

2.4

2.4 Simulation of Random Samples from Parametric Distributions

All programs are attached at the end of the report.

1. The Exponential Distributions

Question 1:

Using the probability density function as given, the equation becomes

$$\frac{1}{2} = \int_0^m f(x | \theta) dx = \int_0^m \theta e^{-\theta x} dx = 1 - e^{-\theta m}$$

Hence $\theta = \frac{\log 2}{m}$, where log denotes the natural logarithm.

Moreover,

$$g(x | m) = f(x | \theta(m)) = \frac{\log 2}{m} e^{-\frac{\log 2}{m} x} = \frac{\log 2}{m} \left(\frac{1}{2}\right)^{\frac{x}{m}} \quad (1)$$

Question 2:

To compute x_i , note that by the definition of x_i we have

$$x_i = -\frac{\log(1 - u_i)}{\theta_0}$$

We run the case where $n = 6$, $\theta_0 = 1.2$. The result is listed below.

Table 1: Random Samples generated by rand when $n = 6$, $\theta_0 = 1.2$						
u_i	0.031833	0.276923	0.046171	0.097132	0.823458	0.694829
x_i	0.026959	0.270200	0.039393	0.085149	1.445163	0.989068

The resulting *loglikelihood* function $l_n(m)$ is plotted in the following:

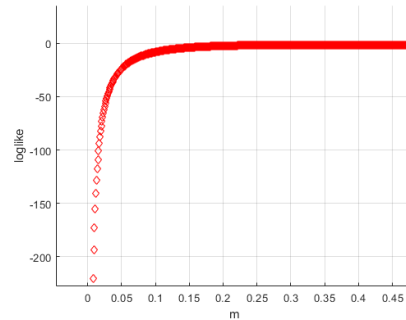


Figure 1: plot of loglike function against median for $n = 6$

MLE is obtained by maximising the *loglikelihood*:

$$0 = \frac{d}{dm} l_n(m) = \frac{d}{dm} \log\left(\left(\frac{\log 2}{m}\right)^n \left(\frac{1}{2}\right)^{\frac{\sum_0^n x_i}{m}}\right) = -\frac{n}{m} + \frac{\log 2 \sum_0^n x_i}{m^2} \quad (2)$$

Solving equation (2) gives $m_{MLE} = \frac{\log 2}{n} \sum_1^n x_i$. Substituting x_i 's from above, the MLE for this data x is $m_{MLE} = 0.32993(m_0 = 0.57762)$. We see that the MLE does not deviate much from m_0 (in the sense of absolute value). This is expected, since m_{MLE} is an unbiased estimator for the true median m_0 :

$$E(m_{MLE}) = E\left(\frac{\log 2}{n} \sum_1^n x_i\right) = \frac{\log 2}{n} \sum_1^n E(x_i) = \frac{\log 2}{\theta} = m_0$$

Question 3:

We repeat the procedure in Question 2 for $n = 25, 50, 100$, producing the following:

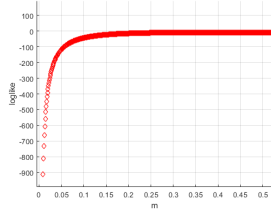


Figure 2: $n = 25$

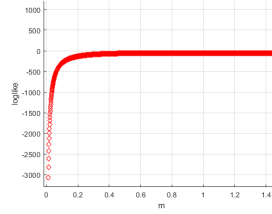


Figure 3: $n = 50$

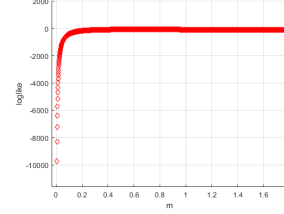


Figure 4: $n = 100$

For $n = 25, 50$, and 100 , the resulting m_{MLE} 's are $0.6318, 0.5667$, and 0.5706 .

From Figure 2, 3, and 4 we do not see any noticeable qualitative change in shape of $l_n(m)$.

