Weining Bao weiningb@usc.edu 8287723653

Assignment 5

1. Let X and Y be geometrically distributed random variables with $E[X] = E[Y] = \frac{1}{p} = 20$; and $Z = \min\{X,Y\}$. Simulate X and Y to find the distribution of Z. Compare to analytical results.

2. Let $\{X_i\}$ be a set of iid exponentially distributed random variables with $E[X_i] = \frac{1}{\lambda} = 10$; and let $S_n = \frac{1}{n}\sum_{i=1}^{n} X_i$. By means of a simulation experiment estimate $E[S_n]$ and $VAR[S_n]$; and plot a histogram approximation to the densities $f_{S_n}(x)$ for n=1,…,5. Compare to analytical results.

3. (Based on notes section 5.4.3) Using the rejection method, generate samples of a Normal (i.e. Gaussian(0,1)) random variable Z; generate 5000 samples and compute the sample mean and variance.

Matlab Simulation

with code at last

1.

Let X and Y be geometrically distributed random variables with $E[X] = E[Y] = \frac{1}{p} = 20$; and $Z = \min\{X,Y\}$. Simulate X and Y to find the distribution of Z. Compare to analytical results.

X and Y both follow discrete geometric distribution $P(X = k) = p(1-p)^{k-1} \quad 1 \le k < \infty$,

where p = 1/20. So $Z = \min\{X,Y\}$ takes value k when both X and Y take k or, either X or Y takes k while the other variable takes value larger than k. So probability distribution of Z

$$P(Z = k) = P(X = k) * P(Y = k) + P(X = k) * \sum_{n=k+1}^{\infty} P(Y = n) + P(Y = k) * \sum_{n=k+1}^{\infty} P(X = n)$$

$$= p^2(1-p)^{2k-2} + 2p(1-p)^{k-1}\sum_{n=k+1}^{\infty} p(1-p)^{n-1} = p(2-p)(1-p)^{2k-2} \quad k = 1,2,...$$

So $E[Z] = \frac{1}{2p-p^2} = \frac{400}{39} \approx 10.26$.

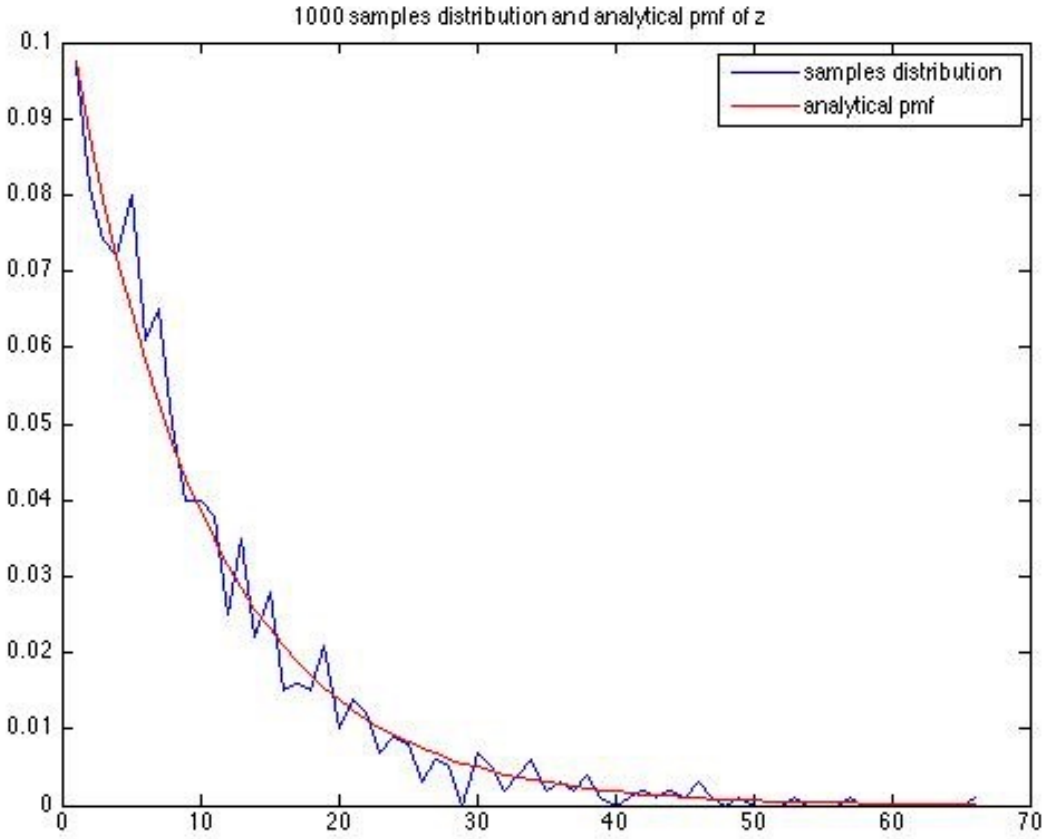
1000, 10000 and 100000 samples of Z are generated for distribution estimate. To generate a sample following geometric distribution, serial random numbers in the open range of (0,1) are generated till the random number is smaller than p. Normalized histograms of z's samples are plotted and compared with the analytic distribution. The results follow:

Prob 1:

