Problem Set 8

Ian McGroarty

03APRIL2020

Problem 2 Exercise 4.10

Show that the Accept-Reject sample $(X_1, \ldots, X_n)$ can be associates with two iid samples; $(U_1, \ldots U_N)$ and $(Y_1 m, \ldots Y_N)$ N is the stopping time associates with the aceptance of n variables Y_j. Then Show that:

$$E_f[h(X)] = \delta_1 = \frac{1}{n}\sum_{i=1}^{n} = \frac{1}{n}\sum_{j=1}^{N} h(Y_j) I_{U_j \le w_j}$$

Note: $I(\phi)$ is a Bernouli random variable with success probability $$

$$\begin{aligned}
Y_1 \cdots Y_N & \qquad Y \sim g \\
U_1 \cdots U_N & \qquad U \sim Norm \\
X_i = Y_i \cdot I(U \le \frac{f(Y_j)}{Mg(Y_j)}) & \qquad \text{By Accept Reject Algorithm} \\
X_1 \cdots X_n & \qquad X \sim f
\end{aligned}$$

So we have n values of X. But for a moment let us consider what N is. n is the total number of accepted Y_j given that we generated N Y_j.

$$n = \sum_{j=1}^{N} I(U \le \frac{f(Y_j)}{Mg(Y_j)})$$

Now we consider:

$$\begin{aligned}
E_f[h(X)] &= \frac{1}{n}\sum_{i=1}^{n} h(X_i) && \text{By definition} \\
&= \frac{1}{n}\sum_{i=1}^{n} h(Y_i \cdot I(U \le \frac{f(Y_j)}{Mg(Y_j)})) && \text{Substituting for } X_i \\
&= \frac{1}{n}\sum_{j=1}^{N} h(Y_i \cdot I(U \le \frac{f(Y_j)}{Mg(Y_j)})) && \text{need to change the sum for j} \\
&= \frac{1}{n}\sum_{j=1}^{N} I(U \le \frac{f(Y_j)}{Mg(Y_j)}) \cdot h(Y_i) && \text{Equivalent because its 0,1} \\
&= \frac{1}{n}\sum_{j=1}^{N} I(U_j \le w_j) \cdot h(Y_i) && \text{Let } w_j = \frac{f(Y_j}{Mg(Y_j)}
\end{aligned}$$

Problem 3: Exercise 4.15 b and c

Exercise 4.15 - Included for notes!

$X|y \sim P(y),\ Y \sim Ga(a,b)$ A is negative Binomial.

The pmf of a negative binomial is

$$P(X=x) = (\frac{x-1}{r-1})p^r(1-p)^{x-r}$$

with $E(X) = \mu = r(1-p)/p$ and $Var(X) = \mu + \mu^2/r^2$. In R we can generate values of a negative Binomial: $X \sim Negbin(5, 0.5)$

```
# Set the iterations
  Nsim <- 1000
  mu <- 5
  r <- 0.5

# Generate X ~Negbin(5,0.5)
  x1 <- rnegbin(1000,mu,5)
# Mean and standard deviation of x1
  mean.x1 <- mean(x1)
  sd.x1 <- sd(x1)/sqrt(Nsim)
  print(paste0("The mean of X=",mean.x1," The standard deviation of x1=",sd.x1))
```

```
## [1] "The mean of X=5.092 The standard deviation of x1=0.0992238952137104"
```

I can also generate values of X by taking advantage of the fact that if

$$X|y \sim Poi(y) \text{ and } Y \sim G(r,(1-p)/p)\$ \implies X \sim Negbin(r,p)$$

I can tke advantage of this relationship and use Rao-Blakwellization to generate an estimate of the mean with a smaller standard deviation. Since \$(1-p)/p = (1-0.5)/.5 = 1 \$ then $Y \sim G(5,1)$.

```
y <- rgamma(Nsim,5,1)
mean.y <- mean(y)
sd.y <- sd(y)/sqrt(Nsim)
  print(paste0("The mean of Y=",mean.y," The standard deviation of y=",sd.y))
```

```
## [1] "The mean of Y=5.01231800203443 The standard deviation of y=0.0696107495303356"
```

Exercise 4.15 (b)

\$ X|y N(0,y), Y G(a,b)\$ Okay so lets say we want to estimate $X \sim T$ with 6 degrees fo freedom.