Question 4: Let X and Y be identically independent distributed exponential variable with mean $\frac{1}{\theta}$. The moment generating function of X is

$$M_X(\lambda) = E(e^{\lambda X}) = \int_0^{+\infty} e^{\lambda x} \theta e^{-\theta x} dx = \int_0^{+\infty} \theta e^{(\lambda - \theta)x} dx = \begin{cases} \frac{\theta}{\theta - \lambda} & \lambda < \theta \\ \infty & \text{otherwise} \end{cases}$$

Now consider the moment generating function for $X + Y$:

$$E(e^{\lambda(X+Y)}) = E(e^{\lambda X + \lambda Y}) = E(e^{\lambda X} e^{\lambda Y}) = E(e^{\lambda X}) E(e^{\lambda Y}) = \left(\frac{\theta}{\theta - \lambda}\right)^2 = \left(1 - \frac{\lambda}{\theta}\right)^2$$

Hence $X + Y \sim \Gamma(2, \theta)$, since it has the same MGF as a $\Gamma(2, \theta)$ variable.

Question 5: Now we consider $f(x | \theta) = \theta^2 x e^{-\theta x}$ ($x \geq 0$). The distribution function $F(x)$ is given by

$$F(x) = \int_0^x \theta^2 v e^{-\theta v} dv = -\theta v e^{-\theta v} \Big|_0^x + \int_0^x \theta e^{-\theta v} dv = 1 - (\theta x + 1)e^{-\theta x},$$

where we integrate by parts in the second equality. A closed form of F^{-1} is not possible (in terms of elementary functions), but numerical methods (Newton's Method) do apply since $F(x)$ is smooth.

Question 6: MLE is obtained by maximising the *loglikelihood*:

$$0 = \frac{d}{d\theta} l_n(\theta) = \frac{d}{d\theta} (2n \log(\theta) + \sum_1^n \log(x_i) - \theta \sum_1^n x_i) = \frac{2n}{\theta} - \sum_1^n x_i \quad (3)$$

Solving Equation (3) we see that the MLE is $\theta_{MLE} = \frac{2n}{\sum_1^n x_i}$.

Question 7:

Now take $\theta_0 = 2.2$. We still use the Uniform Inversion Method to generate the desired random sample. Although an analytic solution cannot be found, we may use Newton's Method to approximate the solution to the equation

$$1 - (\theta x_i + 1)e^{-\theta x_i} = u_i \quad (4)$$

We implement this scheme and plot $l_n(\theta)$ against θ as follows:

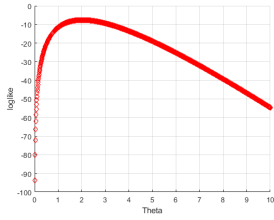


Figure 5: $n = 10$

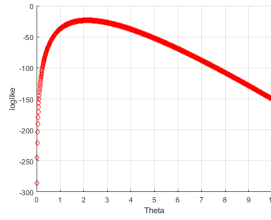


Figure 6: $n = 30$

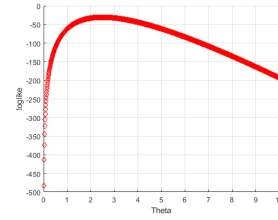


Figure 7: $n = 50$

The resulting θ_{MLE} 's are 2.0192, 2.1497, and 2.2939, corresponding to $n = 10$, 30, and 50. From Figure 5, 6, and 7 we don't see any noticeable change in the shape of $l_n(\theta)$. But one noticeable difference between Figure 5, 6 and 7 and Figure 2, 3, and 4 is that the curve decreases much faster in Figure 5, 6, and 7.

Question 8:

Following the procedure as given in the question we generate the histogram for

$\theta_{MLE}(1), \dots, \theta_{MLE}(200)$, each with sample size 10.