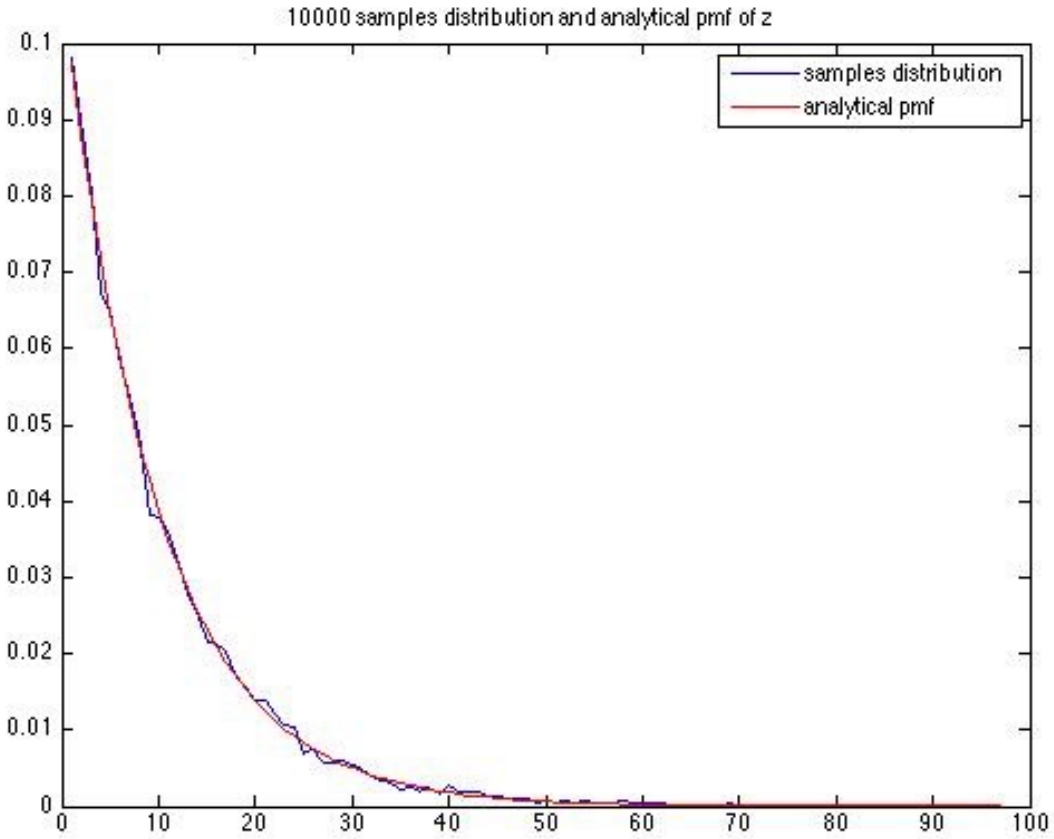
Generating 1000 samples for z:
Mean: 9.866000. Variance: 83.771816.

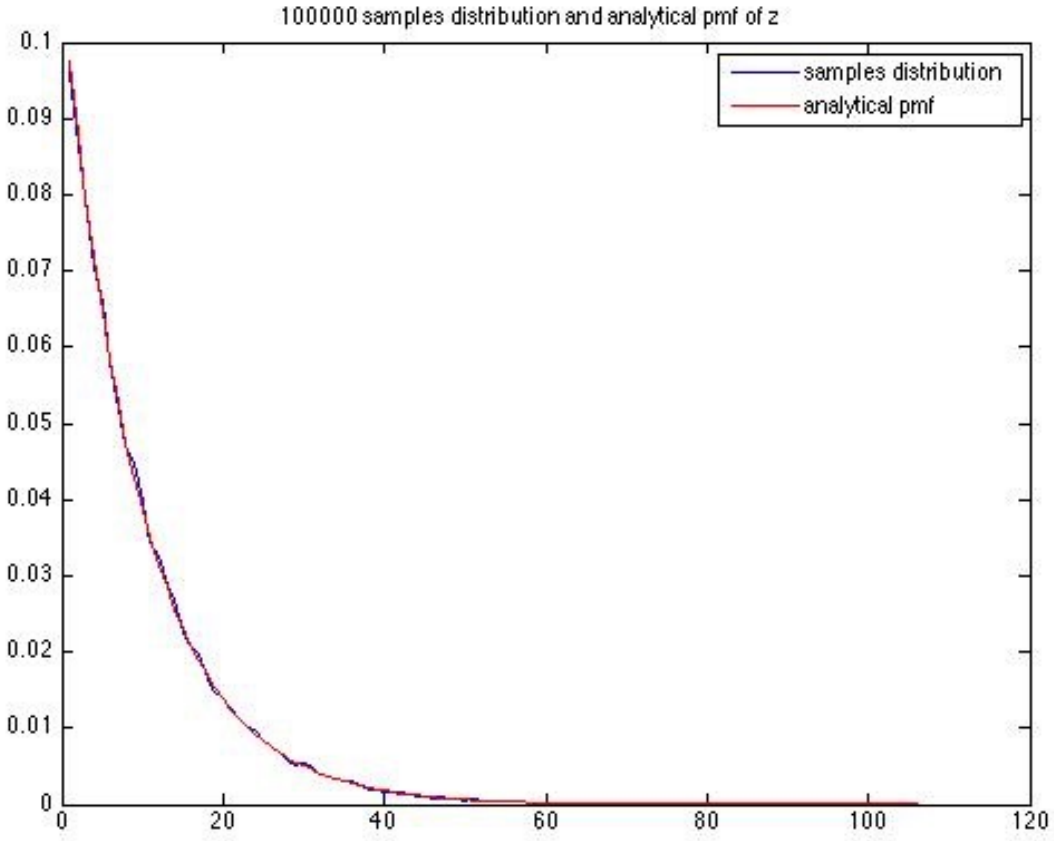
Generating 10000 samples for z:
Mean: 10.385000. Variance: 100.277003.

