Homework 1

Jieqi Tu

Question 1

(a)

$$\int_0^\infty x(1+x^2)^{-2}dx$$

Set $u = (1+x^2)$, then we can get: $\int_0^\infty x(1+x^2)^{-2}dx = \frac{1}{2}\int_1^\infty u^{-2}du = \frac{1}{2}$
In order to transform $\int_0^\infty$ to $\int_0^1$, set $y = \frac{1}{1+x}$.
So $dx = -\frac{1}{y^2}dy$. Then we have $\int_1^0 -\frac{1}{y^2}\frac{(1/y-1)}{(1+(1/y-1)^2)^2}dy$.

```
# set seed
set.seed(1029)

# check theoretical value of the given integral using R:
integral_a = function(x) {
  x*(1+x^{2})^{-2}
}
integrate(integral_a, lower = 0, upper = Inf)$value
```

```
## [1] 0.5
```

```
# simulate using uniform distribution
s = runif(100000)
transform = (1/s^2)*(1/s-1)/(1+(1/s-1)^2)^2
mean(transform)
```

```
## [1] 0.4983382
```

(b)

$$\int_{-\infty}^{\infty} e^{-x^2}dx$$

Because we already know the probability density function of Normal distribution: $f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}}$, we can let $\mu = 1$, $\sigma = 1/\sqrt{2}$. Then according to the property of probability density function, we have: $\int_{-\infty}^{\infty} e^{x^2}dx = \sqrt{\pi}$
To transform upper bound and lower bound, let $y = \frac{e^x}{1+e^x}$. Then we have $\int_0^1 e^{\frac{logit^2(y)}{y(1-y)}}dy$

```r
# set seed
set.seed(1029)

# check theoretical value of the given integral using R:
integral_b = function(x) {
  exp(-x^2)
}
integrate(integral_b, lower = -Inf, upper = Inf)$value
```

```
## [1] 1.772454
```

```r
# simulate using uniform distribution
s = runif(100000)
transform = exp(-(log(s/(1-s)))^2)/(s*(1-s))
mean(transform)
```

```
## [1] 1.764829
```

(c)

$$\int_2^4 lnxdx$$

$\int_2^4 lnxdx = xlnx|_2^4 - \int_2^4 xdlnx = 4ln4 - 2ln2 - 2 \approx 2.1588$. To transform the upper and lower bound, set $y = \frac{x-2}{2}$. Then we have$\int_0^1 2ln(2y+2)dy$.

```r
# set seed
set.seed(1029)

# check theoretical value of the given integral using R:
integral_c = function(x) {
  log(x)
}
integrate(integral_c, lower = 2, upper = 4)$value
```

```
## [1] 2.158883
```

```r
# simulate using uniform distribution
s = runif(1000000000)
transform = 2*log(2*s+2)
mean(transform)
```

```
## [1] 2.158887
```

(d)

$$\int_0^1 \int_0^1 e^{x+y} dxdy$$

The exact value of this integral is $\int_0^1 e^x dx \int_0^1 e^y dy = (e-1)^2 \approx 2.952492$.

```r
# set seed
set.seed(1029)

# simulate using uniform distribution
x = runif(100000)
y = runif(100000)
transform = exp(x+y)
mean(transform)
```

```
## [1] 2.950352
```

Question 2

$Cov(U, \sqrt{1-U^2} = E(U^2\sqrt{1-U^2}) - E(U^2)E(\sqrt{1-U^2})$.
$E(U^2) = \int_0^1 u^2 du = u/3|_0^1 = \frac{1}{3}$.
Let $u = sin\theta, du/d\theta = cos\theta$. Then $E(U^2\sqrt{1-U^2}) = \int_0^{\pi/2} sin^2\theta cos^2\theta d\theta = \int_0^{\pi/2} \frac{1}{4}sin^2 2\theta d\theta = \frac{1}{8}\int_0^{\pi/2} 1 - cos4\theta d\theta = \frac{\pi}{16}$
$E(\sqrt{1-U^2}) = \int_0^1 \sqrt{1-u^2}du = \int_0^{\pi/2} cos\theta dsin\theta = \int_0^{\pi/2} cos^2\theta d\theta = \int_0^{\pi/2} \frac{1+cos2\theta}{2} d\theta = \int_0^{\pi/2}(\frac{1}{2}\theta + \frac{1}{4}sin2\theta)d\theta = \frac{\pi}{4}$
$Cov(U^2, \sqrt{1-U^2}) = \frac{\pi}{16} - \frac{1}{3} \times \frac{\pi}{4} \approx -0.06544985$

```r
# set seed
set.seed(1029)

# simulate
u = runif(1000000)
y = u^2
z = sqrt(1 - y)
cov(y, z)
```

```
## [1] -0.06543874
```

