STAT4CI3/6CI3 Computational Methods for Inference

Assignment 2 Solutions

R code for this solution in a plain text file is also available separately

Q. 1 **a)** Since U_1 and U_2 are independent and identically distributed random variables and the same transformation is made to both we see that X and Y are also independent and identically distributed random variables. It therefore suffices to get the distribution of X.

$$\begin{aligned} F_X(x) &= \mathrm{P}(X \leqslant x) \\ &= \mathrm{P}(-\log U_1 \leqslant x) \\ &= \mathrm{P}\left(U_1 \geqslant \mathrm{e}^{-x}\right) \\ &= 1 - F_{U_1}\left(\mathrm{e}^{-x}\right) \\ &= \begin{cases} 0 & x \leqslant 0 \\ 1 - \mathrm{e}^{-x} & x > 0 \end{cases} \end{aligned}$$

We recognize this as the cumulative distribution function of the exponential(1) distribution.

Hence X and Y are independent exp(1) random variables. **[3/2 marks]**

b) For $y > 0$ we have

$$\begin{aligned} f_Y(y \mid 2X \geqslant (Y-1)^2) &= \frac{\mathrm{P}(2X \geqslant (Y-1)^2 \mid Y = y) f_Y(y)}{\displaystyle\int_0^\infty \mathrm{P}(2X \geqslant (Y-1)^2 \mid Y = y) f_Y(y)\, dy} \\ &\propto \mathrm{P}\left(X \geqslant \frac{1}{2}(y-1)^2\right) f_Y(y) \\ &= \exp\left\{-\frac{1}{2}(y-1)^2\right\} \mathrm{e}^{-y} \\ &= \exp\left\{-\frac{1}{2}(y^2+1)\right\} \\ &\propto \mathrm{e}^{-y^2/2} \end{aligned}$$

[4/3 marks]

For $y \leqslant 0$, $f_Y(y) = 0$ so $f_Y(y \mid 2X \geqslant (Y-1)^2) = 0$ also. **[0/1 marks]**

c) Consider the transformation

$$x = \frac{y^2}{2} \;\Rightarrow\; y = \sqrt{2x} \;\Rightarrow\; dy = \frac{dx}{\sqrt{2x}}$$

Hence the required integral is

$$\begin{aligned}
\int_0^\infty \mathrm{e}^{-y^2/2}\, dy &= \int_0^\infty \mathrm{e}^{-x} \frac{dx}{\sqrt{2x}} \\
&= \frac{1}{\sqrt{2}} \int_0^\infty x^{-0.5}\, \mathrm{e}^{-x}\, dx \\
&= \frac{\Gamma(0.5)}{\sqrt{2}} \\
&= \sqrt{\frac{\pi}{2}}
\end{aligned}$$

using the fact that $\Gamma(0.5) = \sqrt{\pi}$. **[5/4 marks]**

Since we saw in part (b) that the density of the accepted observations is

$$f_Y(y \mid Y \text{ accepted}) \;\propto\; \mathrm{e}^{-y^2/2}$$

we have that

$$f_Y(y \mid Y \text{ accepted}) = \frac{\mathrm{e}^{-y^2/2}}{\displaystyle\int_0^\infty \mathrm{e}^{-y^2/2}\, dy} = \sqrt{\frac{2}{\pi}}\, \mathrm{e}^{-y^2/2} \qquad \text{for } y > 0$$

[2/1 marks]

d) The easiest way to show this is to show that $F_Z(z) = \mathrm{P}(Z \leqslant z)$ is the cdf of the standard normal $\Phi(z)$.

Suppose that $z < 0$ then $Z \leqslant z$ if, and only if, $U_3 \leqslant 0.5$ and $-Y \leqslant z$. U_3 and Y are independent so we have

$$\begin{aligned}
F_Z(z) &= \mathrm{P}(-Y \leqslant z)\mathrm{P}(U_3 \leqslant 0.5) \\
&= \frac{1}{2}\mathrm{P}(Y \geqslant -z) \\
&= \frac{1}{2} \int_{-z}^\infty \sqrt{\frac{2}{\pi}}\, \mathrm{e}^{-y^2/2}\, dy \\
&= \int_{-z}^\infty \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-y^2/2}\, dy \\
&= \int_{-\infty}^z \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-x^2/2}\, dx \quad (x = -y) \\
&= \Phi(z)
\end{aligned}$$

[3/2 marks]

Now if $z \geqslant 0$ then $Z \leqslant z$ if $U_3 \leqslant 0.5$ (because in that case $Z < 0 < z$) or if $U_3 > 0.5$ and $Y \leqslant z$. So we have

$$\begin{aligned}
F_Z(z) &= \mathrm{P}(U_3 \leqslant 0.5) + \mathrm{P}(U_3 > 0.5)\mathrm{P}(Y \leqslant z) \\
&= \frac{1}{2} + \frac{1}{2}\int_0^z \sqrt{\frac{2}{\pi}}\, \mathrm{e}^{-y^2/2}\, dy \\
&= \frac{1}{2} + \int_0^z \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-y^2/2}\, dy \\
&= \int_{-\infty}^0 \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-y^2/2}\, dy + \int_0^z \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-y^2/2}\, dy \\
&= \int_{-\infty}^z \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-y^2/2}\, dy \\
&= \Phi(z)
\end{aligned}$$