Generating 100000 samples for z:
Mean: 10.277310. Variance: 93.976069.



From the results above and below, it can be found that sample means get close to the true mean with the increase of sample size. Sample distributions also get smoother and fit the analytical probability mass function better with the increase of sample size.


10000 samples distribution and analytical pmf of z
samples distribution
analytical pmf
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
0
10
20
30
40
50
60
70
80
90
100


100000 samples distribution and analytical pmf of z
samples distribution
analytical pmf
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
0
20
40
60
80
100
120

2.

Let $\{X_i\}$ be a set of iid exponentially distributed random variables with $E[X_i] = \frac{1}{\lambda} = 10$; and let $S_n = \frac{1}{n}\sum_{i=1}^{n} X_i$. By means of a simulation experiment estimate $E[S_n]$ and $VAR[S_n]$; and plot a histogram approximation to the densities $f_{S_n}(x)$ for n=1,…,5. Compare to analytical results.

X_i follows continuous exponential distribution, so $F_{X_i}(x) = 1 - e^{-\lambda x}, \quad f_{X_i}(x) = \lambda e^{-\lambda x}, \ x \geq 0.$

$E[X_i] = \frac{1}{\lambda} = 10, VAR[X_i] = \frac{1}{\lambda^2} = 100.$ $S_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ and X_i are iid, so

$E[S_n] = E[\frac{1}{n}\sum_{i=1}^{n} X_i] = E[X_i], VAR[S_n] = VAR[\frac{1}{n}\sum_{i=1}^{n} X_i] = \frac{1}{n}VAR[X_i]$.

1000, 10000 and 100000 samples of Sn with n = 1,2,3,4,5 are generated for mean and variance estimate. Exponential random number generator exprnd() is used to generate samples of exponential random variable. With the increase of n, the distribution of Sn approaches normal distribution. So normal distribution with mean = 10, variance = 100/5 = 20 is also plotted to compare with the estimating normalized histograms. Here are the results:

Prob 2:

Generating 1000 samples for s1:
Mean: 10.089153. Variance: 112.013994.

Generating 1000 samples for s2:
Mean: 10.122964. Variance: 50.122447.

Generating 1000 samples for s3:
Mean: 9.845969. Variance: 34.499387.

Generating 1000 samples for s4:
Mean: 10.085208. Variance: 24.900049.

Generating 1000 samples for s5:
Mean: 9.854659. Variance: 20.332083.

Generating 10000 samples for s1:
Mean: 10.006345. Variance: 100.038342.

Generating 10000 samples for s2:
Mean: 10.066975. Variance: 51.136349.

Generating 10000 samples for s3:
Mean: 10.023550. Variance: 32.826035.

Generating 10000 samples for s4:

Mean: 9.924639. Variance: 24.225986.

Generating 10000 samples for s5:
Mean: 9.975474. Variance: 19.940960.

Generating 100000 samples for s1:
Mean: 10.034448. Variance: 101.211379.

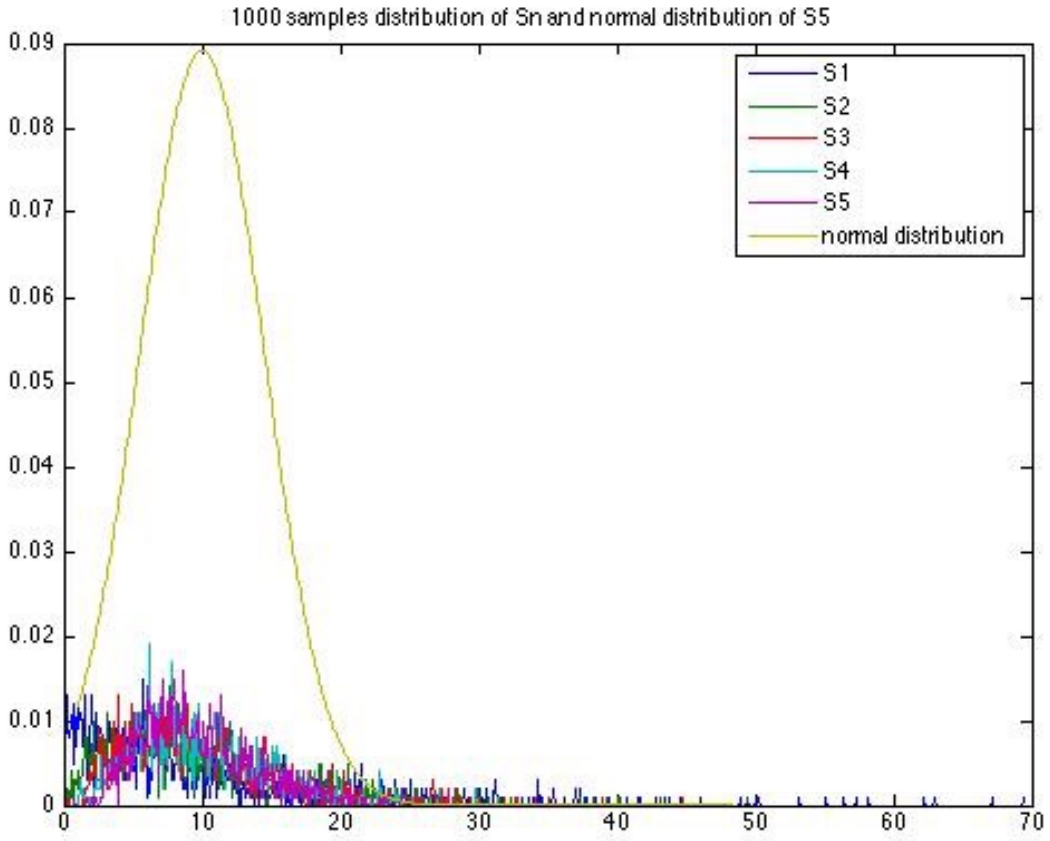
Generating 100000 samples for s2:
Mean: 10.022400. Variance: 50.246313.

