STAT 428 Statistical Computing

Exam 1

Sample Solution

1.

a) $g(y) = \frac{1}{2}$ for $y \in [0, 2]$. Then

$$f(x) \leq 2g(x), x \in [0, 2]$$

The algorithm for acceptance/rejection sample is

- First, draw sample x from Uniform[0,2]
- Second, draw sample u from Uniform[0,1]
- Accept x if

$$u \leq \frac{f(x)}{2g(x)} = f(x)$$

b)

R code:

```
f_x = function(x){ ifelse(x<=1, x, 2-x)}

myfunc1 = function(n){
      counter = 0
      Xsample = NULL
      while (counter < n){
            x = runif(1, 0, 2)
            u = runif(1, 0, 1)
            if (u <= f_x(x)){
                Xsample = c(Xsample, x)
                counter = counter + 1
            }
      }
      Xsample
}
```

c) Use the inverse method.

$$f(x) = \begin{cases} x & 0 < x \le 1 \\ 2 - x & 1 < x < 2 \end{cases}$$

Then we have

$$F(x) = \begin{cases} \frac{1}{2}x^2 & 0 < x \le 1 \\ -\frac{1}{2}(x-2)^2 + 1 & 1 < x < 2 \end{cases}$$

use inverse method

$$F^{-1}(u) = \begin{cases} \sqrt{2u} & 0 < u \le 0.5 \\ 2 - \sqrt{2 - 2u} & 0.5 < x < 1 \end{cases}$$

R code:

```
myfunc2 = function(n){
      u = runif(n, 0, 1)
      Xsample = ifelse(u<=0.5, sqrt(2*u), 2-sqrt(2-2*u))
      Xsample
}
```

#

2.

a)

R code:

```
myfunc3 = function(n,p){
    xsample = NULL
    counter = 0
    while(counter < n){
      myx = rbinom(1,1,p)
      Idx = 1
      while (myx != 1){
        myx = rbinom(1,1,p)
        Idx = Idx + 1
      }
      xsample = c(xsample, Idx1)
      counter =  counter + 1
    }
    xsample
}
```

b)

R code:
```
myfunc4 = function(n, k, p){
     xsample = NULL
     counter = 0
     while(counter < n){
        x = sum(myfunc3(k, p))
        xsample = c(xsample, x)
        counter = counter + 1
     }
     xsample
}
```

#

3.

a)
$f(x)$ is a probability density function supported on $(0, \infty)$ since

- $f(x) = 2xe^{-x^2}$ is nonnegative when $x > 0$
- $\int_0^\infty f(x)\,dx = \int_0^\infty 2xe^{-x^2}\,dx = \int_0^\infty d\left(-e^{-x^2}\right) = -e^{-x^2}|_0^\infty = 1 - 0 = 1$

b)

$$F(x) = \int_0^x f(t)\,dt = \int_0^x 2te^{-t^2}\,dt = -e^{-t^2}|_0^x = 1 - e^{-x^2}$$

c)

$$F^{-1}(u) = \sqrt{-\ln(1-u)}$$

R code:

```
myfunc5 = function(n){
  u = runif(n, 0, 1)
  Xsample = sqrt(-log(1-u))
  Xsample
}
```

#

4.

a)

R code:

```
m = 10000
Usample = runif(m)
Xsample = exp(Usample^2)
theta = mean(Xsample)
```

b)

R code:

```
std_error = sqrt(var(Xsample)/m)
```

c)

R code:

```
m = 10000
k = 10
r = M / k
N = 50
T2 = numeric(k)
estimates = rep(0, k)
fx = function(x){exp(x^2)}
for (i in 1:k){
    T2[i] = mean(fx(runif(M/k, (i-1)/k, i/k)))
}
theta = mean(T2)
```

d)

$$\int_0^1 \phi(x)\,dx = \int_0^1 \frac{e^x}{c}dx = \int_0^1 \frac{de^x}{c} = 1 \quad \Longrightarrow \quad c = e - 1$$

e)

- First, draw i.i.d sample $x_1, x_2, \ldots, x_{1000}$ from $\phi(x) = \frac{e^x}{e-1}$ using inverse CDF method
- Second, estimate $\theta = E\left(e^{x^2}\right)$ by

$$\hat{\theta} = \frac{1}{1000}\left[\frac{e^{x_1{}^2}}{\phi(x_1)} + \frac{e^{x_2{}^2}}{\phi(x_2)} + \cdots + \frac{e^{x_{1000}{}^2}}{\phi(x_{1000})}\right]$$

R code:

```
phi_x = function(x){ exp(x)/(exp(1)-1) }
fx = function(x){exp(x^2)}

myfunc6 = function(n){
  usample = runif(n)
  xsample1 = log(1+usample(exp(1)-1))
  xsample2 = fx(xsample1)/phi_x(xsample1)
  theta = mean(xsample2)
  return(theta)
}
```

#