Project 7: Generating Random Variables

EE 511 – Section Thursday 9 am

Name: Junquan Yu

Student ID #: 3372029142

1. Problem Statement

If Y is uniformly distributed over (0,1); then $X=-\ln(1-Y)$ or $X=-\ln(Y)$ will be exponentially distributed with parameter λ.

a) Why is this so?

b) Use this to generate 1000 samples X_i with $\lambda=1$ and plot the empirical distribution for these samples and compare to an exponentially distributed RV.

c) Define the random variable $S_n=\sum_{i=1}^{n} X_i$. The pdf for S_n can be shown to be:

$$f_{S_n}(x)=\frac{(\lambda x)^{n-1}}{(n-1)!}\lambda e^{-\lambda x}$$

And the CDF can be expressed as:

$$F_{S_n}(x)=1-\sum_{j=0}^{n-1}\frac{1}{j!}(\lambda x)^j e^{-\lambda x} \quad \text{(defining } 0!=1 \text{ for convenience)}$$

This distribution is the Erlang-*n* distribution. So, we can generate an RV that has an Erlang-*n* distribution by summing *n* samples of an RV which is exponentially distributed. Use this approach to generate 1000 samples from an Erlang-3 distribution.

d) A more (computationally) efficient way to generate samples for S_n uses:

$$S_n=\frac{-\ln\left(\prod_{i=1}^{n} U_i\right)}{\lambda}$$

Where $U_i \sim Uniform(0,1)$. (Why does this work?) Use this approach to generate a series of samples for S_n (for *n*=3), compute mean and sample variance, and compare to part c).

e) Use the rejection method to generate samples from an Erlang-3 distribution directly. Compute mean and variance and compare to c) and d). (See notes for Lecture 10/11 on rejection method for CRVs))

f) Compare the time taken to generate *N* samples of Erlang-3 distributions using the three approaches. (Use a value of *N* that causes the slowest method to take ~30 seconds.)

2. Theoretical Exploration or Analysis

For the problem a):

Because Y is uniformly distributed, the pdf of Y is:

$$f_Y(y) = \begin{cases} 1 & 0 \le y \le 1 \\ 0 & \text{others} \end{cases}$$

Because Y= e^{-x} , we can get:

$$f_X(x) = |\frac{dy}{dx}| * f_Y(e^{-x}) = e^{-x} \quad (x > 0)$$

The pdf of X is:

$$f_X(x) = \begin{cases} e^{-x} & x > 0 \\ 0 & x \le 0 \end{cases}$$

Therefore, the random variable X is exponentially distributed with the $\lambda = 1$

For the problem d):

Assuming that X_1, X_2 and X_3 are three exponentially distributed random variables with $\lambda = 1$ and U_1, U_2 and U_3 are three uniformly distributed random variables

From $S_n = \sum_{i=1}^{3} X_i$, we can get that:

$$S_n = -\ln U_1 - \ln U_2 - \ln U_3$$

According to the logarithmic property, we can get that:

$$S_n = -\ln(\prod_{i=1}^{3} U_i) \quad (\lambda = 1)$$

3. Simulation Methodology

For the problem a)

I answered this question in the above Theoretical Exploration or Analysis part.

For the problem b)

First, I generated 1000 uniformly distributed samples Y_i with the function rand( ), then I got 1000 according samples X_i with the given relation $X_i = -\ln Y_i$. I divided the interval (0,5) to several equal-sized intervals and checked every sample X_i. If the X_i I checked is less than the up bound of certain interval, then Fx counts for that interval plus one. After traversing all the number, I can get the Fx for these data, which is Fx=Fx counts/1000 and sketch the empirical distribution with the plot( ) function. In addition to this, I also sketched the cdf of the exponentially distributed RV with $\lambda = 1$ in the same figure to compare it with the empirical distribution.

