

Scilab Manual for  
Probability Theory and Random Processes  
by Prof Shital Thakkar  
Others  
Dharmsinh Desai University<sup>1</sup>

Solutions provided by  
Prof Shital Thakkar  
Others  
Dharmsinh Desai University

May 11, 2025

<sup>1</sup>Funded by a grant from the National Mission on Education through ICT, <http://spoken-tutorial.org/NMEICT-Intro>. This Scilab Manual and Scilab codes written in it can be downloaded from the "Migrated Labs" section at the website <http://scilab.in>



# Contents

List of Scilab Solutions	3
1 Write a program to find probability of tossing a coin and rolling a die through large no. of experimentation.	5
2 To generate Uniform, Gaussian and Exponential distributed data for given mean and variance.	9
3 Write a program to generate M trials of a random experiment having specific number of outcomes with specified probabilities.	15
4 To find estimated and true mean of Uniform, Gaussian and Exponential distributed data.	18
5 To find density and distribution function of a function of random variable $Y = 2X + 1$ . where X is gaussian R.V.	22
6 Estimate the mean and variance of $Y = 2X + 1$ , where X is a gaussian random variable.	26
7 Plot Joint density and distribution function of sum of two Gaussian random variable ( $Z = X + Y$ ).	28
8 Estimate the mean and variance of a R.V. $Z = X + Y$ . Where X and Y are also random variables.	33
9 Simulation of Central Limit Theorem.	35

# List of Experiments

Solution 1.1	Calculates probability of sum of tossing two dice .	5
Solution 1.2	Finds probability of getting Head when a coin is tossed . . . . .	6
Solution 2.1	Generation of Uniform Data . . . . .	9
Solution 2.2	Gaussian Data Generation . . . . .	11
Solution 2.3	Exponential Data Generation . . . . .	12
Solution 3.1	Random experiment with outputs in specific range	15
Solution 4.1	Comaparison of True and estimated statics of Uniform Data . . . . .	18
Solution 4.2	Comaparison of True and estimated statics of Gaussian Data . . . . .	19
Solution 4.3	Comaparison of True and estimated statics of Exponential Data . . . . .	20
Solution 5.1	Density and Distribution plot generation for one function of Random Variable . . . . .	22
Solution 6.1	Statistics of Function of one random variable . . .	26
Solution 7.1	Joint Density and Distribution of Function of two random variable . . . . .	28
Solution 8.1	Estimation of mean and variance of sum of two random variable . . . . .	33
Solution 9.1	Summation of two random variable leads to Gaussian density function . . . . .	35

# List of Figures

1.1	Finds probability of getting Head when a coin is tossed . . .	7
2.1	Generation of Uniform Data . . . . .	10
2.2	Gaussian Data Generation . . . . .	11
2.3	Exponential Data Generation . . . . .	13
5.1	Density and Distribution plot generation for one function of Random Variable . . . . .	23
7.1	Joint Density and Distribution of Function of two random variable . . . . .	29
7.2	Joint Density and Distribution of Function of two random variable . . . . .	30
9.1	Summation of two random variable leads to Gaussian density function . . . . .	36

## Experiment: 1

Write a program to find probability of tossing a coin and rolling a die through large no. of experimentation.

**Scilab code Solution 1.1** Calculates probability of sum of tossing two dice

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //Program of Tossing two dice and observing
   Probability of sum of their front face.
5 //for e.g. Probability of sum of two dice = 2 is
   1/36 as there are 36 possibilities and sum = 2
   can// occur only one combination that is both
   face = 1
6 clc;
7 clear;
8 clf;
9 N = 10000; // Number of times tossing of die
   performed
10 count = 0; // Counter for counting number of times
```

```

sum of die
11 for i = 1:N
12     y1 = ceil(rand(1)*6); // output of die 1
13     y2 = ceil(rand(1)*6); // output of die 2
14     if((y1+y2) == 3)      // check for sum of front
        face of both die is == 3(change sum and)
15         count = count + 1; //increment the count
        value when sum = 3 occurs
16     end
17     prob1(i) = count/i;    // no. of times sum of
        die = 3/total no. trials
18 end
19 plot(prob1)
20 xlabel('Number of Trials');
21 ylabel('Probability');
22 title('Probability of getting sum of dots on faces
        of a dice to be 3');
23
24
25 //Assignment : Program can be checked for other
        values of sum at line number 10.

```

---

**Scilab code Solution 1.2** Finds probability of getting Head when a coin is tossed