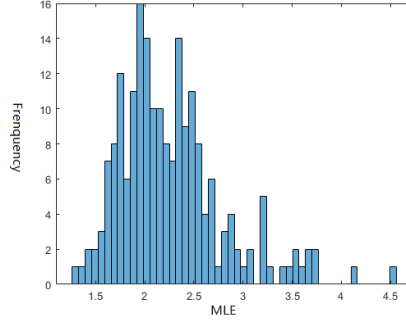


Figure 8: Sample size = 10, 200 samples

If we change n from 10 to 50, the histogram is

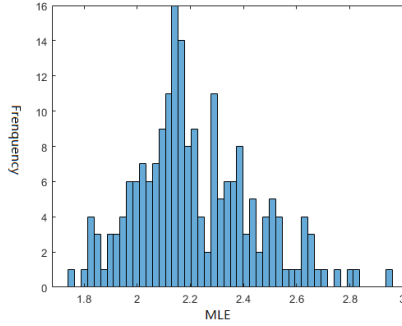


Figure 9: Sample size = 50, 200 samples

When we increase n from 10 to 50, the shape of MLE remains the same but the range of MLE's (the range of θ_{MLE} is the range that contributes mostly to the distribution) becomes much narrower. The modes of θ_{MLE} in two histograms are the same.

2. The Normal Distributions

Question 9:

Suppose $f(\phi, v) = \frac{1}{4\pi} e^{-v/2}$, $X = \mu_1 + \sigma\sqrt{V} \cos\Phi$, and $Y = \mu_2 + \sigma\sqrt{V} \sin\Phi$.

Then $V = \frac{1}{\sigma^2}((X - \mu_1)^2 + (Y - \mu_2)^2)$, $\Phi = \arctan(\frac{Y - \mu_2}{X - \mu_1})$. Hence $\left| \frac{\partial(\phi, v)}{\partial(x, y)} \right|$

$$= \left| \begin{bmatrix} \frac{-(y - \mu_2)}{(x - \mu_1)^2 + (y - \mu_2)^2} & \frac{2(x - \mu_1)}{\sigma^2} \\ \frac{-(x - \mu_1)}{(x - \mu_1)^2 + (y - \mu_2)^2} & \frac{2(y - \mu_2)}{\sigma^2} \end{bmatrix} \right| = \frac{2}{\sigma^2}.$$

Apply the transformation formula, $g(x, y) = \frac{1}{2\pi\sigma^2} e^{-((X-\mu_1)^2 + (Y-\mu_2)^2)/2\sigma^2}$. Therefore X and Y are independent $N(\mu_1, \sigma^2)$ and $N(\mu_2, \sigma^2)$ random variables.

Now let A, B be two independent $\text{Unif}[0, 1]$ random variables. Define $\Phi = 2\pi A$ and $V = -2\log(1 - B)$, then Φ and V are independent. In addition the probability density function of Φ is $g(\phi) = \frac{1}{2\pi}$, $0 \leq \phi \leq 2\pi$. The distribution function of V is

$$H(v) = P(V \leq v) = P(-2 \log(1-B) \leq v) = P(B \leq 1 - e^{-v/2}) = 1 - e^{-v/2}, v \geq 0 \quad (5)$$

The p.d.f of V comes from differentiating (5): $h(v) = \frac{1}{2}e^{-v/2}$. Thus the joint p.d.f for Φ and V is $f(\phi, v) = \frac{1}{4\pi}e^{-v/2}$, $0 \leq v \leq 2\pi$, $v \geq 0$. By defining X, Y in the same way as in Question 9 we have effectively found a way of generating normal random variables.

Question 10:

The program to generate a random sample is attached at the end of the report.

An 80% confidence interval can be constructed by noticing that for (X_1, \dots, X_n) i.i.d $N(\mu, 1)$ normal variables the random variable defined by $\frac{\bar{X} - \mu}{\sqrt{1/n}}$ is a $N(0, 1)$ variable, where \bar{X} is the average of (X_1, \dots, X_n) . Let $\Phi(\alpha)$ be the α upper point of $N(0, 1)$, i.e.

$$P(Z \leq \Phi(\alpha)) = 1 - \alpha$$

Then the required 80% confidence interval is given by

$$-\Phi(0.1) \leq \frac{\bar{X} - \mu}{\sqrt{1/n}} \leq \Phi(0.1)$$

or equivalently by

$$\bar{X} - \sqrt{\frac{1}{n}}\Phi(0.1) \leq \mu \leq \bar{X} + \sqrt{\frac{1}{n}}\Phi(0.1)$$

Question 11: We run the program in Question 10 with $\mu = 0$, $n = 100$:

Table 2: Simulation of a Normal Random Sample where $n = 100$, $\mu = 0$		
Sample Mean	Lower Bound of CI	Upper Bound of CI
-0.0766	-0.2047	0.0516

We see that the confidence interval does contain μ .