Question 3

```r
# set seed
set.seed(1029)

# simulate
sim_n = 10000
n_value = c()
for(i in 1:sim_n) {
  count = 0
  sum = 0
  while(sum <= 1) {
    u = runif(1)
    sum = sum + u
    count = count + 1
  }
  n_value[i] = count
```

```
}

mean(n_value)
```

```
## [1] 2.7068
```

Question 4

```
# set seed
set.seed(1029)

# simulate
n_sim = 100000
n_value = c()
for (i in 1:n_sim) {
    count = 0
    prod = 1
  while(prod >=exp(-3)) {
    u = runif(1)
    prod = prod * u
    count = count + 1
  }
  n_value[i] = count

}
mean(n_value)
```

```
## [1] 4.00038
```

Question 5

```
library(extraDistr)
```

```
##
## Attaching package: 'extraDistr'

## The following object is masked from 'package:purrr':
##
##     rdunif
```

```
set.seed(1029)
n_sim = 100000
n_list = numeric(n_sim) # to store all the values of the number of experiments
for (i in 1:n_sim) {
  d_list = c()
  statement = F
  n_value = 0
```

```
  while (statement == F) {
    sum = 0
    sum12 = sum(rdunif(2, 1, 6))
    n_value = n_value + 1
    d_list = c(d_list, sum12)
    for (j in 2:12) {
      # whether or not this outcome shows up in our experiments
      within = as.numeric(j %in% d_list)
      sum = sum + within
    }
     # If one outcome is not present in the experiments, we will start a new experiment
    statement = (sum == 11)
  }
  n_list[i] = n_value
  }
round(mean(n_list))
```

```
## [1] 61
```

Question 6

(a)

Since $p_j = \frac{(j-1)!}{(j-r)!(r-1)!}p^r(1-p)^{j-r}$, $p_{j+1} = \frac{(j)!}{(j+1-r)!(r-1)!}p^r(1-p)^{j+1-r}$, we have $\frac{p_{j+1}}{p_j} = j(1-p)/(j+1-r)$

(b)

The algorithm to generate a negative binomial variable:

- Generate a random variable U, which follows uniform distribution. $U \sim Uniform(0,1)$. In this case, r means the number of success in a total number of j trials. However, the last trial is defined to be successful.
- Set $c = 1-p$, $j = r$, $pr = p^r$, and $F = pr$.
- If $U < F$, stop and report j.
- Then update: $pr = \frac{c*pr}{j+1-r}pr$, $F = F + pr$, $j = j+1$.
- Go to step 3 and repeat.

(c)

Theoretical value calculation:$E(X) = \frac{rp}{1-p} = \frac{5}{0.6} = 12.5$, $Var(X) = \frac{r(1-p)}{p^2} = \frac{5*0.6}{0.16} = 18.75$

```
# set seed
set.seed(1029)

# simulate
n_sim = 1000000
p = 0.4
r = 5
c = 1 - p
n_value = c()
for (i in 1:n_sim) {
  j = r
```

```
  pr = p^r
  u = runif(1, 0, 1)
  F_store = pr
  while (u>F_store) {
    pr = c*pr*j/(j+1-r)
    F_store = F_store + pr
    j = j + 1
  }
  n_value[i] = j
}

mean(n_value)
```

```
## [1] 12.50136
```

```
var(n_value)
```

```
## [1] 18.73096
```

Question 7

Brute force to solve $E(|Z|)$:
$E(|Z|) = 2\int_0^\infty z\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}z^2}dz =$. Let $u = e^{-\frac{1}{2}z^2}$, then we have $du/dz = -ze^{-\frac{1}{2}z^2}$. So $E(|Z|) = -\int_1^0 \frac{2}{\sqrt{2\pi}}du = \frac{2}{\sqrt{2\pi}} \approx 0.798$

```
set.seed(1029)

# create a large number of standard normally distributed variables
z = rnorm(10000000, 0, 1)
abs_z = abs(z)

# check the average
mean(abs_z)
```

```
## [1] 0.7980856
```

Question 8

$E(X) = \int_0^1 x\frac{e^x}{e-1}dx = \frac{1}{e-1}e^x(x-1)|_0^1 \approx 0.581977.$
$Var(X) = E(X^2) - (E(X))^2 = \frac{1}{e-1}\int_0^1 x^2e^xdx - (\frac{1}{e-1})^2 = \frac{e-2}{e-1} - (\frac{1}{e-1})^2 \approx 0.079326$ Simulation algorithm:
Let $F(X)$ be the CDF for x, so $F(X) = \frac{1}{e-1}\int_0^x e^tdt = \frac{1}{e-1}(e^x - 1)$.
Let $U = F(X)$, then $U \sim Uniform(0,1)$. So we have $x = log[(e-1)u+1]$.

```
# set.seed
set.seed(1029)

# simulate
n_sim = 100000
x_value = c()
```

```
for(i in 1:n_sim) {
  # create the CDF for X
  u = runif(1, 0, 1)
  # express X using CDF
  x = log((exp(1)-1)*u+1)
  x_value[i] = x
}

mean(x_value)
```

```
## [1] 0.5820019
```

```
var(x_value)
```

```
## [1] 0.07966918
```