Hence we have $F_Z(z) = \Phi(z)$ for every $x \in \mathbb{R}$ and so $Z \sim \mathrm{N}(0,1)$. **[3/3 marks]**

e) Here is a function to generate standard normals using this method

```
rnorm.ar <- function(n) {
#
# Function to use accept-reject method to generate n
# standard normal random variates.
#
  n1 <- n #n1 will count how many observations we still need.
  Z <- rep(NA, n)
  while (n1>0) {
    X <- -log(runif(n1))
    Y <- -log(runif(n1))
    accept <- 2*X >= (Y-1)^2
    n2 <- sum(accept)
    Y <- Y[accept]
    U3 <- runif(n2)
    Z[(n-n1+1):(n-n1+n2)] <- ifelse(U3<0.5, -Y, Y)
    n1 <- n1-n2
  }
  Z
}
```

[5/4 marks]

Q. 2 **a)** Suppose that $U \sim \text{Unif}[0,1]$ and let $X = -\ln U$. Then from Question 1 we know that $X \sim \exp(1)$ and this is a distribution which has all of its moments so we can safely apply the Monte Carlo technique.

Thus we have

$$\begin{aligned}\Gamma(\alpha) &= \int_0^\infty x^{\alpha-1}\,\mathrm{e}^{-x}\,dx \\ &= \mathrm{E}\left(X^{\alpha-1}\right) \\ &= \mathrm{E}\left[(-\ln U)^{\alpha-1}\right]\end{aligned}$$

Thus a Monte Carlo estimate of $\Gamma(\alpha)$ is

$$\hat{\Gamma}(\alpha) = \frac{1}{N}\sum_{i=1}^{N}(-\ln u_i)^{\alpha-1}$$

where $u_1, \ldots, u_N$ are generated from the Unif$[0,1]$ distribution. **[6/4 marks]**

b)

$$\begin{aligned}\mathrm{Var}\left(\hat{\Gamma}(\alpha)\right) &= \frac{1}{N}\mathrm{Var}\left[(-\ln U)^{\alpha-1}\right] \\ &= \frac{1}{N}\left\{\mathrm{E}\left[(-\ln U)^{2\alpha-2}\right] - \left[\mathrm{E}\left[(-\ln U)^{\alpha-1}\right]^2\right]\right\} \\ &= \frac{1}{N}\left\{\int_0^1 (-\ln u)^{2\alpha-2}\,du - \left[\int_0^1 (-\ln u)^{\alpha-1}\,du\right]^2\right\} \\ &= \frac{1}{N}\left\{\int_0^\infty x^{2\alpha-2}\,\mathrm{e}^{-x}\,dx - \left[\int_0^\infty x^{\alpha-1}\,\mathrm{e}^{-x}\,dx\right]^2\right\} \\ &= \frac{1}{N}\left[\Gamma(2\alpha-1) - \left[\Gamma(\alpha)\right]^2\right]\end{aligned}$$

It is important to note that this works only for $\alpha > 0.5$ since we require $2\alpha - 1 > 0$ for the variance to exist. **[6/4 marks]**

c) An R function to estimate the gamma integral using Monte Carlo and return both the estimate and its standard error is

```
gamma.est <- function(N, alpha) {
  #
  # Function to estimate Gamma(alpha) by Monte Carlo using a
  # simulation size of N.
  #
  # The function requires alpha>0.5 to get a finite variance.
  #
  if (alpha<=0.5)
    stop("This method cannot be used with alpha<=0.5")
  if (N<2)
    stop("The number of Monte Carlo replicates must be greater than 1")
  if (N!=ceiling(N)) {
```

```
    warning("N rounded up to ",ceiling(N))
    N <- ceiling(N)
  }
  U <- runif(N)
  X <- -log(U)
  Xa1 <- X^(alpha-1)
  gamma.hat <- mean(Xa1)
  temp <- mean(Xa1^2)
  se <- sqrt((temp-gamma.hat^2)/N)
  c(gamma.hat, se)
}
```

Note that for $\alpha = 1$ this algorithm will always give exactly the right answer $\Gamma(1) = 1$.

[6/5 marks]

The following code will examine the performance of the algorithm for various values of $\alpha > 0.5$ and simulation sizes N. The results are presented in a table on the next page.

```
N <- 10^(2:5)
al <- c(0.51, 0.6, 0.75, 0.95, 1.1, 1.25, 1.5, 2, 3, 4, 5)
set.seed(240219)
results <- matrix(NA, nrow=length(al), ncol=2*length(N))
for (i in 1:length(al))
  for (j in 1:length(N))
      results[i,(2*j-1):(2*j)] <- gamma.est(N[j], al[i])
results <- cbind(gamma(al), results)
rownames(results) <- paste("alpha=",al, sep="")
colnames(results) <- c("Value", paste(c("est","se"), rep(N, each=2), sep="."))
```