```

1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4
5 //This program find probability of getting Head when
    a coin is tossed.
6 //Probability =  $1/2 = 0.5$  as there are two possible
    outcomes in coin tossing experiment.

```

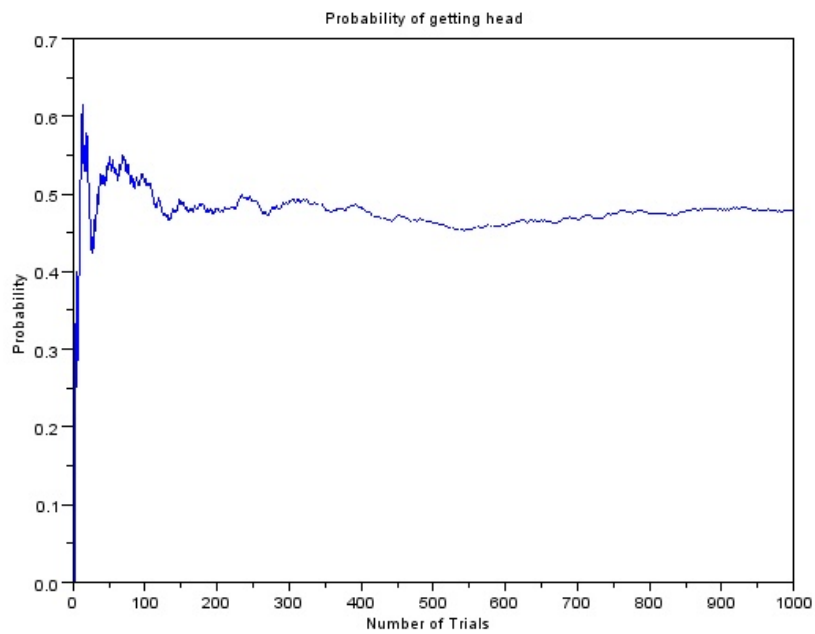


Figure 1.1: Finds probability of getting Head when a coin is tossed



```

7  clc;
8  clear all;
9  clf;
10 a1 = 1000;
11 count1 = 0;
12 for i = 1:a1
13 x = round(rand(1));
14     //round: the elements to nearest integer
15     //rand:returns a pseudorandom, scalar value
        drawn from a uniform
16     //distribution on the unit interval.
17     if(x==1) // HEAD- '1',condition that detects ' 
        HEAD' comes or not
18         count1 = count1 + 1;//increment the count
            value when head occurs
19     end
20     p(i)= count1/i;//probability of head occuring at
        ith trail
21 end
22 plot(1:a1,p)
23 //plot the prob. at ith trail(plots discrete
    sequence)
24 xlabel('Number of Trials');
25 ylabel('Probability');
26 title('Probability of getting head');
27 //Assignment:
28 //1.   perform above experiment with n = 100,1000 .
29 //2.   Extend the above experiment to find
    probability of 3 heads in 4 coin tosses.
30 //     Match the result theoretically.

```

---

## Experiment: 2

To generate Uniform, Gaussian and Exponential distributed data for given mean and variance.

Scilab code Solution 2.1 Generation of Uniform Data