Question 9

$f(x) = \frac{dF(x)}{dx} = x + 1/2$, so $E(X) = \int_0^1 xf(x)dx = \frac{7}{12} \approx 0.5833$. Similarly, we have $E(X^2) = \int_0^1 x^2 f(x)dx = (\frac{x^4}{4} + \frac{x^3}{6})|_0^1 = \frac{5}{12}$.
Therefore, $Var(X) = E(X^2) - (E(X))^2 = \frac{5}{12} - \frac{49}{144} \approx 0.07639$. Set $V = F(X) = \frac{x^2+x}{2}$, then we have $X = -\frac{1}{2} + \frac{1}{2}\sqrt{1+8v}$ as $0 \leqslant x \leqslant 1$.

```
# set seed
set.seed(1029)

# simulate
n_sim = 100000
v = runif(n_sim, 0, 1)
x = -0.5+0.5*sqrt(1+8*v)
mean(x)
```

```
## [1] 0.5833679
```

```
var(x)
```

```
## [1] 0.07671202
```

Question 10

$X \sim Exp(1)$, so we have $f(x) = e^{-x}$. Since $P(E|F) = \frac{P(EF)}{P(F)}$, $f(X|X < 0.5) = \frac{f(X)}{P(X<0.5)}$.
Therefore, we have $f(X|X < 0.5) = \frac{e^{-x}}{1-e^{-0.5}}(0 < x < 0.5)$.
$E(X|X < 0.5) = \frac{1}{1-e^{-0.5}} \int_0^{0.5} xe^{-x} = \frac{1-(1.5)e^{-0.5}}{1-e^{-0.5}} \approx 0.2293$

```
# set seed
set.seed(1029)
```

```r
# simulate
n_sim = 100000000
x = rexp(n_sim, 1)
conditional_x = x[x<0.5]
mean(conditional_x)
```

```
## [1] 0.2292634
```

Question 11

Since it would be hard to find the root of the CDF, it is better to use acceptance-rejection method in this case. Let $g(X) = 1$, then we have $\frac{f(x)}{g(x)} = 30(x^2 - 2x^3 + x^4) = h(x)$. Set $h'(x) = 60x(x-1)(2x-1) = 0$, the roots are $x_1 = 0, x_2 = 1, x_3 = 0.5$. Take the second derivative, when $x = 0.5$, $h''(x) < 0$, so $Max(h(x)) = h(0.5) = 15/8 = 1.875$. So, we set $c = 1.875$. Therefore, $\frac{f(x)}{cg(x)} = 16(x^2 - 2x^3 + x^4)$.
Algorithm:

- Step 1: Generate random variable $U \sim Uniform(0, 1)$ and $Y \sim Uniform(0, 1)$.
- Step 2: If $U \leqslant 16(Y^2 - 2Y^3 + Y^4)$, set $X = Y$. Otherwise, go back to step 1.

```r
set.seed(1029)

# simulate
n_sim = 1000000
x_value = c() # empty set to store accepted Y
for (i in 1:n_sim) {
  u = runif(1) # Uniform distribution
  y = runif(1) # Y~g(Y)
  if(u<=16*(y^2-2*y^3+y^4)) {
    x_value[i] = y
  }
  else {x_value[i]=NA}
}
mean(x_value, na.rm = T)
```

```
## [1] 0.5002983
```

```r
var(x_value, na.rm = T)
```

```
## [1] 0.03575381
```

Question 12

With replacement

```r
set.seed(1029)
n_sim = 100000
n = 100
compare = numeric(n_sim)
for (i in 1:n_sim) {
```

```
  sample_i = sample(seq(1,n), n, replace = T, prob = rep(1/n,n))
  num = seq(1,n)
  compare[i] = sum(num == sample_i)
}
sum(compare!=0)/n_sim
```

```
## [1] 0.63308
```

Without replacement

```
set.seed(1029)
n_sim = 100000
n = 100
compare = numeric(n_sim)
for (i in 1:n_sim) {
  sample_i = sample(seq(1, n), n, replace = F, prob = rep(1/n, n))
  num = seq(1,n)
  compare[i] = sum(num == sample_i)
}
sum(compare!=0)/n_sim
```

```
## [1] 0.63003
```