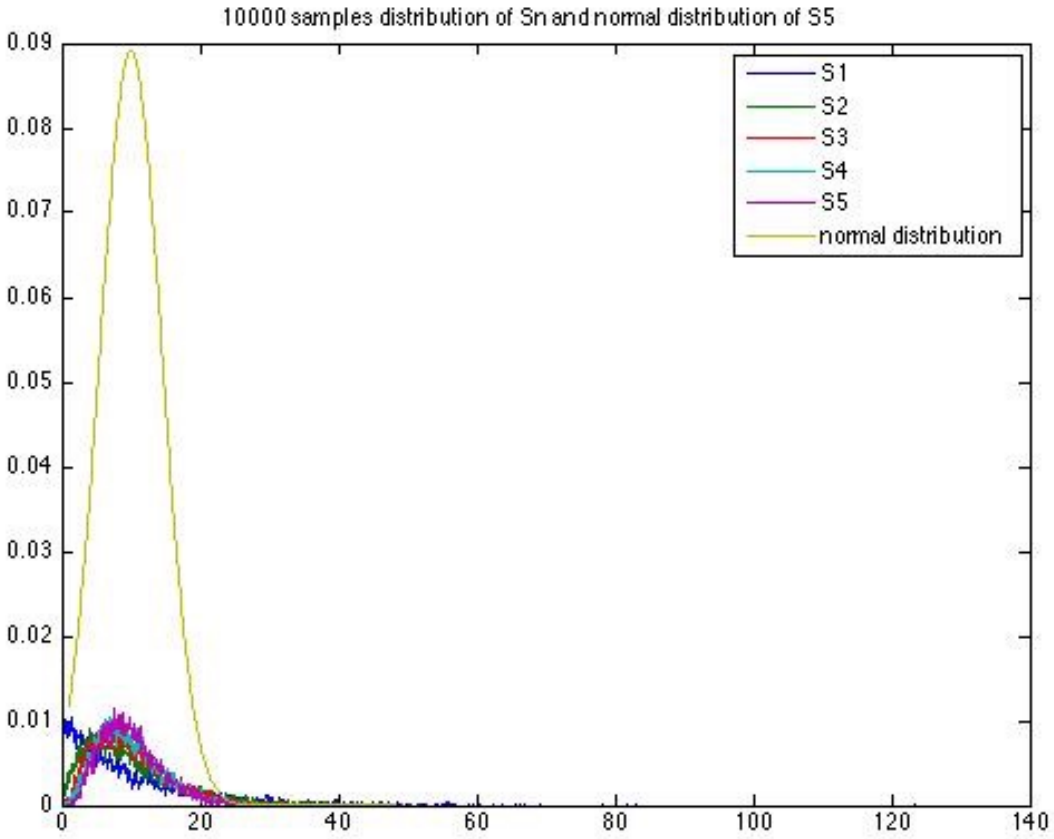
Generating 100000 samples for s3:
Mean: 10.000198. Variance: 33.354362.

