Scilab Manual for
Probability Theory and Statistics Lab
by Prof S N Chandra Shekhar
Electronics Engineering
Sreenidhi Institute Of Science & Technology[1]

Solutions provided by
Prof S N Chandra Shekhar
Electronics Engineering
Sreenidhi Institute Of Science & Technology

April 22, 2026

Contents

List of Experiments

Experiment: 1

Poisson Distribution

Scilab code Solution 1.1 1

```
1
2  // Poisson Distribution
3  // OS: Windows 10
4  // Scilab 5.5.2
5
6  clear;
7  clc;
8  close;
9
10 Frequency =[12 ,10 ,19 ,17 ,10 ,8 ,7 ,5 ,5 ,3 ,3 ,1]
11 n= sum ( Frequency )
12 X =[0 ,1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,11]
13
14 F =[]
15 P1 =[]
16 P2 =[]
17
18 mprintf (" ——————————————————————————————\n")
19 mprintf ("      Arrivals            Frequency \n")
20 mprintf (" ——————————————————————————————")
21 for i =1:12
```

```scilab
22      mprintf("\n        %d                              %d",i-1,
              Frequency(i))
23      F($+1)=Frequency(i)
24      product1=F(i)*X(i)
25      product2=F(i)*(X(i)**2)
26      P1($+1)=product1
27      P2($+1)=product2
28      sumP1 =sum(P1)
29      sumP2 =sum(P2)
30  end
31  mprintf("\n    ————————————————————————————————")
32  Xbar = sumP1/n
33  S_sqr = (sumP2 - n*Xbar**2)/(n-1)
34  S = sqrt(S_sqr)
35
36  //Result
37  mprintf("\n\n Total number of arrivals = %d",n)
38  mprintf("\nalpha = Xbar = Sum(fj*Xj)/n = %f",Xbar)
39  mprintf("\nSample variance = (Sum(fj*Xj^2) - n*Xbar
          ^2) / n-1 = %f",S_sqr)
```

Experiment: 2

Lognormal Distribution

Scilab code Solution 2.2 2

```
1
2  //Lognormal Distribution
3  //OS:Windows 10
4  //Scilab 5.5.2
5
6  clear all;
7  clc;
8  close;
9
10 X = [18.8, 27.9, 21.0, 6.1, 37.4, 5.0, 22.9, 1.0,
       3.1, 8.3] // rate
11
12 X1=[]
13 X2=[]
14
15 n=length(X)
16
17 for i=1:n
18     X1($+1)=log(X(i))
19     X2($+1)=X1(i)**2
20 end
```

```
21  Xbar = sum(X1)/n
22  S2 = (sum(X2) - (n*(Xbar)**2))/(n-1)
23
24  //Result
25  mprintf("
            ————————————————————————————\n")
26  mprintf("        Rates(X)                        lognormal(X1)
        ")
27  mprintf("\n
            ————————————————————————————\n")
28  for i=1:10
29      mprintf("\n        %.1f\t\t\t%.1f",X(i),X1(i))
30  end
31  mprintf("\n
            ————————————————————————————\n")
32  mprintf("\n N=%d",n)
33  mprintf("\n mu = Xbar = Sum(lognormal)/N = %f",Xbar)
34  mprintf("\nS^2 = (Sum(X1i^2) - (N*(Xbar)^2))/(N-1) =
        %f",S2)
```

