Joseph High
9E1FDC

HW 10 R Code and Results

```
> #Problem A
> #Part (a)
> P = pnorm(3.5, mean = 0, sd = 1, lower.tail = FALSE)  #Probability X
> 3.5 (Uppertail only)
> L = 2*P  #Since N(0,1) is a symmetric distribution, P(abs(X) > 3.5)
= 2*P(X > 3.5)
> #Thus, l = P(abs(X) > 3.5) = 0.0004652582
>
>
> #Part (b)  - Estimating using CMC
> set.seed(1)
> N = 10^5                  #number of independent samples
> x = matrix(NA, N, 10);  #storage for the 10 estimates
> H = matrix(NA, N, 10);  #storage for the 10 estimates
> L.hat = matrix(NA, 1, 10);
> for(j in 1:10){
+   x[,j] = rnorm(N, mean = 0, sd = 1)  # sampling from f
+   for (i in 1:N) {
+     if (x[i,j] >= qnorm(1-P)) {
+       H[i,j] = 1
+     }
+     else if (x[i,j] <= qnorm(P)) {
+       H[i,j] = 1
+     }
+     else {
+       H[i,j] = 0
+     }
+   }
+   for(k in 1:10) {
+     L.hat[,k] = (1/N)*sum(H[,k])
+   }
+ }
>
> show(L.hat)  #list of all 10 estimates of Prob(abs(X) >= 3.5)
        [,1]    [,2]    [,3]  [,4]    [,5]  [,6]    [,7]    [,8]
[,9]
[1,] 0.00047 0.00045 0.00054 5e-04 0.00044 3e-04 0.00037 0.00051
0.00058
       [,10]
[1,] 0.00044
>
> #Variance of L.hat:
> var.Lhat = (1/N)*(L - L^2)
>
> var.Lhat  #Var(L.hat) = ~ 4.650417e-09
[1] 4.650417e-09
>
>
> #Part (c)  - Estimating using IS with g = N(3.5, 1):
```

Joseph High
9E1FDC

```
> y = matrix(NA, N, 10);
> h = matrix(NA, N, 10);
> w = matrix(NA, N, 10);
> f = matrix(NA, N, 10);
> g = matrix(NA, N, 10);
> Lhat.IS = matrix(NA, 1, 10);
> for(j in 1:10){
+   y[,j] = rnorm(N, mean = 3.5, sd = 1) #sampling from g
+   f[,j] = dnorm(y[,j], mean = 0, sd = 1)
+   g[,j] = dnorm(y[,j], mean = 3.5, sd = 1)
+   w[,j] = f[,j]/g[,j]   #weights
+   for (i in 1:N) {
+     if (y[i,j] >= qnorm(1-P)) {
+       h[i,j] = 1
+     }
+     else {
+       h[i,j] = 0
+     }
+   }
+   for(k in 1:10) {
+     Lhat.IS[,k] = 2*(1/N)*sum(h[,k]*w[,k])  #Since g is symmetric,
multiply
+                                             #by 2 to attain an
estimate for
+                                             #both tails.
+     }
+ }
> show(Lhat.IS)
             [,1]         [,2]        [,3]         [,4]         [,5]
[1,] 0.0004628242 0.0004625311 0.000467592 0.0004650986 0.0004611431
            [,6]         [,7]         [,8]         [,9]        [,10]
[1,] 0.000460414 0.0004627089 0.0004642176 0.0004686603 0.0004671747
>
> #Variance of Lhat.IS:
> var.LhatIS = var(Lhat.IS[,])
> var.LhatIS  #Variance of Importance Sampling estimate = ~ 7.958055e-
12
[1] 7.958055e-12
>
> #Part (d)
> #Relative Error, RE = sqrt(Var(L.hat))/E(L.hat) = sqrt(Var(L.hat))/L
> RE.Standard = sqrt(var.Lhat)/L
> RE.IS = sqrt(var.LhatIS)/L
> RE.Standard  #0.1465723 ~ 14.66% error relative to the true value
[1] 0.1465723
> RE.IS  #0.006063306 ~ 0.61% error relative to the true value
[1] 0.006063306
>
> #The Relative error for the Importance Sampling method is
significantly
> #smaller than the Relative Error for the standard CMC method. Thus,
the
```

Joseph High
9E1FDC

```
> #relative accuracy of the IS estimates to the true value is much 
greater
> #than that of the standard CMC estimates. Indeed, the IS method uses 
a
> #standard normal distribution centered at 3.5, significantly 
increasing the
> #probability that samples of X >=3.5 are generated.
>
> #Extra Credit  Part (e)
y = matrix(NA, N, 10);
h = matrix(NA, N, 10);
w = matrix(NA, N, 10);
f = matrix(NA, N, 10);
g = matrix(NA, N, 10);
Lhat.IS = matrix(NA, 1, 10);
for(j in 1:10){
  y[,j] = rnorm(N, mean = mu*, sd = 1) #sampling from g
  f[,j] = dnorm(y[,j], mean = 0, sd = 1)
  g[,j] = dnorm(y[,j], mean = mu*, sd = 1)
  w[,j] = f[,j]/g[,j]   #weights
  for (i in 1:N) {
    if (y[i,j] >= qnorm(1-P)) {
      h[i,j] = 1
    }
    else {
      h[i,j] = 0
    }
  }
  for(k in 1:10) {
    Lhat.IS[,k] = 2*(1/N)*sum(h[,k]*w[,k])  #Since g is symmetric, 
multiply
                                            #by 2 to attain an 
estimate for
                                            #both tails.
    }
}
```