For the problem c)

First I generated 3 uniformly distributed random variable samples Y_1, Y_2 and Y_3, then I transferred them into three random variables X_1, X_2 and X_3 with the relation $X_i = -\ln Y_i$. After that, I added these three samples to get one sample for the random variable S_n. I repeated this process for 1000 times to get 1000 samples.

For the problem d)

First I generated 3 uniformly distributed random variable samples Y_1, Y_2 and Y_3, then I used $S_n = -\ln(\prod_{i=1}^{3} U_i)$ $(\lambda = 1)$ to get one sample for the random variable S_n. I repeated this process for 1000 times to get 1000 samples.

For the problem e)

To generate $S_n \sim f_{S_n}(x) = x^2 * e^{-x} / 2$ (n=3)

Using $Y \sim U(0,10)$, i.e. $g_Y(y) = \begin{cases} \frac{1}{10} & (0 < y < 10) \\ 0 & \text{others} \end{cases}$

Set $h(x) = \frac{f_{S_n}(x)}{g_Y(x)} = 5x^2 * e^{-x}$ (0<x<10)

Differentiate $\frac{dh(x)}{dx} = 10x*e^{-x} - 5x^2 * e^{-x}$

And the maximum is found at x=2

So, we set $c = h(2) = 20e^{-2}$

Finally, we have following procedure:

1. Generate $Y \sim U(0,10)$ and $U \sim U(0,1)$

2. If $U \le \frac{f_{S_n}(y)}{c * g_Y(y)} = \frac{y^2 * e^{-y}}{4 * e^{-2}}$, then set $S_n = Y$ otherwise return to step 1

For the problem f)

I used the function tic and toc to track the time it takes to generate N samples of Erlang distribution using three approaches.

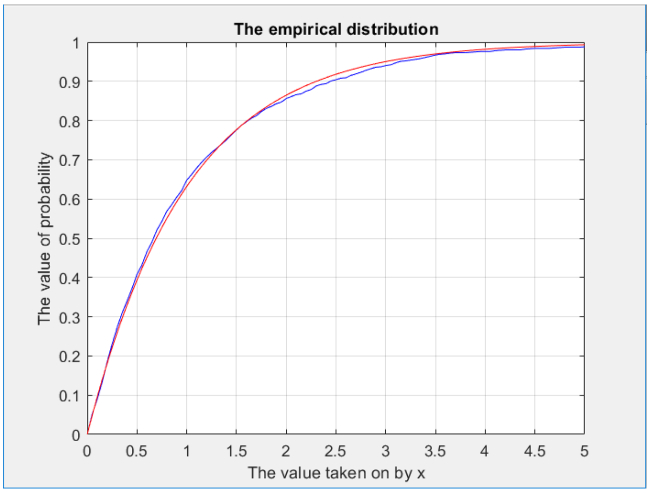
4. Experiments and Results

For the problem a)

I answered this question in the above Theoretical Exploration or Analysis part.

For the problem b)

I generated 1000 samples X_i with $\lambda = 1$ and plot their empirical distribution, and then I compared it with the cdf of the exponentially distributed RV:



(The blue line represents the empirical distribution)

(The red line represents the cdf of the exponentially distributed RV)

It can be seen from above diagram that the empirical distribution of 1000 samples is very close the cdf of the exponentially distributed RV, which matches the theoretical analysis very well.

For the problem c)

The first approach:

I set n=3 in order to compare the result of this approach with the result from the other approaches. Following the instruction in the document, I ran the program to generate 1000 samples from the Erlang-3 distribution. In addition to this, I also calculated the mean and variance of these samples. After that, I repeated this process for five times to get five sets of results. The results are as follows:

	1	2	3	4	5
Mean	3.1202	3.0204	3.0987	3.1585	3.0044
Variance	3.0505	2.8703	3.0124	3.1080	2.9966

For the problem d)

The second approach:

I set n=3 in order to compare the result of this approach with the result from the other approaches. Following the instruction in the document, I ran the program to generate 1000 samples from the

Erlang-3 distribution. In addition to this, I also calculated the mean and variance of these samples. After that, I repeated this process for five times to get five sets of results. The results are as follows:

	1	2	3	4	5
Mean	3.0355	2.8037	3.0168	2.9501	2.9612
Variance	2.9185	2.4448	2.9462	2.7838	2.9613

For the problem e)

The third approach:

I set n=3 in order to compare the result of this approach with the result from the other approaches. Following the instruction in the document, I ran the program to generate 1000 samples from the Erlang-3 distribution. In addition to this, I also calculated the mean and variance of these samples. After that, I repeated this process for five times to get five sets of results. The results are as follows:

	1	2	3	4	5
Mean	3.1398	3.0862	3.0231	3.1246	3.1441
Variance	2.9612	3.0829	2.8005	3.1192	3.1521

For the problem f)

I ran the program for five times to find that the optimal value for N is about 3000000, my results are as follows:

Time (Sec)	1	2	3	4	5
The first approach	0.5769	0.5874	0.5937	0.6004	0.5819
The second approach	0.3781	0.3866	0.3675	0.3773	0.3709
The third approach	29.3117	27.7855	27.3421	28.5988	28.8897

It can be seen from above table that the first approach is the fastest whereas the third way is slowest. When N=3000000, it takes about 30 seconds for the third approach to generate these samples.

5. References

1. *Alberto Leon-Garcia. (2008). Probability, Statistics, and Random Processes for Electrical Engineering. Upper Saddle River, NJ 07458. Pearson Education, Inc.*

2. *Zhou Sheng, Shiqian Xie, Chengyi Pan. (2008). Probability Theory and Mathematical Statistics. No.4, Dewai Street, Xicheng District, Beijing. Higher Education Press.*

6. Source Code

For the problem b)

```
a=linspace(0,5,101);
Fx=zeros(1,101);
for i=1:1000
   b(i)=rand();
   c(i)=-log(b(i));
   for j=1:101
    if(c(i)<=a(j))
    Fx(j)=Fx(j)+1;
    end
   end
end
Fx=Fx/1000;
plot(a,Fx,'b');
hold on;
y=1-exp(-a);
plot(a,y,'r');
grid on
title('The empirical distribution')
ylabel('The value of probability')
xlabel('The value taken on by x')
```

For the problem c), d) and e)

```
for i=1:1000
   for j=1:3
    b(j)=rand;
    c(j)=-log(b(j));
   end
   a(i)=c(1)+c(2)+c(3);
end
```

```
mean_er=mean(a);
var_er=var(a);

for i=1:1000
    for j=1:3
        d(j)=rand;
    end
e(i)=d(1)*d(2)*d(3);
f(i)=-log(e(i));
end
mean_com=mean(f);
var_com=var(f);

j=1;
i=1;
while (j<=1000)
    g(i)=unifrnd(0,10);
    h(i)=rand;
    p(i)=(g(i))^2*exp(-g(i))*5/2;
   if (h(i)<=p(i))
       k(j)=g(i);
       j=j+1;
   end
   i=i+1;
end
mean_re=mean(k);
var_re=var(k);
```

for the problem f)

```
tic;
for i=1:3000000
    for j=1:3
     b(j)=rand;
     c(j)=-log(b(j));
    end
    a(i)=c(1)+c(2)+c(3);
end
toc;
mean_er=mean(a);
var_er=var(a);

tic;
for i=1:3000000
    for j=1:3
```

```
        d(j)=rand;
    end
e(i)=d(1)*d(2)*d(3);
f(i)=-log(e(i));
end
toc;
mean_com=mean(f);
var_com=var(f);

tic;
j=1;
i=1;
while (j<=3000000)
    g(i)=unifrnd(0,10);
    h(i)=rand;
    p(i)=(g(i))^2*exp(-g(i))/(4*exp(-2));
   if (h(i)<=p(i))
       k(j)=g(i);
       j=j+1;
   end
   i=i+1;
end
toc;
mean_re=mean(k);
var_re=var(k);
```