Here is the table of results

		$N = 100$		$N = 1000$		$N = 10000$		$N = 100000$	
α	$\Gamma(\alpha)$	$\hat{\Gamma}(\alpha)$	se	$\hat{\Gamma}(\alpha)$	se	$\hat{\Gamma}(\alpha)$	se	$\hat{\Gamma}(\alpha)$	se
0.51	1.738	1.375	0.074	1.679	0.058	1.714	0.023	1.733	0.009
0.6	1.489	1.283	0.060	1.451	0.035	1.485	0.013	1.498	0.005
0.75	1.225	1.292	0.059	1.227	0.016	1.230	0.005	1.223	0.002
0.95	1.031	1.029	0.006	1.032	0.002	1.032	0.001	1.032	0.000
1.1	0.951	0.947	0.013	0.954	0.004	0.951	0.001	0.951	0.000
1.25	0.906	0.882	0.025	0.914	0.008	0.906	0.003	0.904	0.001
1.5	0.886	0.918	0.051	0.880	0.015	0.884	0.005	0.886	0.001
2	1.000	1.014	0.105	1.032	0.033	1.001	0.010	0.999	0.003
3	2.000	2.708	0.522	1.906	0.143	1.965	0.045	1.982	0.014
4	6.000	5.940	1.430	5.785	0.724	5.906	0.273	5.838	0.081
5	24.000	8.121	2.406	19.900	3.798	22.747	1.613	23.993	0.607

From this table we see that the standard error goes down with increasing N as expected. For $\alpha < 1$ the standard error goes down as α approaches 1 but then it increases quite quickly for increasing $\alpha > 1$. Even for values of α of 4 or 5 we need very large sample sizes to get an estimate which we are confident is accurate to the first place of decimal.

[7/7 marks]

Q. 3 First we note that the true value of I can be found using numeric integration methods to be $I = 1.462652$. Although this is not necessary for the question, it does provide us with a good check on our methods. If the Monte Carlo estimates are not close to this value then it is likely that there is something wrong with our derivation or coding.

a) For the regular Monte Carlo method we note that

$$I \;=\; \mathrm{E}\left(\mathrm{e}^{X^2}\right) \qquad \text{where } X \sim \text{Uniform}(0,1)$$

Hence we have the estimator

$$\hat{I}_{mc} \;=\; \frac{1}{N}\sum_{i=1}^{N} \mathrm{e}^{X_i^2} \qquad \text{where } X_1 \ldots, X_N \overset{iid}{\sim} \text{Uniform}(0,1)$$

The variance of this estimator is

$$\mathrm{Var}(\hat{I}_{mc}) \;=\; \frac{\mathrm{Var}\left(\mathrm{e}^{X^2}\right)}{N} \;=\; \frac{\mathrm{E}\left(\mathrm{e}^{2X^2}\right) - \left(\mathrm{E}\left(\mathrm{e}^{X^2}\right)\right)^2}{N}$$

We can use Monte Carlo estimators of these two expectations to get the standard error

$$\mathrm{se}\left(\hat{I}_{mc}\right) \;=\; \frac{1}{\sqrt{N}}\sqrt{\frac{1}{N}\sum_{i=1}^{N} \mathrm{e}^{2X_i^2} - \hat{I}_{mc}^2}$$

[3/3 marks]

The following code implements this in R.

```
> N <- 100000
> set.seed(24022019)
> X <- runif(N)
> Ihat.mc <- mean(exp(X^2))
> se.Ihat.mc <- sqrt((mean(exp(2*X^2))-Ihat.mc^2)/N)
> round(c(Ihat.mc, se.Ihat.mc),6)
[1] 1.461395 0.001496
```

[2/1 marks]

b) For the control variable method with

$$C \;=\; \frac{1}{N}\sum_{i=1}^{N} X_i^2$$

and we note that if $X \sim \text{Uniform}(0,1)$ then

$$\mathrm{E}\left(X^2\right) \;=\; \frac{1}{3}$$

which gives us the value of μ for the control variable.
To get the best variance reduction we should estimate the optimal β

$$\beta \;=\; \frac{\mathrm{Cov}(\hat{I}_{mc}, C)}{\mathrm{Var}(C)} \;=\; \frac{\mathrm{Cov}\left(\mathrm{e}^{X^2}, X^2\right)}{\mathrm{Var}(X^2)}$$

Now

$$\text{Var}(X^2) \;=\; \text{E}(X^4) - \left(\text{E}(X^2)\right)^2 \;=\; \frac{1}{5} - \left(\frac{1}{3}\right)^2 \;=\; \frac{4}{45}$$

We can use the known value of the variance in the denominator but we will need to estimate the covariance using the simulated $X_1, \ldots, X_N$ so we get

$$\hat{\beta} \;=\; 11.25\widehat{\text{Cov}}\left(\text{e}^{X^2}, X^2\right)$$

For my data I estimate the optimal β to be $\hat{\beta} = 1.56951$ using the code below.