```
1 //To generate Uniform , Gaussian and Exponential
   distributed data.
2 // Operating System : Windows XP or later ,
3 // Scilab           : 5.3.3
4 //NOTE:EXECUTE ONE BY ONE SEGEMENT
5
6 //[1] Uniform Data Generation
7 clc;
8 clear all;
9 //b = input('higher limit of uniform r.v. b = ')//
   Enter higher limit of uniform r.v.
10 //a = input('lower limit of uniform r.v. a =')//
   Enter lower limit of uniform r.v.
```

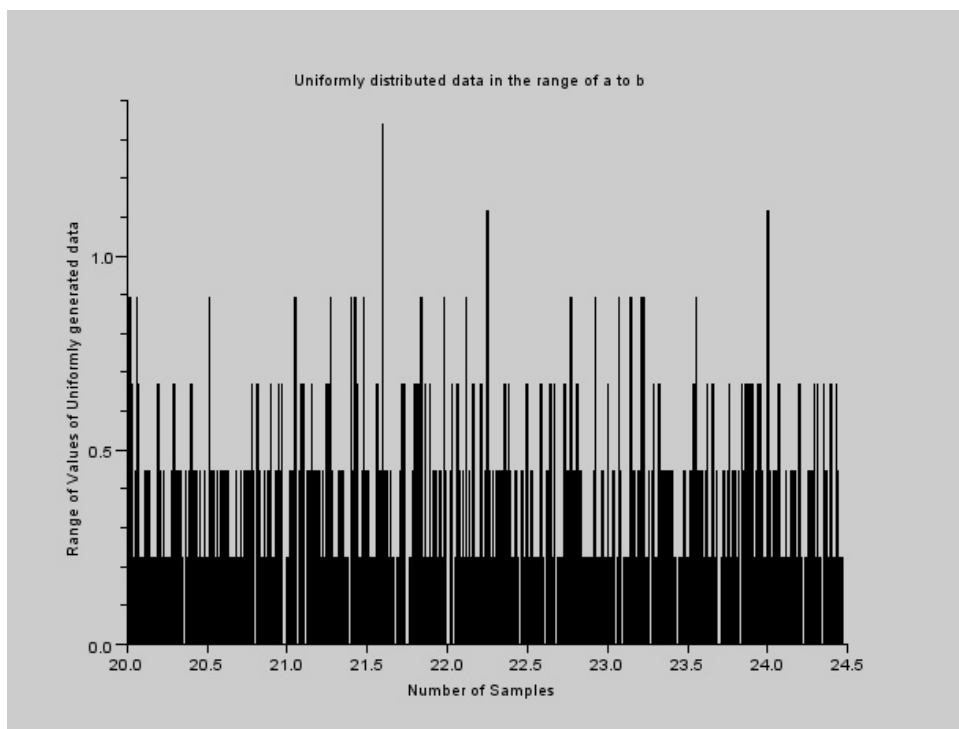


Figure 2.1: Generation of Uniform Data

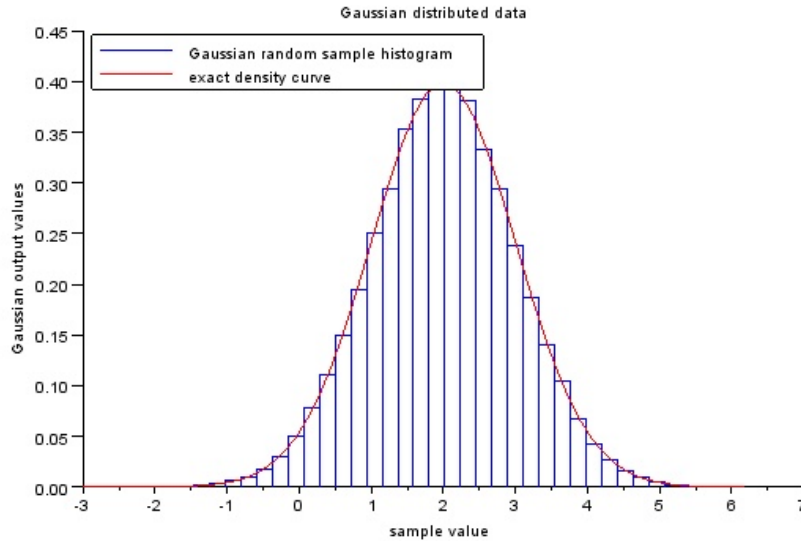


Figure 2.2: Gaussian Data Generation

```

11 clf();
12 b=4; //higher range of uniform data
13 a=2; //lower range of uniform data
14 x=a + (b-a).*rand(1,1000,'uniform'); //uniform data
    generation of length 100
15 histplot(1000,x)//histogram of generated data
16 xlabel('Number of Samples');
17 ylabel('Range of Values of Uniformly generated data'
    );
18 title('Uniformly distributed data in the range of a
    to b');

```

---

## Scilab code Solution 2.2 Gaussian Data Generation

```

1 //To generate Uniform , Gaussian and Exponential
  distributed data.
2 // Operating System : Windows XP or later ,
3 // Scilab           : 5.3.3
4
5 //[2] Expoenential data generation & Mean and
  Variance Calculatcion of Exponential distributed
  data.
6 clc;
7 clear all;
8 clf();
9 //(i) Exponential data generation
10 lambda = 2; //or lambda = input('enter lemda value
    for exponential r.v.') //lemda of exponential data
    ;
11 X = grand(10000,1,"exp", 1/lambda);
12 Xmax = max(X);
13 histplot(40, X, style=2)
14 x = linspace(0,max(Xmax),100)';
15 plot2d(x,lambda*exp(-lambda*x),strf="000",style=5)
16 legend(["exponential random sample histogram" "exact
    density curve"]);
17 xlabel('sample value');
18 ylabel('Exponential output values');
19 title('Exponential distributed data');

```

---

### Scilab code Solution 2.3 Exponential Data Generation