Experiment: 3

Normal Distribution

Scilab code Solution 3.3 3

```scilab
1
2  //Normal Distribution
3  //OS:Windows 10
4  //Scilab 5.5.2
5
6  clear all ;
7  clc ;
8  close ;
9
10
11  mprintf ("In Normal Distribution : \n mu = Xbar \n
        sigma^2 = S^2 \n")
12
13  value =
        [99.79 ,100.26 ,100.23 ,99.55 ,99.96 ,99.56 ,100.41 ,100.27 ,99.62 ,99.90 ,
14
15  V = gsort ( value , 'g' , 'i ')
16  V2 =[]
17  n = length (V)
18
```

```
19  for i=1:n
20      V2($+1)=(V(i)**2)
21  end
22
23  Xbar = sum(V)/n
24  S2 = (sum(V2) - (n*(Xbar)**2))/(n-1)
25  S = sqrt(S2)
26
27  //Result
28  mprintf("\n
                   ————————————————————————————\n")
29  mprintf("        j\t\tvalue\tj\tvalue")
30  mprintf("\n
                   ————————————————————————————\n")
31  ul=n/2
32  for i=1:ul
33      mprintf("\n      %d\t\t%.2f\t%d\t%.2f",i,V(i),(i
            +10),V(i+10))
34  end
35  mprintf("\n
                   ————————————————————————————\n")
36  mprintf("\n N=%d",n)
37  mprintf("\n mu = Xbar = Sum(V)/N = %f",Xbar)
38  mprintf("\nsigma^2 = (S^2) = (%f)^2 second^2",S)
```

Experiment: 4

Gamma Distribution

Scilab code Solution 4.4 4

```
1
2  //Gamma Distribution
3  //OS: Windows 10
4  //Scilab 5.5.2
5
6  clear all;
7  clc;
8  close;
9
10 Order =
       [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]

11 Leadtime = [70.292, 10.107, 48.386, 20.480, 13.053,
       25.292, 14.713, 39.166, 17.421, 13.905, 30.215,
       17.137, 44.024, 10.552, 37.298, 16.314, 28.073,
       39.019,32.330,36.547]
12
13 X = []
14 logX =[]
15
16 n =length(Order)
```

```scilab
17  Xbar = sum(Leadtime)/n
18  logXbar = log(Xbar)
19  for i=1:n
20      X($+1)=Leadtime(i)
21      logX($+1)=log(Leadtime(i))
22  end
23  sumlogX = sum(logX)
24  M = logXbar - (sumlogX/n)
25  Minv = 1/M
26  betav = 3.7411
27  thetav = 1/Xbar
28
29  //Result
30  mprintf("\n
        _____\
        n")
31  mprintf("    Order\tLead time\tOrder\tLead time")
32  mprintf("\n
        _____\
        n")
33  ul=(length(X))/2
34  for i=1:ul
35      mprintf("\n    %d\t\t%.3f\t\t%d\t%.3f",i,Leadtime
            (i),(i+10),Leadtime(i+10))
36  end
37  mprintf("\n
        _____\
        n")
38  mprintf("\n Xbar = Sum(leadtime)/n = %f",Xbar)
39  mprintf("\n ln Xbar = %.2f",logXbar)
40  mprintf("\n Sum(ln(Xi)) = %.2f",sumlogX)
41  mprintf("\n M = ln(Xbar) - (Sum(ln(Xi))/n) = %.2f",M
        )
42  mprintf("\n 1/M = %.2f",Minv)
43  mprintf("\n beta value for 1/M = 7.11 is 3.7411")
44  mprintf("\n Beta = %.2f",betav)
45  mprintf("\n Theta = 1/Xbar = %.2f",thetav)
```