Since we want $X|y \sim N(0,y)$ and we want the lowest variance, we can just get $Y \sim G(0,0)$ which has the mean of 0 and a variance of 0?

```
Nsim <- 1000
a <- 0
b <- 6
y <- rgamma(Nsim,a,b)
mean.y <- mean(y)
sd.y <- sd(y)/sqrt(Nsim)
  print(paste0("The mean of Y=",mean.y," The standard deviation of y=",sd.y))
```

```
## [1] "The mean of Y=0 The standard deviation of y=0"
```

Exercise 4.15 (c)

$ X|y Bin(n,y), Y Be(a,b) X Beta-binomial.$

If $X \sim Beta - Binomial$ then it has the pmf funciton:

$$P(X = x) = \binom{n}{k} \frac{B(k + a, n - k + \beta)}{B(\alpha, \beta)}$$

With $E[X] = \frac{n\alpha}{\alpha+\beta}$ and $Var(X) = \frac{n\alpha\beta(\alpha+\beta+n)}{(\alpha+\beta)^2(\alpha+\beta+1)}$ In R we can generate a beta binomial with *rbetabinom(n,a,b)*

```
# Set parameters
  Nsim <- 1000
  a <- 4
  b <- 4
  m <- 0.5

x1 <- rbetabinom(Nsim,size = a, m=m,b)
# Mean and standard deviation of x1
  mean.x1 <- mean(x1)
  sd.x1 <- sd(x1)/sqrt(Nsim)
  print(paste0("The mean of X=",mean.x1," The standard deviation of x1=",sd.x1))
```

```
## [1] "The mean of X=2.04 The standard deviation of x1=0.0405221179362832"
```

I can also generate values of X by taking advantage of the fact that if:

$$X|y \sim Bin(n, y),\ Y \sim Be(a, b)$$

With $E[Y] = \frac{\alpha}{\alpha+\beta} \implies E[X] = n \cdot E[Y]$

```
# Set parameters
  Nsim <- 1000
  a <- 4
  b <- 4
  m <- 0.5

# Generate Y ~ Be(a,b)
  y <- rbeta(Nsim,a,b)

# Generate X|y \sim bin(n,y)
  x2 <- rbinom(Nsim,size=a, prob = y)

# Mean and standard deviation of x1
  mean.x2 <- mean(x2)
  sd.x2 <- sd(x2)/sqrt(Nsim)
  print(paste0("The mean of X=",mean.x2," The standard deviation of x1=",sd.x2))
```

```
## [1] "The mean of X=2.042 The standard deviation of x1=0.0363532331101039"
```

Problem 4: Exercise 4.18 (i,ii,iv)

A Naive waty to implement the antithetic variable scheme is to use both U and (1-U) in an inverse simulation. Examine empirically whether this method leads to variance reduction for the distributions:

(i) $f_1(x) = 1/\pi(1 + x^2)$

Wow! I must admit I was skeptical of this at first but the difference is undeniable. They each approach the mean of 0.29, but the dyadic approach makes it so much faster and so much more refined I can't even plot the convergence on the same plot because it just looks like a straight line.

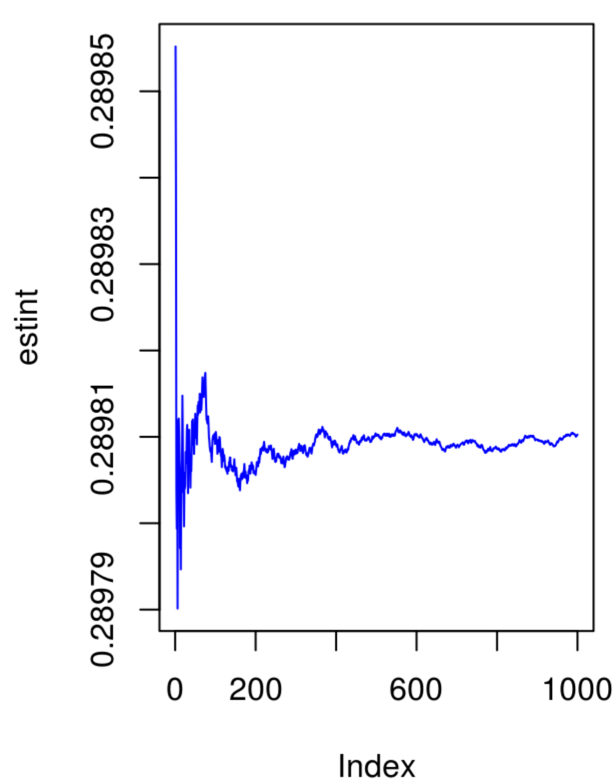
```
# Set parameters
  Nsim <- 1000
  q <- 8
# Set Function
  fx <- function(x){
    1/(pi*1+x^2)
  }

## Generate Uniform
  U <- runif(Nsim,0,1)
  x <- fx(U)
  estx <- cumsum(x)/1:Nsim
  estx[Nsim]
```