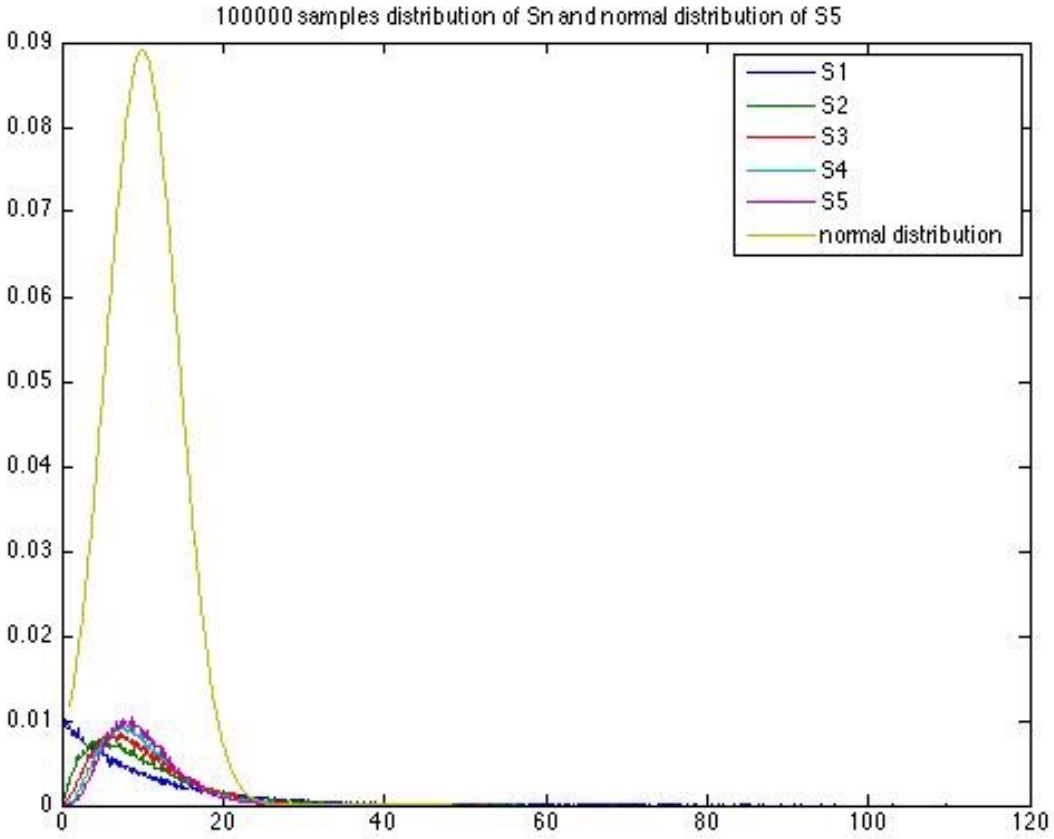
Generating 100000 samples for s4:
Mean: 9.976863. Variance: 25.149128.

Generating 100000 samples for s5:
Mean: 10.028349. Variance: 20.074923.

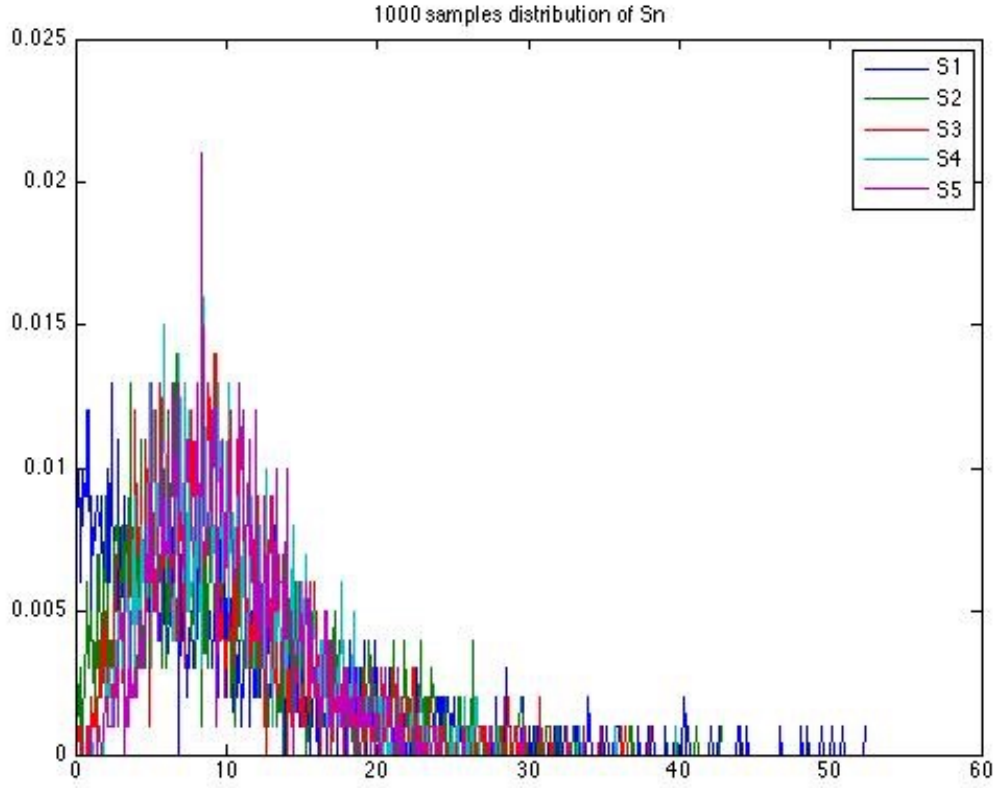
The means and variances are all close to the analytical result $\frac{1}{\lambda} = 10$ and $\frac{1}{n\lambda^2} = \frac{100}{n}$. It can also be found that sample means and variances get close to the true values with the increase of sample size.

