (g(f1_rsamp)/f1(f1_rsamp, lambda = 2)) %>% mean()
theta1.se <- sd(g(f1_rsamp)/f1(f1_rsamp, lambda = 2))/N
theta1.var <-  (theta1.se*N)^2

f2_rsamp <- abs(rnorm(N))+1
theta2 <- (g(f2_rsamp)/f2(f2_rsamp)) %>% mean()
theta2.se <- sd(g(f2_rsamp)/f2(f2_rsamp))/N
theta2.var <-  (theta2.se*N)^2
matrix(c(theta1, theta1.se,theta1.var, theta2, theta2.se,theta2.var),
       ncol=3, byrow = T,
       dimnames = list(c("theta_1", "theta_2"),
                       c("mean", "s.e.", "var")))%>% round(4)
```

```
##            mean  s.e.    var
## theta_1 0.4024 3e-04 0.0831
## theta_2 0.3998 0e+00 0.0020
```

Choosing $f_2(x)$ as the important function performs better than f_1. One can see from Fig.1, when X_i is close to 1, f_1 has much larger weight $(g(x)/f_1(x))$ than choosing f_2, causeing greater variance.

Q2

We want to compute the following integral by Monte Carlo:

$$\int_0^1 e^{-x\cos(\pi x)}dx = E[h(U)],$$

where U is a uniform distribution on $[0, 1]$ and $h(x) = e^{-x\cos(\pi x)}$. Suppose we use the control variate $Y = g(U)$ with $g(x) = e^{-x}$. Note that we can compute the mean of Y explicitly. By comparing a Monte Carlo estimator with and without control variate, please find the variance reduction from the use of control variate.

Ans

First, the mean of Y, $E(Y)$ is:

$$E(e^{-U})) = \int_0^1 e^{-t} \cdot 1dt = 1 - e^{-1}.$$

And the control variate estimator can be expressed as:

$$\hat{\theta}_{\text{cv}} = h(U) + c(g(U) - (1 - e^{-1}))$$

where c can be estimated by -1 times the slope coefficient (i.e., $-\beta_1$) of regressing $h(U)$ on $g(U)$.

```r
h <- function(x) return(exp(-x*cos(pi*x)))
g <- function(x) return(exp(-x))
#Return estimate, s.e., variance of the MC estimator
MC_estimates = function(MC_sample){
  m = mean(MC_sample)
  se = sd(MC_sample)/length(MC_sample)
  var = var(MC_sample)
  return(c("est"=m, "s.e."=se, "var"=var))
}

set.seed(987)
N=1000
u <- runif(N)
y <- g(u)
# simple MC integral
sMC_samp <-  h(u)
simple_MC <- MC_estimates(sMC_samp)
# control variate
c <- -lm(h(u)~y)$coef[2]
CV_samp <- sMC_samp + c*(y - (1-exp(-1)))
control_variate <- MC_estimates(CV_samp)

rbind(simple_MC,control_variate) %>% round(4)
cat(paste0("Reduced variance: ",
          round((simple_MC["var"]-control_variate["var"])/simple_MC["var"], 4)*100,"%"))
```

```
##                    est  s.e.    var
## simple_MC       1.3275 6e-04 0.3362
## control_variate 1.3369 3e-04 0.1143
## Reduced variance: 66%
```

Q3

Estimate the integral $\theta = \int_0^1 e^{x^2} dx$ using Monte Carlo method with $n = 10000$ for the following estimators:

(a) Regular Monte Carlo estimator.

(b) Antithetic variable estimator $\frac{1}{2}e^{U^2} + \frac{1}{2}e^{(1-U)^2}$

(c) Control variate estimator using U as a control variate.

(d) Combining the antithetic variable and control variate methods:

$$\hat{\theta}_{\alpha,c} = \alpha e^{U^2} + (1-\alpha)e^{(1-U)^2} + c(U - \frac{1}{2}).$$

Try to compute the optimal pair (α^*, c^*) which achieves the smallest variance. Discuss the efficiency of four estimators.

Ans:

for (d), first notice that U is just a control variate for the antithetic variable estimator $\alpha e^{U^2} + (1-\alpha)e^{(1-U)^2}$.
To attain optimal c, one only needs to regress the random samples from antithetic estimator on U.
And to minimize the variance of antithetic variable estimator, $\alpha = 1/2$ is the optimal choice.