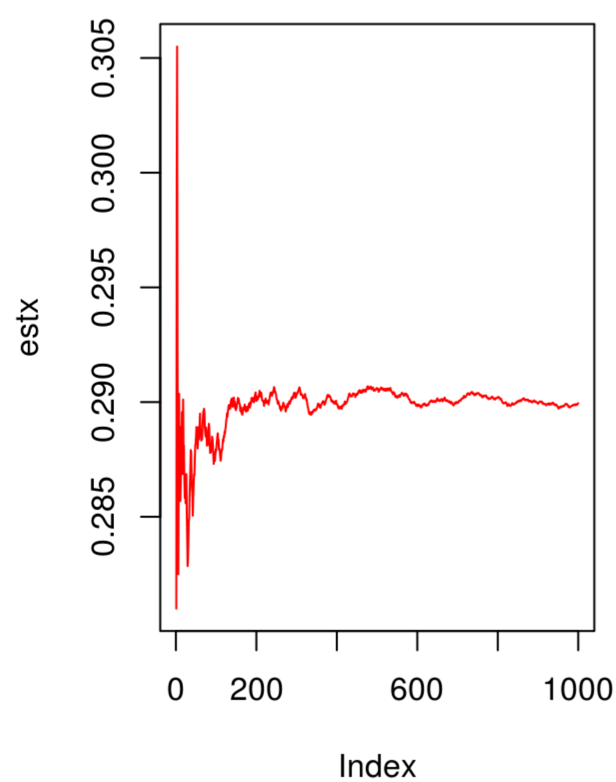
```
## [1] 0.2899497
```

```
## dyadic symetries
  resid <- U%%2^(-q)
  simx <- matrix(resid,ncol=2^q,nrow=Nsim)
  simx[,2^(q-1)+1:2^1] <- 2^(-q)-simx[,2^(q-1)+1:2^1]
  for (i in 1:2^q){
    simx[,i] <- simx[,i] + (i-1)*2^(-q)
  }
  xsym <- fx(simx)
  estint <- cumsum(apply(xsym,1,mean))/(1:Nsim)

## Sum up
  print(paste0("The raw variance is ",var(estx)," The variance with antithetic variable is ", va
r(estint)
              ,"The raw mean is ", mean(estx)," The mean with antithetic variables is ",mean(est
int)))
```

```
## [1] "The raw variance is 1.33514897727641e-06 The variance with antithetic variable is 6.6259
9687097657e-12The raw mean is 0.289814987446302 The mean with antithetic variables is 0.28980908
2921018"
```

```
## Plot
  par(mfrow = c(1,2))
  plot(estint , type = 'l', col = "blue")
  plot(estx, type='l', col = "red")
```