[3/2 marks]

The estimator is then

$$\hat{I}_{C2} \;=\; \hat{I}_{mc} - \hat{\beta}\left(\frac{1}{N}\sum_{i=1}^{N} X_i^2 - \frac{1}{3}\right)$$

and the standard error is

$$\text{se}\left(\hat{I}_{C1}\right) \;=\; \sqrt{\text{se}^2\left(\hat{I}_{mc}\right) + 4\hat{\beta}^2/(45N) - 2\hat{\beta}\widehat{\text{Cov}}\left(X_i^2, \text{e}^{X_i^2}\right)/N}$$

[2/1 marks]

In R we have

```
> C <- mean(X^2)
> mu <- 1/3
> cov <- cov(exp(X^2), X^2)
> beta <- 11.25*cov
> beta
[1] 1.56951
> Ihat.C <- Ihat.mc-beta*(C-mu)
> se.Ihat.C <- sqrt(se.Ihat.mc^2+beta^2/(11.25*N)-2*beta*cov/N)
> round(c(Ihat.C, se.Ihat.C),6)
[1] 1.462402 0.000219
```

Use of the control variable reduces the standard error so now it is more than a factor of 7 lower than for the basic Monte Carlo estimator. **[2/2 marks]**

c) For the antithetic variable method we have the two estimators

$$\hat{I}_1 \;=\; \hat{I}_{mc} \;=\; \frac{1}{N}\sum_{i=1}^{N} \text{e}^{X_i^2} \qquad \hat{I}_2 \;=\; \frac{1}{N}\sum_{i=1}^{N} \text{e}^{(1-X_i)^2}$$

and the antithetic variable estimator

$$\hat{I}_A \;=\; 0.5(\hat{I}_1 + \hat{I}_2)$$

[2/1 marks]

From class notes we have

$$\begin{aligned}\mathrm{Var}(\hat{I}_A) &= \frac{\mathrm{Var}(\hat{I}_1)}{4} + \frac{\mathrm{Var}(\hat{I}_2)}{4} + \frac{\mathrm{Cov}(\hat{I}_1, \hat{I}_2)}{2} \\ &= \frac{\mathrm{Var}\left(\mathrm{e}^{X^2}\right)}{4N} + \frac{\mathrm{Var}\left(\mathrm{e}^{(1-X)^2}\right)}{4N} + \frac{\mathrm{Cov}\left(\mathrm{e}^{X^2}, \mathrm{e}^{(1-X)^2}\right)}{2N}\end{aligned}$$

We can use the simulated $X_1, \ldots, X_N$ to estimate these variances and the covariance and then take the square root to get the standard error. **[2/2 marks]**

In R we have

```
> Ihat1 <- Ihat.mc
> Ihat2 <- mean(exp((1-X)^2))
> Ihat.A <- (Ihat1+Ihat2)/2
> vI1 <- var(exp(X^2))/N
> vI2 <- var(exp((1-X)^2))/N
> covI1I2 <- cov(exp(X^2), exp((1-X)^2))/N
> se.Ihat.A <- sqrt(vI1/4+vI2/4+covI1I2/2)
> round(c(Ihat.A, se.Ihat.A),6)
[1] 1.461845 0.000527
```

The antithetic variable method in this case has reduced variability relative to the basic Monte Carlo method but it is not as good as the control variate method from part (b). **[2/2 marks]**

d) The crucial thing here is estimation of the α parameter to use. Clearly we need $\alpha > 1$ so that the beta density is increasing in x as is the integrand. The simplest method is to use a range of possible values of α and see which one gives closest to a constant ratio between the integrand and the beta density. There are a number of ways to evaluate this. In this solution I will evaluate both functions at an evenly spaced grid of x points and examine the variance of the ratio.

```
> xx <- seq(0,1,by=0.01)[-1]
> alpha <- seq(1,2, by=0.1)
> vars <- rep(NA, 10)
> for (i in 1:10)
+   vars[i] <- var(exp(xx^2)/dbeta(xx,alpha[i],1))
> a1 <- alpha[which.min(vars)]
> alpha <- seq(a1-0.1, a1+0.1, by=0.01)[-1]
> vars <- rep(NA, 20)
> for (i in 1:20)
+   vars[i] <- var(exp(xx^2)/dbeta(xx,alpha[i],1))
> alpha[which.min(vars)]
[1] 1.24
```

In my code I found a good value in two stages but this is not necessary. Any well-justified method which results in a value of α around 1.2–1.3 is acceptable. **[4/4 marks]**

We now generate from this beta distribution, calculate the appropriate weights and construct our estimate and its standard error. For the standard error we can use the result on Page 3-23 of my notes.

```
> set.seed(24022019)
> X.beta <- rbeta(N, 1.24, 1)
> W <- 1/dbeta(X.beta, 1.24, 1)
> hX <- exp(X.beta^2)
> Ihat.IS <- mean(hX*W)
> se.Ihat.IS <- sqrt((mean((hX*W)^2)-Ihat.IS^2)/N)
> round(c(Ihat.IS, se.Ihat.IS),6)
[1] 1.460748 0.000956
```

Again this method improves the standard error relative to the basic Monte Carlo method but only by a factor of about 1.5. For this example it is not as good as either the control variable or antithetic variable method. **[3/2 marks]**

Q. 4 **a)** If $X_1, \ldots, X_n \overset{iid}{\sim} \text{normal}(\mu, \sigma^2)$ then we can write $X_i = \mu + \sigma Z_i$ where $Z_1, \ldots, Z_n \overset{iid}{\sim} \text{normal}(0, 1)$. Hence for any sample we have

$$\overline{x} = \mu + \sigma \overline{z} \qquad s_x = \sigma s_z$$

Hence we can write

$$\begin{aligned} \overline{x} - z_{\alpha/2}\frac{s_x}{\sqrt{n}} < \mu &\iff (\mu + \sigma\overline{z}) - z_{\alpha/2}\frac{\sigma s_z}{\sqrt{n}} < \mu \\ &\iff \sigma\overline{z} - z_{\alpha/2}\frac{\sigma s_z}{\sqrt{n}} < 0 \\ &\iff \overline{z} - z_{\alpha/2}\frac{s_z}{\sqrt{n}} < 0 \end{aligned}$$

and similarly

$$\overline{x} + z_{\alpha/2}\frac{s_x}{\sqrt{n}} > \mu \iff \overline{z} + z_{\alpha/2}\frac{s_z}{\sqrt{n}} > 0$$