Experiment: 5

Calculate Mean and Standard Deviation

Scilab code Solution 5.5 5

```
1
2  //Calculate Mean and Standard Deviation
3  //OS: Windows 10
4  //Scilab 5.5.2
5
6  clear;
7  clc;
8  close;
9
10 disp('The first row denotes the size of item')
11 A(1,:)=[6 7 8 9 10 11 12];
12 disp('The second row denotes the corresponding
        frequency (f)')
13 A(2,:)=[3 6 9 13 8 5 4];
14 disp('The third row denotes the corresponding
        deviation (d)')
15 A(3,:)=[-3 -2 -1 0 1 2 3];
16 disp('The fourth row denotes the corresponding f*d '
        )
```

```matlab
17  for i=1:7
18    A(4,i)=A(2,i)*A(3,i);
19  end
20  disp('The fifth row denotes the corresponding f*d^2'
       )
21  for i=1:7
22    A(5,i)=A(2,i)*(A(3,i)^2);
23  end
24  disp(A)
25  b=0;
26  for i=1:7
27    b=b+A(4,i);
28  end
29  disp(b,'Sum of fourth row elements')
30  c=0
31  for i=1:7
32    c=c+A(5,i);
33  end
34  disp(c,'Sum of fifth row elements')
35  d=0;
36  for i=1:7
37    d=d+A(2,i);
38  end
39  disp(d,'Sum of all frequencies')
40  mean_value=9+b/d
41  disp(mean_value,'Mean=9+b/d=')
42  standard_deviation=(c/d)^0.5
43  disp(standard_deviation,'Standard Deviation=(c/d)
       ^0.5')
```

Experiment: 6

Calculate Coefficient Of Correlation

Scilab code Solution 6.6 6

```
1
2  //Calculate Coefficient of Correlation
3  //OS: Windows 10
4  //Scilab 5.5.2
5
6  clear;
7  clc;
8
9  disp('The first row of A denotes the corresponding I
       .R. of students ')
10 A(1,:)=[105 104 102 101 100 99 98 96 93 92];
11 disp('The second row denotes the corresponding
       deviation of I.R.')
12 for i=1:10
13    A(2,i)=A(1,i)-99;
14 end
15 disp('The third row denotes the square of
       corresponding deviation of I.R. ')
16 for i=1:10
```

```
17    A(3,i)=A(2,i)^2;
18  end
19  disp('The fourth row denotes the corresponding   E.R.
        of students ')
20    A(4,:)=[101 103 100 98 95 96 104 92 97 94];
21  disp('The fifth row denotes the corresponding
       deviation of E.R. ')
22  for i=1:10
23    A(5,i)=A(4,i)-98;
24  end
25  disp('The sixth row denotes the square of
        corresponding deviation of E.R. ')
26  for i=1:10
27    A(6,i)=A(5,i)^2;
28  end
29  disp('The seventh row denotes the product of the two
        corresponding deviations ')
30  for i=1:10
31    A(7,i)=A(2,i)*A(5,i);
32  end
33  disp(A)
34  a=0;
35  for i=1:10
36    a=a+A(1,i);
37  end
38  disp(a,'The sum of elements of first row')
39  b=0;
40  for i=1:10
41    b=b+A(2,i);
42  end
43  disp(b,'The sum of elements of second row')
44  c=0;
45  for i=1:10
46    c=c+A(3,i);
47  end
48  disp(c,'The sum of elements of third row')
49  d=0;
50  for i=1:10
```

```
51     d=d+A(4,i);
52  end
53  disp(d,'The sum of elements of fourth row')
54  e=0;
55  for i=1:10
56     e=e+A(5,i);
57  end
58  disp(e,'The sum of elements of fifth row')
59  f=0;
60  for i=1:10
61     f=f+A(6,i);
62  end
63  disp(f,'The sum of elements of sixth row')
64  g=0;
65  for i=1:10
66     g=g+A(7,i);
67  end
68  disp(g,'The sum of elements of seventh row')
69  coefficient_correlation=g/(c*f)^0.5
70  disp(coefficient_correlation,'Coefficient of
       correlation=')
```

Experiment: 7

Determine Frequency Distribution

check Appendix AP 1 for dependency:

    freqdist.sci

Scilab code Solution 7.7 7

```
1  //Determine Frequency Distribution
2  //OS: Windows 10
3  //Scilab 5.5.2
4  //The dependency file is freqdist.sci
5
6  x=int(100*rand(1,100));
7  xclass=[0:10:100];
8  freqdist(x,xclass)
```