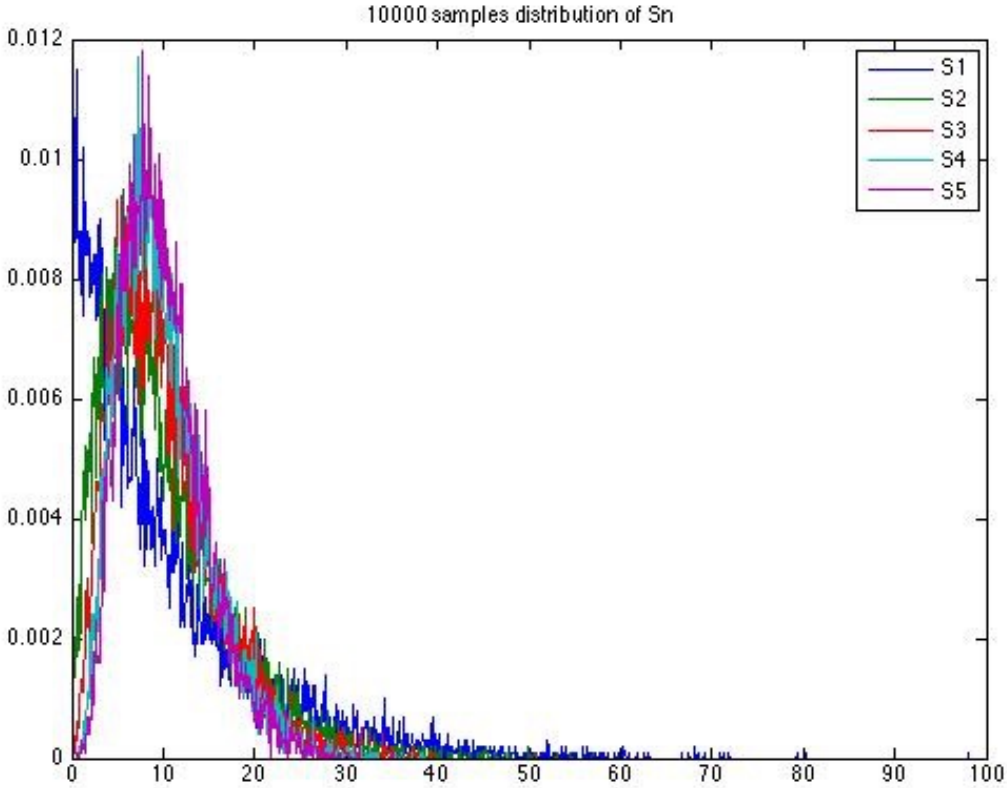


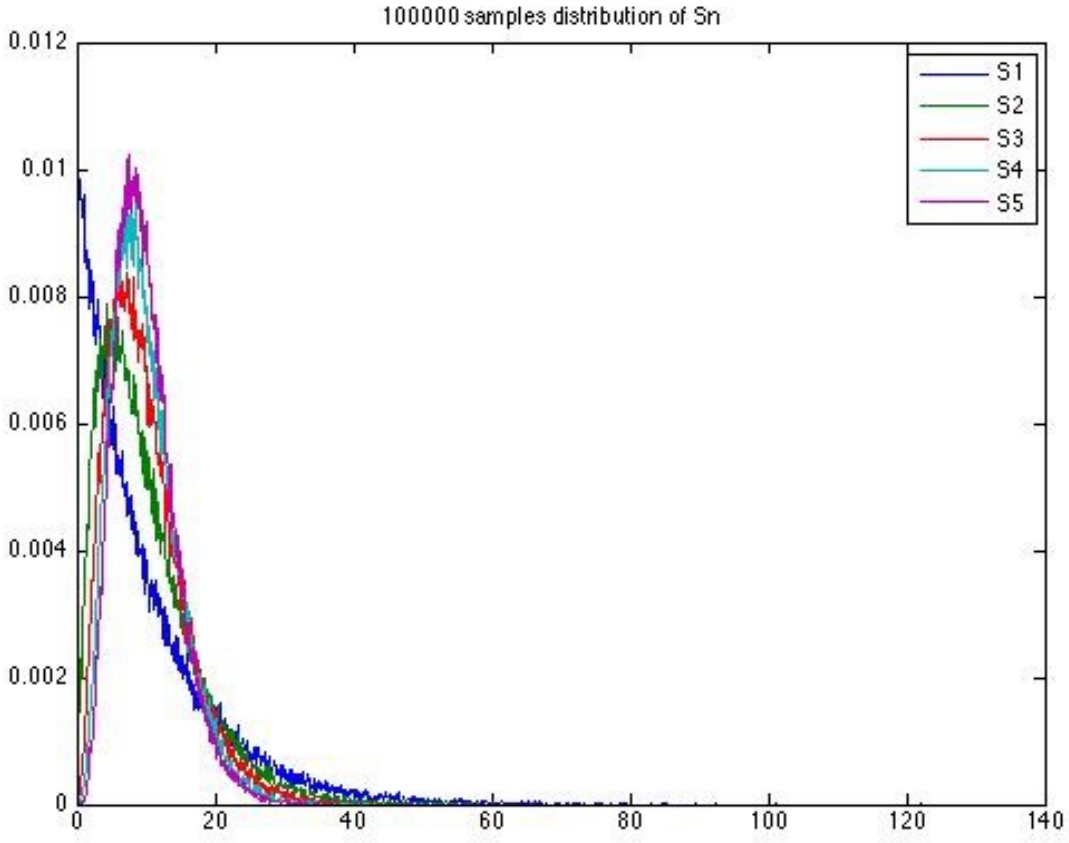

10000 samples distribution of Sn and normal distribution of S5
S1
S2
S3
S4
S5
normal distribution
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
0
20
40
60
80
100
120
140


100000 samples distribution of Sn and normal distribution of S5
S1
S2
S3
S4
S5
normal distribution
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
0
20
40
60
80
100
120

Sample distribution of S5 has the same shape with normal distribution. To see the sample distributions clearly, normal distribution with n = 5 is removed in the following plots:







Distribution curves get smoother and more normal-distribution-like shape with the increase of sample size.