Hence the interval based on the sample $x_1, \ldots, x_n$ contains the true mean μ if, and only if, the interval based on $z_1, \ldots, z_n$ contains 0 and this is true for any μ and $\sigma > 0$ so we need only consider sampling from the standard normal. We shall examine a range of values of n and for each one calculate the empirical coverage probabilities based on $R = 10000$ samples and give the standard error of the estimate. **[5/2 marks]**

Here is the R code to run this simulation study.

```
n <- c(5*(1:10),10*(6:10),200,500,1000,5000,10000)
maxn <- max(n)
N <- 10000
# I will generate all of the random variables in one go and store them in a matrix.
set.seed(220214)
Zmat <- matrix(rnorm(N*maxn), nrow=N)
alpha <- 0.05
z.alpha <- qnorm(1-alpha/2)
results.4a <- matrix(NA,ncol=3,nrow=length(n))
colnames(results.4a) <- c("n", "Cover", "se(Cover)")

for (i in 1:length(n)) {
   ni <- n[i]
   # the first n[i] observations in each row can be considered the sample.
   zbar <- rowMeans(Zmat[,1:ni])
   sez <- apply(Zmat[,1:ni],1,sd)/sqrt(ni)
   lower <- zbar-z.alpha*sez
   upper <- zbar+z.alpha*sez
   p <- mean(lower<0 & upper >0)
   sep <- sqrt(p*(1-p)/R)
   results.4a[i,] <- c(n[i],p,sep)
}
```

[4/2 marks]

The code above produces the following results

n	Cover	se(Cover)
5	0.8784	0.0033
10	0.9235	0.0027
15	0.9322	0.0025
20	0.9346	0.0025
25	0.9373	0.0024
30	0.9407	0.0024
35	0.9463	0.0023
40	0.9445	0.0023
45	0.9425	0.0023
50	0.9437	0.0023
60	0.9426	0.0023
70	0.9452	0.0023
80	0.9443	0.0023
90	0.9471	0.0022
100	0.9481	0.0022
200	0.9476	0.0022
500	0.9493	0.0022
1000	0.9518	0.0021
5000	0.9493	0.0022
10000	0.9515	0.0021

The coverage is generally increasing as n increases although there is some simulation variability so for some sample sizes it appears to go down slightly. This is just simulation variability and does not reflect a true drop in the coverage. The first time the estimated coverage exceeds 94% is when $n = 30$ and it never goes below this value again so we would be comfortable in saying that a sample size of $n = 30$ or more will give coverage within 1 percentage point of the nominal 95%. **[3/2 marks]**

An alternative method which is also valid is to note that the coverage can be written as

$$\begin{aligned} \mathrm{P}\left(\overline{X} - z_{\alpha/2}\frac{S}{\sqrt{n}} < \mu < \overline{X} + z_{\alpha/2}\frac{S}{\sqrt{n}}\right) &= \mathrm{P}\left(-z_{\alpha/2} < \frac{\sqrt{n}(\overline{X}-\mu)}{S} < z_{\alpha/2}\right) \\ &= \mathrm{P}\left(-z_{\alpha/2} < T_{n-1} < z_{\alpha/2}\right) \end{aligned}$$

where T_{n-1} is a Student's t random variable with $n-1$ degrees of freedom. Then we could estimate the coverage using Monte Carlo as follows

```
results.4a1 <- results.4a
set.seed(24022019)
for (i in 1:length(n)) {
  ni <- n[i]
  T <- rt(N, n[i]-1)
  p1 <- mean(T<z.alpha & T > -z.alpha)
  sep1 <- sqrt(p1*(1-p1)/R)
  results.4a1[i,] <- c(n[i],p1,sep1)
}
```

which gives very similar results to the earlier method. The problem is that it does not generalize to other location-scale families and so this method does not work for part (b). Nevertheless it is a valid solution for part (a).

b) For the exponential distribution we can also take advantage of the fact that

$$X \sim \exp(\mu) \iff X = \mu Z \quad \text{where } Z \sim \exp(1).$$

Hence we have

$$\overline{x} = \mu\overline{z} \qquad s_x = \mu s_z$$

and so

$$\overline{x} - z_{\alpha/2}\frac{s_x}{\sqrt{n}} < \mu \iff \mu\overline{z} - z_{\alpha/2}\frac{\mu s_z}{\sqrt{n}} < \mu$$

$$\iff \overline{z} - z_{\alpha/2}\frac{s_z}{\sqrt{n}} < 1$$

and similarly for the upper limit so that the interval based on a sample $X_1, \ldots, X_n$ from and exponential distribution with mean μ covers the true μ if, and only if, the corresponding sample from the exponential distribution with mean 1 covers the value 1. Hence we need only concern ourselves with the standard exponential distribution. The code is identical to that for the normal case except that the data is now generated from an exponential distribution with mean 1 and we look at coverage of $\mu = 1$. Since the mean μ is known to be positive for this type of data, we shall also find the proportion of times that the lower endpoint of the interval is actually negative. **[5/2 marks]**