Experiment: 8

Discrete Probability Distributions

Scilab code Solution 8.8 8

```scilab
1
2  // Discrete  Probability  Distributions
3  //OS: Windows  10
4  // Scilab  5.5.2
5
6  clear ;
7  clc ;
8  close ;
9
10  // ( i ) Binomial
11  deff ( ' [CC]=C( n , r ) ' , 'CC=gamma( n+1 ) . / ( gamma( r +1 ) . *
       gamma( n−r +1 ) ) ')       // Binomial  coefficient
12  deff ( ' [ bb]=b ( x , n , p ) ' , 'bb=C( n , x ) . * p . ^ x . * (1−p ) . ^ ( n–x ) '
       )                  // Binomial  pmf
13  deff ( ' [BB]=B( x , n , p ) ' , 'BB=sum( b ( [ 0 : 1 : x ] , n , p ) ) ')
                            // Binomial  CDF
14  n=10;
15  p=0.25;
16
```

18

```
17  //P(X=2)
18  p=b(2,10,0.25)
19  disp(p)
20
21  //pmf
22  b10=[];
23  for j=0:10,b10=[b10 b(j,10,0.25)]; end;
24  disp(b10)
25
26  //CDF
27  x=2
28  p=B(2,10,0.25)
29  disp(p)
30
31  //P(X>2)=1-P(X<=2)
32  p=1-B(2,10,0.25)
33  disp(p)
34
35  //List of values of CDF
36  B10=[];
37  for j=0:10,B10=[B10 B(j,10,0.25)]; end;
38  disp(B10)
39
40  //(ii)Geometric
41  deff('[gg]=g(x,p)','gg=p.*(1-p).^(x-1)')
                                //Geometric pmf
42  deff('[GG]=G(x,p)','GG=sum(g([1:x],p))')
                                //Geometric CDF
43
44  //P(X=3) and P(X=5),p=0.50
45  p=g(3,0.50)
46  disp(p)
47  p=g(5,0.50)
48  disp(p)
49
50  //Calculate vector of values of pmf
51  disp(g([1:10],0.5))
52
```

19

```scilab
53  //P(X6) ,P(X3) ,P(X1)
54  p1=G(6,0.5)
55  disp(p1)
56  p2=G(3,0.5)
57  disp(p2)
58  p3=G(1,0.5)
59  disp(p3)
60
61  //Vector of values of CDF
62  G10=[];
63  for j=1:10, G10=[G10 G(j,0.5)]; end;
64  disp(G10)
65
66  //(iii)Hypergeometric
67  deff('[hh]=h(x,N,n,a)','hh=C(a,x).*C(N-a,n-x)./C(N,n
       )')                //Hypergeometric pmf
68  deff('[HH]=H(x,N,n,a)','HH=sum(h([0:1:x],N,n,a))')
                          //Hypergeometric CDF
69  N=100;
70  n=20;
71  a=35;
72
73  //P(X=12)
74  p=h(12,N,n,a)
75  disp(p)
76
77  //Cumulative distribution ,x=12
78  cd_x=H(12,N,n,a)
79  disp(cd_x)
80
81  //Generate vector of values of pdf
82  disp(h([0:20],N,n,a))
83
84  //Generate vector of values of CDF
85  H10=[];
86  for j=1:10,H10=[H10 h(j,N,n,a)]; end;
87  disp(H10)
```