3.

(Based on notes section 5.4.3) Using the rejection method, generate samples of a Normal (i.e. Gaussian(0,1)) random variable Z; generate 5000 samples and compute the sample mean and variance.

Rejection Method:

We want to generate an RV, X with pdf $f_X(x)$. We have an (efficient) method to generate an RV Y with pdf $g_Y(x)$ which is defined over the same range as X.

1. Find (smallest) c, such that $\frac{f_X(x)}{g_Y(x)} \le c$ over the range of interest of x.

2. Generate $Y \sim g_Y$, generate $U \sim \text{Uniform}(0,1)$

3. If $U \le \frac{f_X(Y)}{c g_Y(Y)}$ then set $X = Y$ otherwise return to step 2.

So use the rejection method to generate samples of a normal RV Z ~ N(0, 1):
Z is symmetrical over the positive and negative half planes, so we can consider just

the RV $X = \text{abs}(Z)$ (If we want the complete range of values for Z, we can generate an additional RV, $U \sim \text{Uniform}(0,1)$ and set

$$Z = \begin{cases} X & \text{if } U \le 0.5 \\ -X & \text{if } U > 0.5 \text{ if } U < 0.5 \end{cases}$$

The pdf of X is given by:

$$f_X(x) = \frac{2}{\sqrt{2\pi}} e^{-x^2/2} \qquad 0 \le x < \infty$$

Consider: $g_Y(x) = e^{-x}$

Then: $$\frac{f_X(x)}{g_Y(x)} = \frac{2}{\sqrt{2\pi}} e^{x-x^2/2} \qquad 0 \le x < \infty$$

The maximum of this occurs for that value of x that maximizes $x - \frac{x^2}{2}$ which occurs at $x = 1$; for which value we have:

$$c = \frac{f_X(1)}{g_Y(1)} = \frac{2}{\sqrt{2\pi}} e^{1/2} = \sqrt{\frac{2e}{\pi}}$$

Substituting we have:

$$\frac{f_X(x)}{c g_Y(x)} = \sqrt{\frac{\pi}{2e}} \left(\frac{2}{\sqrt{2\pi}} \right) e^{x-x^2/2} = e^{-(1-2x+x^2)/2} = e^{-(1-x)^2/2}$$

So the procedure to generate X (as the absolute value of a Gaussian(0,1) RV):

1. Generate $Y \sim \text{Exponential}(1); U \sim \text{Uniform}(0,1)$

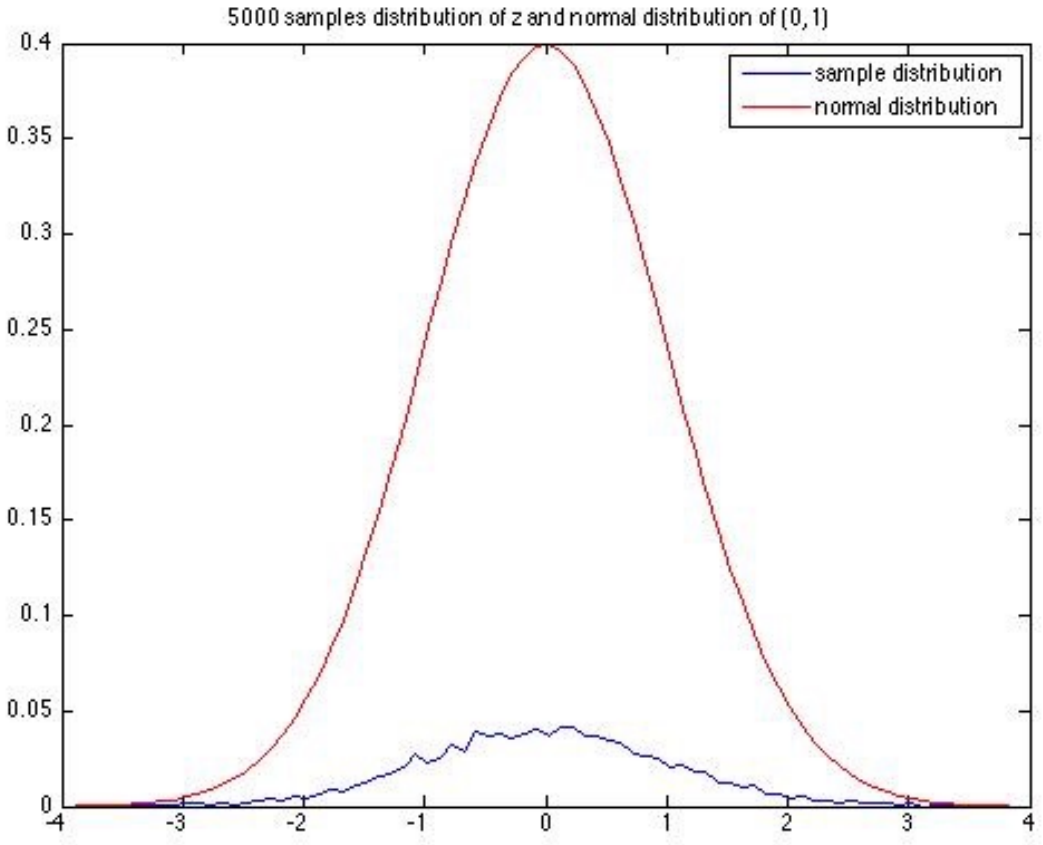
2. If $U \le e^{-(1-Y)^2/2}$ then set $X = Y$ otherwise $\rightarrow$Step 1.