Table 3: The Result of 25 Executions of the Program

Sample Mean	Lower Bound of CI	Upper Bound of CI	
0.160912	0.032752	0.289072	0
-0.06302	-0.19118	0.065141	1
-0.12063	-0.24879	0.007534	1
-0.01773	-0.14589	0.110428	1
0.040278	-0.08788	0.168438	1
0.102936	-0.02522	0.231096	1
0.000966	-0.12719	0.129126	1
0.01573	-0.11243	0.14389	1
-0.01925	-0.14741	0.108913	1
-0.00241	-0.13057	0.125751	1
-0.1377	-0.26586	-0.00954	0
-0.22447	-0.35263	-0.09631	0
-0.00827	-0.13643	0.119893	1
-0.00184	-0.13	0.126324	1
0.193565	0.065405	0.321725	0
0.03678	-0.09138	0.16494	1
0.013125	-0.11504	0.141285	1
0.133845	0.005685	0.262005	0
0.070625	-0.05754	0.198785	1
-0.08706	-0.21522	0.041099	1
-0.0008	-0.12896	0.127357	1
0.093917	-0.03424	0.222077	1
-0.0555	-0.18366	0.072655	1
0.045092	-0.08307	0.173252	1
0.008515	-0.11964	0.136675	1

Now we run the program 25 times and record the results in the following: There are 5 instances in which the CI does not contain the true mean.

Question 12:

5 times, since changing μ only shifts the distribution but does not affect the variance.

3. The χ^2 Distributions

Question 13:

Recall that if $Y = \sum_1^n X_i^2$ where X_i 's are i.i.d $N(0,1)$ then $Y \sim \chi_n^2$.

The program that simulates n samples, all following the chi-square distribution with d degrees of freedom, is attached at the end. The program produces the following:

(a): chi-square with 1 degree of freedom:

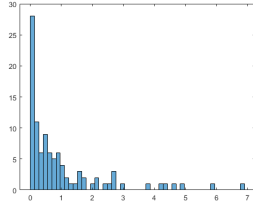


Figure 10: $n = 100$

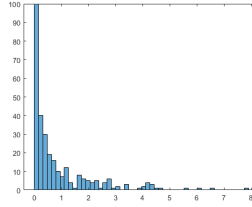


Figure 11: $n = 300$

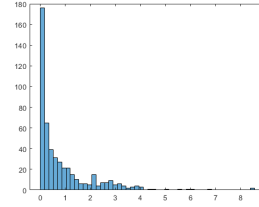


Figure 12: $n = 500$

(b): chi-square with 5 degrees of freedom:

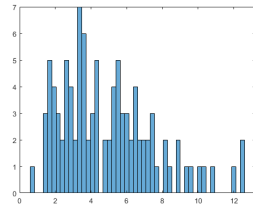


Figure 13: $n = 100$

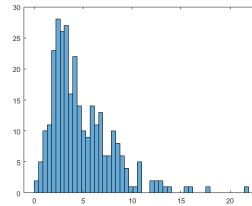


Figure 14: $n = 300$

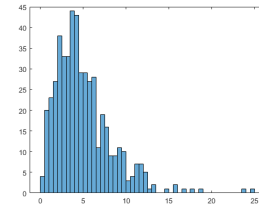


Figure 14: $n = 500$

(c): chi-square with 40 degrees of freedom:

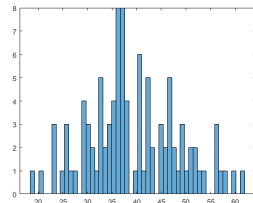


Figure 15: $n = 10$

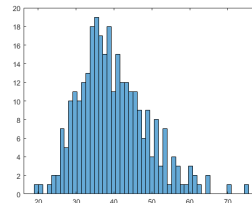


Figure 16: $n = 30$

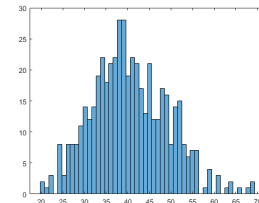


Figure 17: $n = 50$

The histogram becomes more symmetric and flatter as the degree of freedom increases: as N becomes large, the variance becomes larger and the Central Limit Theorem comes into effect: the distribution will be approximately a normal distribution.