```

1 //To generate Uniform , Gaussian and Exponential
  distributed data.
2 // Operating System : Windows XP or later ,
3 // Scilab           : 5.3.3
4

```

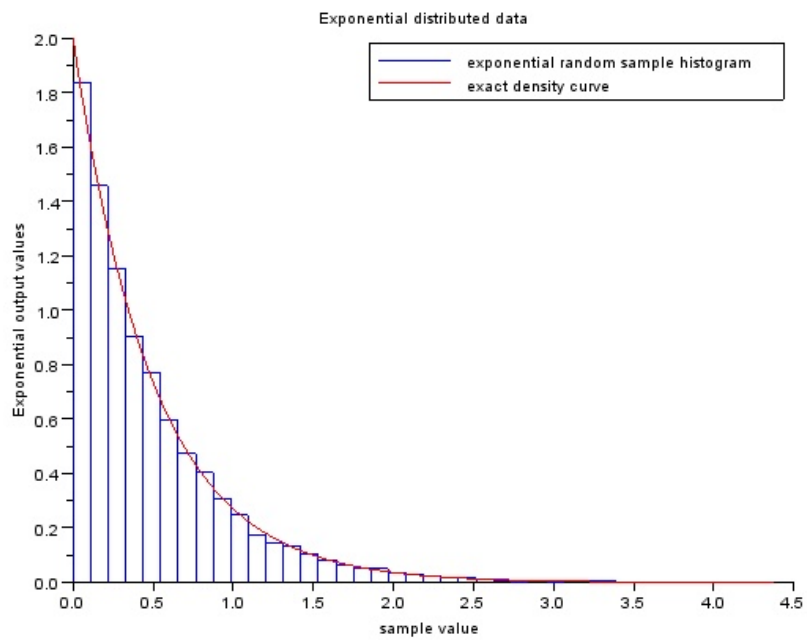


Figure 2.3: Exponential Data Generation

```

5 // [3] Gaussian data generation
6 // Gaussian data generation
7 clc;
8 clear all;
9 clf();
10 //m = input('enter mean value for Gaussian r.v.')
11 //vari = input('enter mean value for Gaussian r.v.')
12 m = 2; //mean value of gaussian data
13 sd = 1; //standard deviation
14 vari = sd^2;
15 X = grand(100000,1,"nor", m,sd);
16 Xmax = max(X);
17 clf()
18 histplot(40, X, style=2)
19 x = linspace(-10,max(Xmax),100)';
20 plot2d(x,(1/(sqrt(2*%pi*vari)))*exp(-0.5*(x-m).^2/
    vari),strf="000",style=5)
21 xlabel('sample value');
22 ylabel('Gaussian output values');
23 title('Gaussian distributed data');
24 legend(["Gaussian random sample histogram" "exact
    density curve"],2);

```

---

## Experiment: 3

Write a program to generate M trials of a random experiment having specific number of outcomes with specified probabilities.

**Scilab code Solution 3.1** Random experiment with outputs in specific range

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //Write a program to generate M trials of a random
   experiment having specific
5 //number of outcomes with specified probabilities.
6 //here No. of trials = 1000,no. outcomes(rv) = 3
   with specied probability entered by user
7 clc;
8 clear all;
9 rand('seed')//check
10 M = 1000; //Number of trials of random experiment
11 outcomes = 3; //Possible number of outcomes of
```



```

    random experiment
12 for i = 1:outcomes-1
13     r(i) = input('enter upper range of probability
        of r.v. (values in the 0<r<1) :')//enter
        values in the 0<r<1
14     if r(i)>1 then
15         error('Enter values in the range 0<r<1')
16     end
17 end
18 x = zeros(M,1);
19 for i = 1:M
20     u = rand(1,1); //random outcome
21     if u <= r(1) then
22         x(i,1)=1; //assign rv value = 1 if u<=r(1)
23     elseif u > r(1) & u<= r(2)
24         x(i,1)=2; //second rv value
25     else
26         x(i,1)=3; //third rv value
27     end
28 end
29 count1=0; count2=0; count3=0;
30 for i=1:1000
31     if x(i,1)==1 then
32         count1 = count1 + 1;
33     elseif x(i,1)== 2
34         count2 = count2 + 1;
35     else
36         count3 = count3 + 1;
37     end
38 end
39 estP1 = count1/M; disp(estP1) //estimated probability
    of generated random variable
40 estP2 = count2/M; disp(estP2) //estimated probability
    of generated random variable
41 estP3 = count3/M; disp(estP3) //estimated probability
    of generated random variable
42
43 //Assignment:

```

```
44 //1. Extend this program for 4 number of random
    variable
45 //2. Extend this program for more number of trials.
    i.e. M = 5000,10000 etc.
```

---

## Experiment: 4

To find estimated and true mean of Uniform, Gaussian and Exponential distributed data.