5000 samples of X which are all larger than 0 are generated first then converted to Z in the range of $(-\infty, \infty)$. Exponential distributed RV is generated with exprnd() and uniform with rand().

Normal distribution of N(0,1) is also plotted to compare with the histogram of all 5000 samples of Z. Here are the results:

Prob 3:

Generating 5000 samples for z:
Mean: -0.008530. Variance: 1.054005.



The sample mean and variance are quite close to the analytical value 0 and 1. The histogram also has the same shape with the analytical normal distribution curve of N(0,1).

Code:

geodrv.m:

```
function g = geodrv(p) % generate a geometric RV sample with p
   g = 1;
   while rand() > p
      g = g + 1;
   end
end
```

assign5.m:

```
% prob 1
fprintf('\nProb 1: \n\n');
p = 1 / 20; % p for the geometrical distribution
for count = 3:5 % test 1000, 10000, 100000 Z samples
   num = 10^count; % number of samples
   zs = zeros(num, 1); % sample storage
   fprintf('Generating %d samples for z:\n', num);
   for i = 1:num  % generate z = min(x,y) samples
      x = geodrv(p);
      y = geodrv(p);
      zs(i) = min(x, y);
   end
   fprintf('Mean: %f. Variance: %f.\n\n', mean(zs), var(zs));
   b = max(zs); % find the max value of z
   z = 1:b; % divide the bins into 1,2...b
   d = p * (2 - p) * ((1 - p).^(2.*z-2)); % analytical pmf
   n = hist(zs, z); % generate the histogram
   figure;
   plot(z, n./num); hold on;
   plot(z, d, 'r'); hold off;
   title([num2str(num), ' samples distribution and analytical pmf of z']);
   legend('samples distribution', 'analytical pmf');
end

% prob 2
fprintf('\nProb 2: \n\n');
m = 10; % 1 / lamda or mean for the exponential distribution
for count = 3:5 % test 1000, 10000, 100000 Sn samples
   num = 10^count; % number of samples
   sn = zeros(num, 1); % sample storage
   mb = 0; % max number of bins for histogram
   figure;
   for i = 1:5 % n = 1,...,5
      fprintf('Generating %d samples for s%d:\n', num, i);
      for j = 1:num
         c = 0; % number of exp rv
         s = 0; % sn
         while c < i % till i exp rv are generated
            s = s + exprnd(m);
            c = c + 1;
         end
         sn(j) = s / i;
      end
      fprintf('Mean: %f. Variance: %f.\n\n', mean(sn), var(sn));
      b = max(sn); % find the max value of sn
      if b > mb % update the max number of bins for histogram
         mb = b;
```

```
        end
        ss = 0:0.1:b; % divide the bins into 0,0.1,0.2...b
        n = hist(sn, ss); % generate the histogram
        plot(ss, n./num); hold all;
    end
    norm = normpdf(1:0.1:mb, m, sqrt(m^2/5)); % the normal distribution
    plot(1:0.1:mb, norm); hold off;
    title([num2str(num), ' samples distribution of Sn and normal distribution of S5']);
    legend('S1', 'S2', 'S3', 'S4', 'S5', 'normal distribution');
end

% prob 3
fprintf('\nProb 3: \n\n');
m = 1; % 1 / lamda or mean for the exponential distribution
num = 5000; % number of samples from normal distribution (0,1)
xs = zeros(num, 1); % sample storage
fprintf('Generating %d samples for z:\n', num);
count = 0; % number of samples generated
while count < num % rejection method
    u = rand(); % uniform rv from (0,1)
    y = exprnd(m);
    if u <= exp(-(1-y)^2/2);
        count = count + 1;
        xs(count) = y;
    end
end
b = max(xs); % find the max value of xs
for i = 1:num % expand the range of z to both negative and positive
    u = rand();
    if u > 0.5
        xs(i) = -xs(i);
    end
end
fprintf('Mean: %f. Variance: %f.\n\n', mean(xs), var(xs));
z = -b:0.1:b; % divide the bins into 0,0.1,0.2...b
n = hist(xs, z); % generate the histogram
plot(z, n./num); hold on;
norm = normpdf(z, 0, 1); % the normal distribution
plot(z, norm, 'r'); hold off;
title([num2str(num), ' samples distribution of z and normal distribution of (0,1)']);
legend('sample distribution', 'normal distribution');
```