Reference

D. A. Berry and B. W. Lindgren, Statistics, Theory and Methods, Duxbury Press, 1995, ISBN 0-534-50479-5.

Source Code

```
1 function [U,X] = Sim(n,t)
2 U = rand(1,n);
3 X = zeros(1,n);
4 Y = ones(1,n);
5 X = -log(Y-U)/t
6 end
```

```
1 [U,X] = Sim(6,1.2)
2 for i = 1:2000
3 M(i) = 0.001*i;
4 end
5
6 L = zeros(1,2000);
7 for i = 1:2000
8 L(i) = loglike(X,0.001*i);
9 end
10 xlabel('m')
11 ylabel('loglike')
12 grid on
13 hold on
14 plot(M,L,'rd')
15 hold off
16 y = 0;
17 for i = 1:6
18 y = y+X(i);
19 end
20 m = y*log(2)/6
```

```
1 %Part I
2 [U,X] = Sim(25,1.2)
```

```

3  for i = 1:2000
4  M(i) = 0.001*i;
5  end
6
7  L = zeros(1,2000);
8  for i = 1:2000
9  L(i) = loglike(X,0.001*i);
10 end
11 xlabel('m')
12 ylabel('loglike')
13 grid on
14 hold on
15 plot(M,L,'rd')
16 hold off
17 y = 0;
18 for i = 1:25
19 y = y+X(i);
20 end
21 m = y*log(2)/25
22
23 %Part II
24 [U,X] = Sim(50,1.2)
25 for i = 1:2000
26 M(i) = 0.001*i;
27 end
28
29 L = zeros(1,2000);
30 for i = 1:2000
31 L(i) = loglike(X,0.001*i);
32 end
33 xlabel('m')
34 ylabel('loglike')
35 grid on
36 hold on
37 plot(M,L,'rd')
38 hold off
39 y = 0;
40 for i = 1:50
41 y = y+X(i);
42 end
43 m = y*log(2)/50
44
45 %Part III
46 [U,X] = Sim(100,1.2)
47 for i = 1:2000
48 M(i) = 0.001*i;
49 end
50
51 L = zeros(1,2000);
52 for i = 1:2000
53 L(i) = loglike(X,0.001*i);
54 end
55 xlabel('m')
56 ylabel('loglike')
57 grid on
58 hold on
59 plot(M,L,'rd')

```

```

60 hold off
61 y = 0;
62 for i = 1:100
63 y = y+X(i);
64 end
65 m = y*log(2)/100

```

```

1 function [l] = loglike(X,m)
2 n = length(X);
3 P = zeros(1,n);
4 l = 0;
5 for i = 1:n
6 P(i) = (log(2)/m)*(1/2)^(X(i)/m);
7 end
8 for i = 1:n
9 l = l+log(P(i));
10 end
11 end

```

```

1 function [U,X] = Sim2(n,t)
2 U = rand(1,n);
3 X = zeros(1,n);
4 for i = 1:n
5 X(i) = Newton(t,U(i));
6 end
7 end

```

```

1 L = zeros(1,25);
2 U = zeros(1,25);
3 P = zeros(1,25);
4 for i = 1:25
5 [X,P(i),L(i),U(i)] = Normal(100,0,1);
6 end

```

```

1 T = zeros(1,200)
2 for i = 1:200
3 [U,X] = Sim2(10,2.2)
4 y = 0;
5 for j = 1:10
6 y = y+X(j);
7 end
8 T(i) = 20/y
9 end
10
11 histogram(T,50)
12
13 T = zeros(1,200)
14 for i = 1:200
15 [U,X] = Sim2(50,2.2)
16 y = 0;