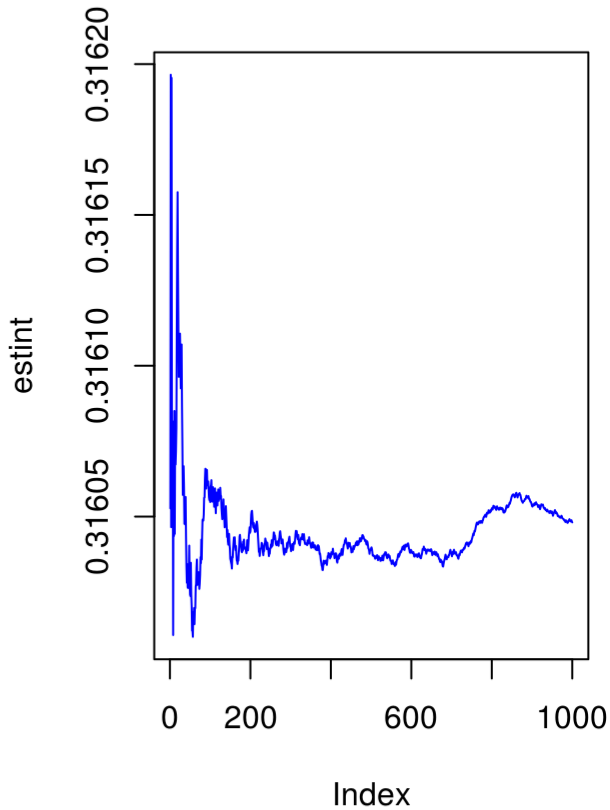


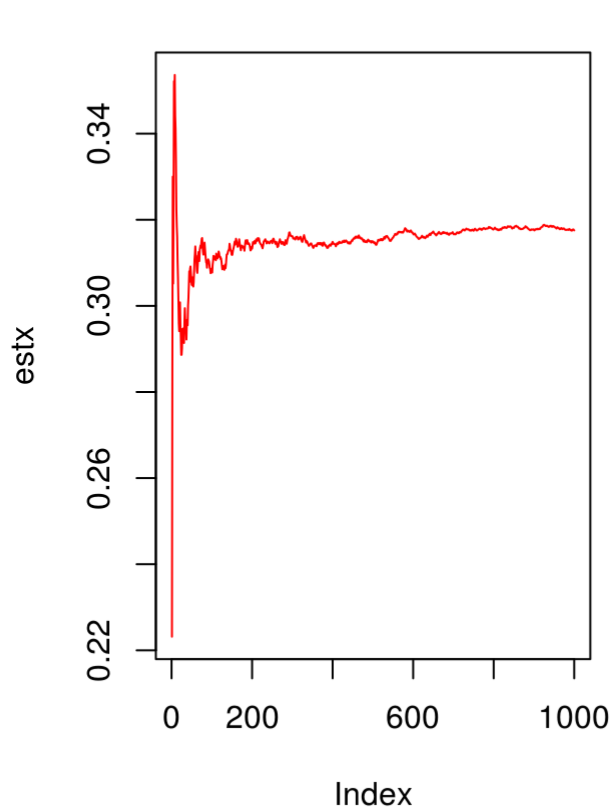
(ii) $f_2(x) = \frac{1}{2}e^{-|x|}$

```
# Set parameters
  Nsim <- 1000
  q <- 8
# Set Function
  fx <- function(x){
    0.5*exp(-1*abs(x))
  }

## Generate Uniform
  U <- runif(Nsim,0,1)
  x <- fx(U)
  estx <- cumsum(x)/1:Nsim

## dyadic symetries
  resid <- U%%2^(-q)
  simx <- matrix(resid,ncol=2^q,nrow=Nsim)
  simx[,2^(q-1)+1:2^1] <- 2^(-q)-simx[,2^(q-1)+1:2^1]
  for (i in 1:2^q){
    simx[,i] <- simx[,i] + (i-1)*2^(-q)
  }
  xsym <- fx(simx)
  estint <- cumsum(apply(xsym,1,mean))/(1:Nsim)

## Sum up
  print(paste0("The raw variance is ",var(estx)," The variance with antithetic variable is ", va
r(estint)
              ,"The raw mean is ", mean(estx)," The mean with antithetic variables is ",mean(est
int)))
```

```
## [1] "The raw variance is 3.22444674964447e-05 The variance with antithetic variable is 1.8731
1083004779e-10The raw mean is 0.315152726615374 The mean with antithetic variables is 0.31604480
8884859"
```

```
## Plot
  par(mfrow = c(1,2))
  plot(estint , type = 'l', col = "blue")
  plot(estx, type='l', col = "red")
```