Experiment: 9

Statistical Inference -Continuous Probability Distribution

Scilab code Solution 9.9 9

```
1
2  //Statistical Inference-Continuous Probability
        Distributions
3  //OS:Windows 10
4  //Scilab 5.5.2
5
6  clear;
7  clc;
8  close;
9
10 //(i)Student t distribution
11 deff('[f]=fT(t,nu)','f=gamma((nu+1)./2).*(1+t.^2./nu
        ).^(-(nu+1)/2)/(sqrt(%pi*nu)*gamma(nu/2))')
12 tt=[-4:0.1:4];
13 ff=fT(tt,6);
14 figure
15 xlabel('t');
```

```
16  ylabel('fT(t)');
17  title('Student t − nu = 6');
18  plot(tt,ff,'x')
19
20  //Probability Calculation
21  [P,Q] = cdft("PQ",0.4,6)
22  disp(Q,'Q')
23  disp(P,'P')
24
25  //Inverse CDF Calculation
26  t = cdft("T",8,0.45,1-0.45)
27  disp(t,'t')
28
29  //Obtaining degrees of freedom
30  nu = cdft("Df",0.7,0.3,0.8)
31  disp(nu,'nu')
32
33  //Plotting CDF
34  xx=[-4:0.1:4];
35  yy=[];
36  for x=-4:0.1:4, yy=[yy cdft('PQ',x,6)]; end;
37  figure
38  xlabel('t');
39  ylabel('fX(t)');
40  title('Student t − nu = 6');
41  plot(xx,yy,'x')
42
43
44  //(ii)Chi−square distribution
45  xx = [0:0.1:10];
46  yy = [];
47  for x=0:0.1:10, yy=[yy cdfchi('PQ',x,4)]; end;
48  figure
49  xlabel('t');
50  ylabel('fX(t)');
51  title('Chi−square − nu = 4');
52  plot(xx,yy,'x')
53
```

```scilab
54  //Probability calculation
55  [P,Q] = cdfchi("PQ",1,10)
56  disp(Q,'Q')
57  disp(P,'P')
58
59  [P,Q] = cdfchi("PQ",0.2,10)
60  disp(Q,'Q')
61  disp(P,'P')
62
63  //Inverse CDF calculation
64  chi2 = cdfchi("X",4,0.4,0.6)
65  disp(chi2,'chi2')
66
67  //Calculating degrees of freedom
68  nu = cdfchi("Df",0.4,0.6,2.7)
69  disp(nu,'nu')
70
71  //Plotting CDF
72  deff('[f]=fC(x,nu)','f=x.^(nu/2-1).*exp(-x./2)/(2.^(
        nu/2).*gamma(nu./2))')
73  cc = [0:0.1:30];
74  ff = fC(cc,10);
75  figure
76  xlabel('chi^2');
77  ylabel('fC(chi^2)');
78  title('Chi-square - nu = 10');
79  plot(cc,ff,'x')
80
81
82  //(iii)F distribution
83  deff('[f]=fF(F,nuN,nuD)','f=gamma((nuN+nuD)./2).*(
        nuN./nuD).^(nuN./2).*F.^(nuN./2-1)./(gamma(nuN
        ./2).*gamma(nuD./2).*(1+nuN.*F./nuD).^((nuN+nuD)
        ./2))')
84  xx = [0:0.1:10];
85  ff = fF(xx,4,6);
86  figure
87  xlabel('F');
```

```
88  ylabel('fF(F)');
89  title('F distribution − nuNum = 4 − nuDen =6');
90  plot(xx,ff,'x')
91
92  //Probability Calculation
93  [P,Q] = cdff("PQ",1.2,6,12)
94  disp(Q,'Q')
95  disp(P,'P')
96
97  //Inverse CDF calculation
98  F = cdff("F",10,2,0.4,0.6)
99  disp(F,'F')
100
101  //Calculating degrees of freedom
102  nuNum = cdff('Dfn',5,0.4,0.6,0.8)
103  disp(nuNum,'nuNum')
104
105  //Plotting CDF
106  xx = [0:0.1:10];
107  yy = [];
108  for x=0:0.1:10, yy=[yy cdff('PQ',x,4,6)]; end;
109  figure
110  xlabel('t');
111  ylabel('fX(t)');
112  title('F − nuNum = 4 − nuDen =6');
113  plot(xx,yy,'x')
```