```

```

17 for j = 1:50
18 y = y+X(j);
19 end
20 T(i) = 100/y
21 end
22
23 histogram(T,50)

```

```

1 %n=10
2 [U,X] = Sim2(10,2.2)
3 M = zeros(1,1000);
4 for i = 1:1000
5 M(i) = 0.01*i;
6 end
7 L = zeros(1,1000);
8 for i = 1:1000
9 L(i) = loglike2(X,0.01*i);
10 end
11 xlabel('Theta')
12 ylabel('loglike')
13 grid on
14 hold on
15 plot(M,L,'rd')
16 hold off
17 y = 0;
18 for i = 1:10
19 y = y+X(i);
20 end
21 m = 20/y
22 %n=30
23 [U,X] = Sim2(30,2.2)
24 M = zeros(1,1000);
25 for i = 1:1000
26 M(i) = 0.01*i;
27 end
28 L = zeros(1,1000);
29 for i = 1:1000
30 L(i) = loglike2(X,0.01*i);
31 end
32 xlabel('Theta')
33 ylabel('loglike')
34 grid on
35 hold on
36 plot(M,L,'rd')
37 hold off
38 y = 0;
39 for i = 1:30
40 y = y+X(i);
41 end
42 m = 60/y
43 %n = 50
44 [U,X] = Sim2(50,2.2)
45 M = zeros(1,1000);
46 for i = 1:1000
47 M(i) = 0.01*i;
48 end

```

```

49 L = zeros(1,1000);
50 for i = 1:1000
51 L(i) = loglike2(X,0.01*i);
52 end
53 xlabel('Theta')
54 ylabel('loglike')
55 grid on
56 hold on
57 plot(M,L,'rd')
58 hold off
59 y = 0;
60 for i = 1:50
61 y = y+X(i);
62 end
63 m = 100/y

```

```

1 function [X,p,l,u] = Normal(n,m,s)
2 A = rand(1,n);
3 B = zeros(1,n);
4 X = zeros(1,n);
5 V = zeros(1,n);
6 for i = 1:n
7 dummy = rand;
8 B(i) = dummy;
9 while abs(dummy-1)<0.00001
10 dummy = rand;
11 B(i) = dummy;
12 end
13 end
14 Phi = cos(2*pi*A);
15 for j = 1:n
16 V(j) = sqrt(-2*log(1-B(j)));
17 end
18
19 for k = 1:n
20 X(k) = m + s*V(k)*Phi(k);
21 end
22 p = 0;
23 for t = 1:n
24 p = p+X(t);
25 end
26 p = p/n;
27 l = p - sqrt(1/n)*1.2816;
28 u = p + sqrt(1/n)*1.2816;
29 end

```

```

1 function [x] = Newton(t,u)
2 x = 0.5;
3 while 1
4 if abs(1-u-(t*x+1)*exp(-t*x)) > 0.00001
5 x = x - (1-u-(t*x+1)*exp(-t*x))/(x*exp(-t*x)*t^(2));
6 else
7 break
8 end

```

```
9 end
10 end
```

```
1 function [l] = loglike2(X,t)
2 n = length(X);
3 l = 2*n*log(t);
4 for i = 1:n
5 l = l + log(X(i))-t*X(i);
6 end
7 end
```

```
1 function [S] = Chi(n,d)
2 S = zeros(1,n);
3 for i = 1:n
4 [X,p,l,u] = Normal(d,0,1);
5 X = X.^(2);
6 r = 0;
7 for j = 1:d
8 r = r+X(j);
9 end
10 S(i) = r;
11 end
12 end
```

```
1 [S] = Chi(100,1)
2 histogram(S,50)
3 [S] = Chi(300,1)
4 histogram(S,50)
5 [S] = Chi(500,1)
6 histogram(S,50)
7 [S] = Chi(100,5)
8 histogram(S,50)
9 [S] = Chi(300,5)
10 histogram(S,50)
11 [S] = Chi(500,5)
12 histogram(S,50)
13 [S] = Chi(100,40)
14 histogram(S,50)
15 [S] = Chi(300,40)
16 histogram(S,50)
17 [S] = Chi(500,40)
18 histogram(S,50)
```