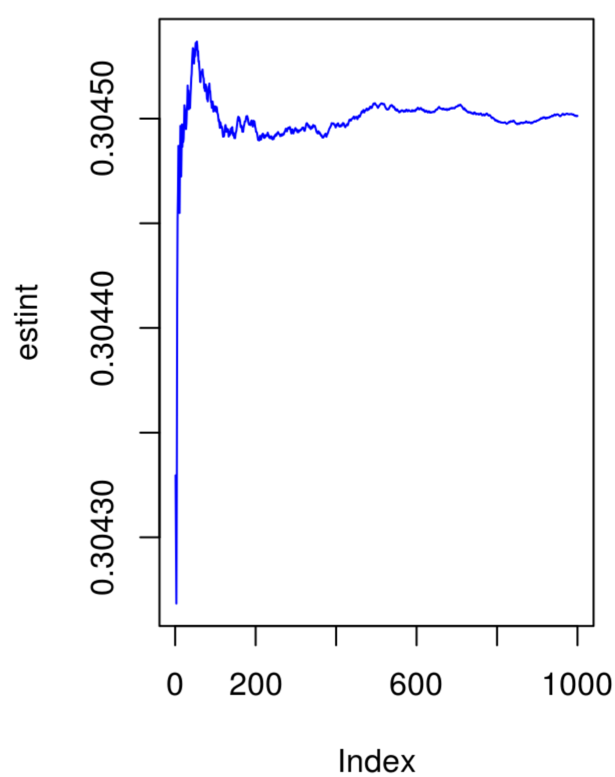


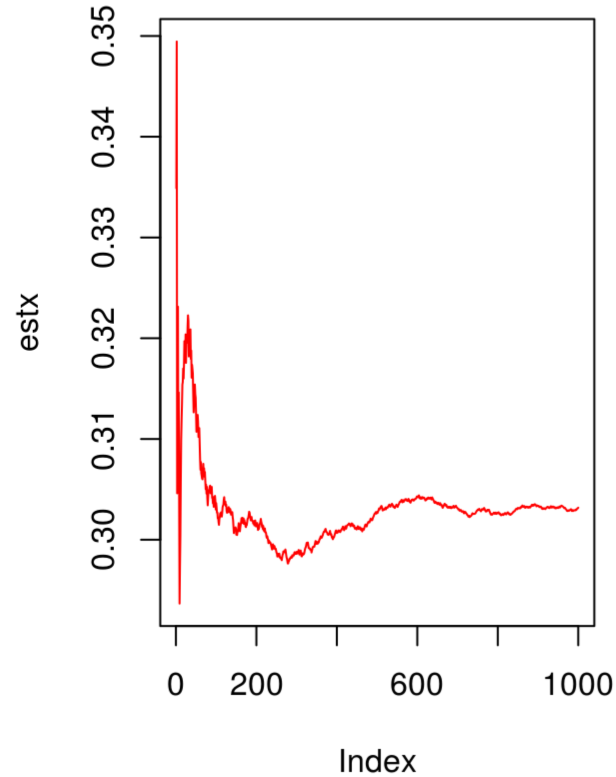
(iv) $f_4(x) = \frac{2}{\pi\sqrt{3}}(1 + x^2/3)^{-2}$

```
# Set parameters
  Nsim <- 1000
  q <- 8
# Set Function
  fx <- function(x){
    (2/(pi*sqrt(3)))*(1+x^2/3)^(-2)
  }

## Generate Uniform
  U <- runif(Nsim,0,1)
  x <- fx(U)
  estx <- cumsum(x)/1:Nsim

## dyadic symetries
  resid <- U%%2^(-q)
  simx <- matrix(resid,ncol=2^q,nrow=Nsim)
  simx[,2^(q-1)+1:2^1] <- 2^(-q)-simx[,2^(q-1)+1:2^1]
  for (i in 1:2^q){
    simx[,i] <- simx[,i] + (i-1)*2^(-q)
  }
  xsym <- fx(simx)
  estint <- cumsum(apply(xsym,1,mean))/(1:Nsim)

## Sum up
  print(paste0("The raw variance is ",var(estx)," The variance with antithetic variable is ", va
r(estint)
              ,"The raw mean is ", mean(estx)," The mean with antithetic variables is ",mean(est
int)))
```

```
## [1] "The raw variance is 1.58208016514588e-05 The variance with antithetic variable is 2.2117
0625011507e-10The raw mean is 0.302987280712266 The mean with antithetic variables is 0.30449967
4030392"
```

```
## Plot
  par(mfrow = c(1,2))
  plot(estint , type = 'l', col = "blue")
  plot(estx, type='l', col = "red")
```





Problem 5:

Assume you use antithetic variables to estimate some parameter of the standard normal distribution. Prove, in this case, that the covariance between X_i and Y_i is -1

If we use Antithetic variables then we know that they have the same variance because they are drawn from the same distribution. In fact, since they are both drawn from the standard normal distribution we know that $Var(X) = Var(Y) = \sigma = 1$. We will see below

$$
\begin{aligned}
Var(aX + bY) &= Var(X + (1 - X)) = 0 \\
Var(aX + bY) &= a^2 Var(X) + b^2 Var(Y) + 2abCov(X, Y) && \text{Theorem 3.9.5 (188)} \\
&= Var(X) + Var(Y) + 2Cov(X, Y) && \text{Let a,b = 1} \\
&= 2\sigma + 2\rho && Var(X) = Var(Y) \\
0 &= 2 + 2\rho && X \sim N(0, 1) \\
\rho &= -1
\end{aligned}
$$