Here is the R code I used

```
set.seed(220214)
Zmat.exp <- matrix(rexp(N*maxn), nrow=N)
alpha <- 0.05
z.alpha <- qnorm(1-alpha/2)
results.4b <- matrix(NA,ncol=5,nrow=length(n))
colnames(results.4b) <- c("n", "P(lo<0)","se(P(lo<0))", "Cover", "se(Cover)")

for (i in 1:length(n)) {
   ni <- n[i]
   zbar <- rowMeans(Zmat.exp[,1:ni])
   sez <- apply(Zmat.exp[,1:ni],1,sd)/sqrt(ni)
   lower <- zbar-z.alpha*sez
   upper <- zbar+z.alpha*sez
   p0 <- mean(lower<0)
   se0 <- sqrt(p0*(1-p0)/R)
   p1 <- mean(lower<1 & upper>1)
   se1 <- sqrt(p1*(1-p1)/R)
   results.4b[i,] <- c(ni,p0,se0,p1,se1)
}
```

[4/2 marks]

Here are the results that I found.

n	P(lo<0)	se(P(lo<0))	Cover	se(Cover)
5	0.1623	0.0037	0.8097	0.0039
10	0.0092	0.0010	0.8687	0.0034
15	0.0001	0.0002	0.8897	0.0031
20	0.0000	0.0000	0.9010	0.0030
25	0.0000	0.0000	0.9115	0.0028
30	0.0000	0.0000	0.9134	0.0028
35	0.0000	0.0000	0.9171	0.0028
40	0.0000	0.0000	0.9203	0.0027
45	0.0000	0.0000	0.9213	0.0027
50	0.0000	0.0000	0.9254	0.0026
60	0.0000	0.0000	0.9264	0.0026
70	0.0000	0.0000	0.9306	0.0025
80	0.0000	0.0000	0.9327	0.0025
90	0.0000	0.0000	0.9354	0.0025
100	0.0000	0.0000	0.9383	0.0024
200	0.0000	0.0000	0.9414	0.0023
500	0.0000	0.0000	0.9460	0.0023
1000	0.0000	0.0000	0.9457	0.0023
5000	0.0000	0.0000	0.9520	0.0021
10000	0.0000	0.0000	0.9493	0.0022

From this we see that once the sample size is bigger than 20, there is essentially zero probability that the interval will include negative values. However we have to have a sample size of at least 200 before the estimated coverage probability gets above 94% and stays there. **[2/2 marks]**

c) STAT6CI3 students only

The parameters μ and σ^2 of the log-normal are not the mean and variance. The mean of the distribution is

$$\mathrm{E}(X) \;=\; \mathrm{E}\left(\mathrm{e}^{Y}\right) \;=\; M_{Y}(1) \;=\; \exp\{\mu + 0.5\sigma^2\}$$

where $Y \sim \text{normal}(\mu, \sigma^2)$ and $M_{Y}(t)$ is the moment generating function of Y. The given confidence interval is for this mean. **[1 mark]**

Furthermore we notice that

$$X \sim \text{log-normal}(\mu, \sigma^2) \quad\Longleftrightarrow\quad X = \mathrm{e}^{\mu} Y \quad \text{where } Y \sim \text{log-normal}(0, \sigma^2)$$

[1 mark]

Hence we can write

$$\overline{x} - z_{\alpha/2}\frac{s_x}{\sqrt{n}} \;<\; \exp\left\{\mu + 0.5\sigma^2\right\} \quad\Longleftrightarrow\quad \mathrm{e}^{\mu}\,\overline{y} - z_{\alpha/2}\frac{\mathrm{e}^{\mu}\,s_y}{\sqrt{n}} \;<\; \exp\left\{\mu + 0.5\sigma^2\right\}$$

$$\Longleftrightarrow\quad \overline{y} - z_{\alpha/2}\frac{s_y}{\sqrt{n}} \;<\; \exp\left\{0.5\sigma^2\right\}$$

and so we need not consider the value of μ but do need to consider different values of $\sigma^2 > 0$. **[1 mark]**

In the code below I will consider $\sigma^2 \in \{0.5, 1, 2, 4\}$ which are sufficient to give a reasonable picture of how the coverage varies with σ^2. As before the true mean is guaranteed to be positive so it is also of interest to examine how often the interval contains negative values. I will only examine values of $n \leqslant 1000$ in this solution but more would be preferable.