```
set.seed(998)
N=1000
target <- function(x) return(exp(x^2))
u <- runif(N)
# simple MC
samp_sMC <- target(u)
Simple_MC <- MC_estimates(samp_sMC)
# antithetic
samp_anti <- target(u)/2 + target(1-u)/2
Antithetic <- MC_estimates(samp_anti)
# control variate
c <- -lm(samp_sMC~u)$coef[2]
samp_cv <- samp_sMC + c*(u-.5) # E(U) = 1/2
Control_Variate <- MC_estimates(samp_cv)
# antithetic variable and control variate
c2 <- -lm(samp_anti~u)$coef[2]
samp_anti_cv <- samp_anti + c2*(u-.5)
Anti_CtrlVar <- MC_estimates(samp_anti_cv)

rbind(Simple_MC, Antithetic, Control_Variate,Anti_CtrlVar) %>% round(6)
```

```
##                      est     s.e.      var
## Simple_MC       1.443744 0.000458 0.209684
## Antithetic      1.454669 0.000168 0.028139
## Control_Variate 1.453543 0.000171 0.029252
## Anti_CtrlVar    1.454485 0.000168 0.028075
```

Discussion

The simple MC estimator yields the greatest variance. Other estimators have smaller and variance with similar values. The antithetic and control variate combined estimator has the lowest variance, followed by antithetic variate estimator, lastly the control variate estimator at last.
As antithetic variate can be view as a special case of control variate with perfect negative correlation, it's no wonder to see it more efficient than control variate estimator.
An additional control variate can reduce the variance of estimate but not much. We can take a look at the 95% CI of $\hat{c}$: (-0.0652,0.0085), which covers 0. It indicates U is not correlated to antithetic variate estimator, and adding such cannot reduce more variance.

Q4

Use importance sampling to estimate the quantity:

$$\theta = \int_0^\infty x \frac{e^{-(0.5-x)^2/2} e^{-3x}}{C} dx$$

where $C = \int_0^\infty e^{-(0.5-x)^2/2}e^{-3x}dx$ is a normalizing constant of a PDF. Plot the converge of the estimator versus sample size.
Note: You may consider $3e^{-3x}$ as the importance function. Hint: use self normalized importance sampling.

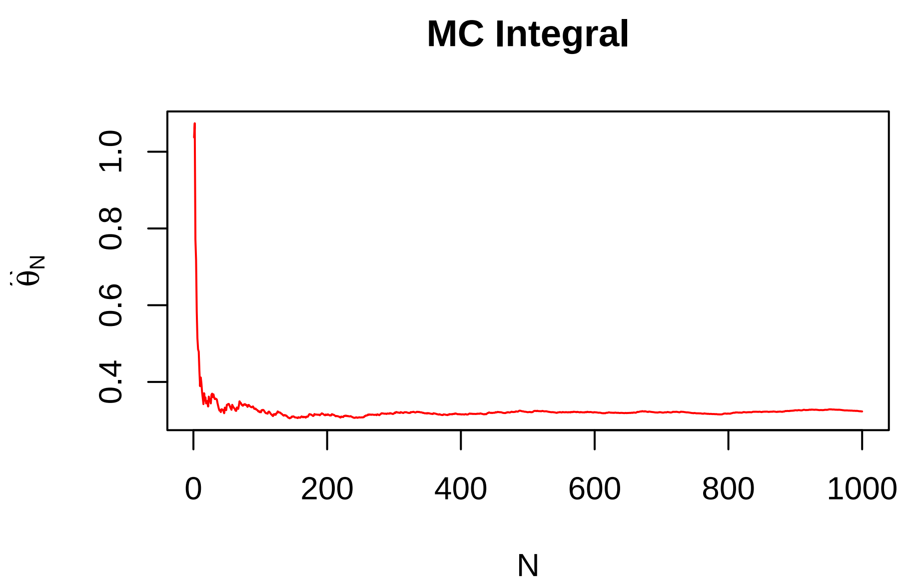
Ans:

Let $f(x) = e^{-(0.5-x)^2/2}e^{-3x}$, $g(x) = 3e^{-3x}$.

$$\begin{aligned}
\theta &= \int_0^\infty x\frac{f(x)}{C}dx \\
&= \int_0^\infty x\frac{f(x)}{Cg(x)}g(x)dx \\
&= E_g[x\frac{f(x)}{Cg(x)}] = E_g[x\frac{f(x)}{Cg(x)}]/E_g[\frac{f(x)}{Cg(x)}], \\
\hat{\theta} &= \frac{1}{n}\Sigma_{i=1}^n X_i\frac{f(X_i)}{Cg(X_i)}/\frac{1}{n}\Sigma_{i=1}^n\frac{f(X_i)}{Cg(X_i)} \\
&= \Sigma_{i=1}^n X_i[\frac{f(X_i)}{g(X_i)}/\Sigma_{i=j}^n\frac{f(X_j)}{g(X_j)}], X_i \sim g
\end{aligned}$$

Note that $g(x)$ is the exponential density with $\lambda = 3$.

```r
set.seed(997)
f <- function(x) return(exp(-(.5-x)^2/2)*exp(-3*x))
N=1000
xi <-  rexp(N, rate = 3)
weight <- (f(xi)/dexp(xi, 3)) # weight i.e., f/g
cat(paste("theta estimate:",round(sum(xi*weight/sum(weight)),4)))
plot(x=1:N, cumsum(xi*weight)/cumsum(weight),
     lwd=1, col = "red", type = "l", xlab = "N",
     ylab = expression(hat(theta)[N]), main = "MC Integral")
```



```
## theta estimate: 0.3231
```