Experiment: 10

Weibull distribution

Scilab code Solution 10.10 10

```
1
2 //Weibull Distribution
3 //OS:Windows 10
4 //Scilab 5.5.2
5
6 clear;
7 clc;
8 close;
9
10 deff('[ww]=w(x,a,b)','ww=a.*b.*x^(b-1).*exp(-a.*x.^b
     )')                    //pdf
11 deff('[WW]=W(x,a,b)','WW=1-exp(-a.*x.^b)')
                                    //cdf
12 //alpha=2,beta=3
13 xx=(0:0.01:2);
14 yy=w(xx,2,3);
15 figure
16 xlabel('x');
17 ylabel('fX(x)');
18 title('Weibull distribution');
19 plot(xx,yy,'x')
```

```
20  yyy=[];
21  for x=0:0.01:2, yyy=[yyy W(x,2,3)]; end;
22  figure
23  xlabel('x');
24  ylabel('FX(x)');
25  title('Weibull distribution');
26  plot(xx,yyy,'x')
27
28  //Probability calculations
29  //P(X<1.5)
30  p1=W(1.5,2,3)
31  disp(p1,'P(X<1.5)')
32
33  //P(X>0.6)
34  p2=1-W(0.6,2,3)
35  disp(p2,'P(X>0.6)')
36
37  //P(0.5<X<1.2)
38  p3=W(1.2,2,3)-W(0.5,2,3)
39  disp(p3,'P(0.5<X<1.2)')
```

Appendix

Scilab code AP 1 ```1 clear;```
```scilab
2 clc;
3 close;
4
5 function freqdist(x,xclass)
6     [m n]=size(x);              //sample size
7     [m nB]=size(xclass);        //no.of class
         boundaries
8     k=nB - 1;
9
10    cmark = zeros(1,k);
11    for ii = 1:k
12         cmark(ii) = 0.5*(xclass(ii)+xclass(ii+1));
13    end
14
15    fcount=zeros(2,k);
16    fbelow=0;
17    fabove=0;
18
19    for ii=1:n
20         if x(ii) < xclass(1)
21             fbelow = fbelow + 1;
22         elseif x(ii) > xclass(nB)
23             fabove = fabove + 1;
24         else
25             for jj = 1:k
```

```
26                      if x(ii)>= xclass(jj) & x(ii)<
                            xclass(jj+1)
27                          fcount(jj) = fcount(jj) +1;
28                      end
29                  end
30          end
31      end
32
33      frel=fcount/n
34
35      fcumul = zeros(1,k);
36      fcumul(1) = fcount(1);
37      for ii = 2:k
38          fcumul(ii) = fcumul(ii-1) + fcount(ii);
39      end;
40
41      fcumulrel = fcumul/n;
42
43      mprintf('\n
        _____
        ');
44      mprintf('\nFrequency distribution');
45      mprintf('\n
        _____
        ');
46      mprintf('\n   Class\tLowBound\tUpBound\t\tClass
        Mark\tFreq.\t\tRelFreq.\tCumFreq.\tRelCumFreq
        .');
47      mprintf('\n
        _____
        ');
48      for ii = 1:k
49          mprintf('\n%5.0f\t%10.6g\t%10.6g\t%10.6g\
            t%10.6g\t%10.6g\t%10.6g\t%10.6g\t\n',ii,
            xclass(ii),xclass(ii+1),cmark(ii),fcount(
            ii),frel(ii),fcumul(ii),fcumulrel(ii))
50      end
```

```
51        disp('\n
          _____
          ');
52        disp(' ')
53
54        if fbelow ~= 0
55            mprintf('Outliers below minimum class
                  boundary = %10.6g \n',fbelow)
56        end
57        if fabove ~= 0
58            mprintf('Outliers above maximum class
                  boundary = %10.6g \n',fabove)
59        end
60        mprintf('Total no of data points = %10.6g \n',n)
              ;
61        mprintf('Total no of classes = %10.6g \n',k);
62
63        disp(' ');
64
65        xset('window',1);
66        histplot(k,x);
67        xtitle('histogram','x','rel.f');
68        xset('window',2);
69        xset('mark',-9,2);
70        plot2d(cmark,fcumulrel,-9);
71        plot2d(cmark,fcumulrel,1);
72        xtitle('ogive','x','cum.rel.f');
73  endfunction
```

freqdist