Here is the R code for the simulation study.

```
n <- c(5*(1:6),10*(4:10),200,500,1000)
maxn <- max(n)
sigma <- c(0.5,1,2,4)
z.alpha <- qnorm(1-alpha/2)
out <- matrix(NA,ncol=5,nrow=length(n))
colnames(out) <- c("n", "P(lo<0)","se(P(lo<0))", "Cover", "se(Cover)")
results.4c <- list("sigma=0.5"=out, "sigma=1"=out, "sigma=2"=out,
                   "sigma=4"=out)

set.seed(24022019)
Zmat.norm <- matrix(rnorm(N*maxn), nrow=N)
for (j in 1:length(sigma)) {
   Ymat <- exp(sigma[j]*Zmat.norm)
   true.mean <- exp(sigma[j]^2/2)
   for (i in 1:length(n)) {
      ni <- n[i]
      ybar <- rowMeans(Ymat[,1:ni])
      sey <- apply(Ymat[,1:ni],1,sd)/sqrt(ni)
      lower <- ybar-z.alpha*sey
      upper <- ybar+z.alpha*sey
      p0 <- mean(lower<0)
      se0 <- sqrt(p0*(1-p0)/R)
      p1 <- mean(lower<true.mean & upper>true.mean)
      se1 <- sqrt(p1*(1-p1)/R)
      results.4c[[j]][i,] <- c(ni,p0,se0,p1,se1)
   }
}
```

[2 marks]

Here are the four tables

$\sigma = 0.5$

n	$\hat{\mathrm{P}}(\mathrm{lo} < 0)$	$\mathrm{se}(\hat{\mathrm{P}}(\mathrm{lo} < 0))$	Cover	se(Cover)
5	0.0018	0.0004	0.8403	0.0037
10	0.0000	0.0000	0.8889	0.0031
15	0.0000	0.0000	0.9050	0.0029
20	0.0000	0.0000	0.9151	0.0028
25	0.0000	0.0000	0.9221	0.0027
30	0.0000	0.0000	0.9260	0.0026
40	0.0000	0.0000	0.9306	0.0025
50	0.0000	0.0000	0.9337	0.0025
60	0.0000	0.0000	0.9370	0.0024
70	0.0000	0.0000	0.9379	0.0024
80	0.0000	0.0000	0.9390	0.0024
90	0.0000	0.0000	0.9399	0.0024
100	0.0000	0.0000	0.9413	0.0024
200	0.0000	0.0000	0.9454	0.0023
500	0.0000	0.0000	0.9450	0.0022
1000	0.0000	0.0000	0.9490	0.0022

$\sigma = 1$

n	$\hat{\mathrm{P}}(\mathrm{lo} < 0)$	$\mathrm{se}(\hat{\mathrm{P}}(\mathrm{lo} < 0))$	Cover	se(Cover)
5	0.1774	0.0038	0.7431	0.0044
10	0.0479	0.0021	0.8049	0.0040
15	0.0176	0.0013	0.8340	0.0037
20	0.0077	0.0009	0.8523	0.0035
25	0.0040	0.0006	0.8644	0.0034
30	0.0022	0.0005	0.8734	0.0033
40	0.0008	0.0003	0.8872	0.0032
50	0.0003	0.0002	0.8938	0.0031
60	0.0002	0.0001	0.9007	0.0030
70	0.0001	0.0001	0.9054	0.0029
80	0.0001	0.0001	0.9087	0.0029
90	0.0000	0.0001	0.9111	0.0028
100	0.0000	0.0000	0.9140	0.0028
200	0.0000	0.0000	0.9273	0.0026
500	0.0000	0.0000	0.9399	0.0024
1000	0.0000	0.0000	0.9439	0.0023

$\sigma = 2$

n	$\hat{P}(\text{lo} < 0)$	se($\hat{P}(\text{lo} < 0)$)	Cover	se(Cover)
5	0.6511	0.0048	0.4668	0.0050
10	0.5084	0.0050	0.5413	0.0050
15	0.4095	0.0049	0.5825	0.0049
20	0.3400	0.0047	0.6086	0.0049
25	0.2881	0.0045	0.6300	0.0048
30	0.2496	0.0043	0.6475	0.0048
40	0.1960	0.0040	0.6718	0.0047
50	0.1595	0.0037	0.6879	0.0046
60	0.1347	0.0034	0.7027	0.0046
70	0.1165	0.0032	0.7131	0.0045
80	0.1018	0.0030	0.7232	0.0045
90	0.0900	0.0029	0.7312	0.0044
100	0.0804	0.0027	0.7384	0.0044
200	0.0377	0.0019	0.7819	0.0041
500	0.0120	0.0011	0.8254	0.0038
1000	0.0047	0.0007	0.8536	0.0035

$\sigma = 4$

n	$\hat{P}(\text{lo} < 0)$	se($\hat{P}(\text{lo} < 0)$)	Cover	se(Cover)
5	0.9082	0.0029	0.0819	0.0027
10	0.8889	0.0031	0.1050	0.0031
15	0.8696	0.0034	0.1204	0.0033
20	0.8462	0.0036	0.1315	0.0034
25	0.8264	0.0038	0.1408	0.0035
30	0.8090	0.003	0.1486	0.0036
40	0.7766	0.0042	0.1616	0.0037
50	0.7496	0.0043	0.1709	0.0038
60	0.7269	0.0045	0.1791	0.0038
70	0.7074	0.0045	0.1861	0.0039
80	0.6917	0.0046	0.1921	0.0039
90	0.6758	0.0047	0.1968	0.0040
100	0.6624	0.0047	0.2029	0.0040
200	0.5714	0.0049	0.2380	0.0043
500	0.4527	0.0050	0.2866	0.0045
1000	0.3792	0.0049	0.3254	0.0047

For small values of σ the interval performs reasonably well but the performance of the interval gets very bad for large values of σ. If $\sigma = 2$ then even a sample of size 1000 only gives around 85% actual coverage. For the most extreme case examined ($\sigma = 4$) the coverage of the interval with a sample of size 1000 is under 33% and about 38% of the time the intervals include negative values. Large values of σ imply very heavy population skewness making the interval inappropriate. **[3 marks]**

Q. 5 STAT6CI3 students only

a) Suppose we use importance sampling from the importance density g to estimate

$$I = \int h(x)f(x)dx$$

where f is a density and $f(x) > 0 \Rightarrow g(x) > 0$. Then we sample $X_1, \ldots, X_N \overset{iid}{\sim} g$ and

$$\hat{I}_{IS} = \frac{1}{N}\sum_{i=1}^{N} \frac{h(X_i)f(X_i)}{g(X_i)}$$

The sampling variability of this estimator is then

$$\begin{aligned}
\text{Var}(\hat{I}_{IS}) &= \frac{1}{N}\text{Var}\left(\frac{h(X)f(X)}{g(X)}\right) \\
&= \frac{1}{N}\left\{\text{E}\left[\left(\frac{h(X)f(X)}{g(X)}\right)^2\right] - \left(\text{E}\left[\frac{h(X)f(X)}{g(X)}\right]\right)^2\right\} \\
&= \frac{1}{N}\left\{\int \left(\frac{h(x)f(x)}{g(x)}\right)^2 g(x)\,dx - \left(\int \left(\frac{h(x)f(x)}{g(x)}\right) g(x)\,dx\right)^2\right\} \\
&= \frac{1}{N}\left\{\int \frac{h^2(x)f^2(x)}{g(x)}\,dx - \left(\int h(x)f(x)\,dx\right)^2\right\} \\
&= \frac{1}{N}\left\{\int \frac{h^2(x)f^2(x)}{g(x)}\,dx - I^2\right\}
\end{aligned}$$

[6 marks]

b) If we take $g(x) \propto |h(x)|f(x)$ then we have that

$$g(x) = \frac{|h(x)|f(x)}{\int |h(x)|f(x)\,dx}$$

If we now plug this into the integral in $\text{Var}(\hat{I}_{IS})$ in part (a) we get

$$\begin{aligned}
\int \frac{h^2(x)f^2(x)}{g(x)}\,dx &= \int \frac{h^2(x)f^2(x)}{\left(\frac{|h(x)|f(x)}{\int |h(x)|f(x)\,dx}\right)}\,dx \\
&= \int \frac{h^2(x)f^2(x)}{|h(x)|f(x)}\,dx \int |h(x)|f(x)\,dx \\
&= \int |h(x)|f(x)\,dx\,dx \int |h(x)|f(x)\,dx\,dx \\
&= \left(\int |h(x)|f(x)\,dx\right)^2
\end{aligned}$$

Hence we see that for this particular choice of g we have

$$\mathrm{Var}(\hat{I}_{IS}) \;=\; \frac{1}{N}\left\{\left(\int |h(x)|f(x)\,dx\right)^2 - I^2\right\}$$

[6 marks]

Now all that remains is to show that for any other choice of g

$$\mathrm{Var}(\hat{I}_{IS}) \;=\; \frac{1}{N}\left\{\int \frac{h^2(x)f^2(x)}{g(x)}\,dx - I^2\right\} \;\geqslant\; \frac{1}{N}\left\{\left(\int |h(x)|f(x)\,dx\right)^2 - I^2\right\}$$

Clearly this is true if and only if

$$\int \frac{h^2(x)f^2(x)}{g(x)}\,dx \;\geqslant\; \left(\int |h(x)|f(x)\,dx\right)^2$$

Now suppose that X is a random variable with pdf g and define the random variable

$$Y \;=\; \frac{|h(X)|f(X)}{g(X)}$$

Now we can write

$$\begin{aligned}
\mathrm{Var}(Y) &= \mathrm{E}\left(Y^2\right) - \left[\mathrm{E}(Y)\right]^2 \\
&= \mathrm{E}_g\left(\frac{h^2(X)f^2(X)}{g^2(X)}\right) - \left[\mathrm{E}_g\left(\frac{|h(X)|f(X)}{g(X)}\right)\right]^2 \\
&= \int \left(\frac{h^2(x)f^2(x)}{g^2(x)}\right) g(x)\,dx - \left[\int \left(\frac{|h(X)|f(X)}{g(X)}\right) g(x)\right]^2 \\
&= \int \frac{h^2(x)f^2(x)}{g(x)}\,dx - \left(\int |h(x)|f(x)\,dx\right)^2
\end{aligned}$$

but $\mathrm{Var}(Y) \geqslant 0$ for any random variable Y and so

$$\int \frac{h^2(x)f^2(x)}{g(x)}\,dx \geqslant \left(\int |h(x)|f(x)\,dx\right)^2$$

Hence we have that for any arbitrary importance sampling function g

$$\mathrm{Var}(\hat{I}_{IS}) \;\geqslant\; \frac{1}{N}\left\{\left(\int |h(x)|f(x)\,dx\right)^2 - I^2\right\}$$

and that the right-hand side of this is the variance of $\hat{I}_{IS}$ when we have the particular $g(x) \;\propto\; |h(x)|f(x)$. This proves the assertions of Theorem 10 in my notes. **[8 marks]**