**Scilab code Solution 4.1** Comparison of True and estimated statistics of Uniform Data

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3 //here we generate uniformly distributed data
  compare its statistics with calculated using
  equation.
4
5
6
7 //[1] Mean and Variance Calculation of Uniformly
  distributed data
8
9 clc;
10 clear all;
11 b=4; //higher range of uniform data
12 a=2; //lower range of uniform data
13 x=a + (b-a).*rand(1,1000); //Uniform data generation
```

```

        using function
14 Uni_true_mean=mean(x)
15 mprintf('Uniform True Mean = %f',Uni_true_mean)
16 Uni_true_var = variance(x)
17 mprintf('\n Uniform True Mean = %f',Uni_true_var)
18 px = 1/(b-a)//pdf calculaion of uniform r.v.
19 m_uniform=integrate('x*px','x',a,b)//mean
        calculaion of uniform r.v.
20 fsq_uniform=integrate('(x^2)*px','x',a,b)//mean
        square value of uniform r.v.
21 var_uniform = fsq_uniform - (m_uniform).^2//variance
        of uniform r.v.
22
23 mprintf('\n Uniform Calculated Mean = %f',m_uniform)
24 mprintf('\n Uniform Calculated Variance = %f',
        var_uniform)

```

---

**Scilab code Solution 4.2** Comaparison of True and estimated statics of Gaussian Data

```

1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //here we generate Gaussian distributed data compare
    its statistics with calculated using equation.
5
6 //[3] Mean & Variance calculation of Gaussian Data
7
8 clc;
9 clear all;
10 m = 2;//mean value of gaussian data
11 sd = 1;//standard deviation
12 vari = sd^2;
13 X = grand(10000,1,"nor", m,sd);//gaussian data
    generation using function

```

```

14 Gaus_true_mean=mean(X)
15 mprintf('Gaussian True Mean = %f',Gaus_true_mean)
16 Gaus_true_var=variance(X)
17 mprintf('\n Gaussian True Variance = %f',
    Gaus_true_var)
18
19 //Mean and variance calculaion using formula
20 gs_mean=integrate('x*(1/sqrt(2*%pi*vari)*exp(-(x-m)
    ^2/(2*vari)))','x',0,100)
21 gs_fsq=integrate('(x^2)*(1/sqrt(2*%pi*vari)*exp(-(x-
    m)^2/(2*vari)))','x',0,100);
22 gs_var = gs_fsq - gs_mean.^2;
23 mprintf('\n Gaussian Calculated Mean = %f',gs_mean)
24 mprintf('\n Gaussian Calculated Variance = %f',
    gs_var)

```

---

**Scilab code Solution 4.3** Comaparison of True and estimated statics of Exponential Data

```

1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //here we generate Exponentially distributed data
    compare its statistics with calculated using
    equation.
5
6 //[2] Mean and Variance Calculation of Exponential
    data
7 clc;
8 clear all;
9 lambda = 2; //or lambda = input('enter lemda value
    for exponential r.v.') //lemda of exponential data
    ;
10 X = grand(1000,1,"exp", 1/lambda); //Exponential data
    generation using function

```

```

11 Expo_true_mean=mean(X)
12 mprintf('Exponential True Mean = %f',Expo_true_mean)
13 Expo_true_var=variance(X)
14 mprintf('\n Exponential True Variance = %f',
    Expo_true_var)
15
16 //Mean and variance calculaion using formula
17 ex_mean=integrate('x*(lambda*exp(-lambda*x))','x',
    ,0,100)
18 ex_fsq=integrate('(x^2)*(lambda*exp(-lambda*x))','x',
    ,0,100);
19 ex_var = ex_fsq - ex_mean.^2
20 mprintf('\n Exponential Calculated Mean = %f',
    ex_mean)
21 mprintf('\n Exponential Calculated Variance = %f',
    ex_var)

```

---

## Experiment: 5

To find density and distribution function of a function of random variable  $Y = 2X + 1$ . where  $X$  is gaussian R.V.

**Scilab code Solution 5.1** Density and Distribution plot generation for one function of Random Variable

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //To find density(pdf) and distribution(cdf)
   function of a function of random variable
5 //Y = 2X + 1(having form Y = aX + b). where X is
   gaussian R.V.
6
7 clc;
8 clear all;
9 clf();
10 mean_x = 1; //mean value of gaussian data
```

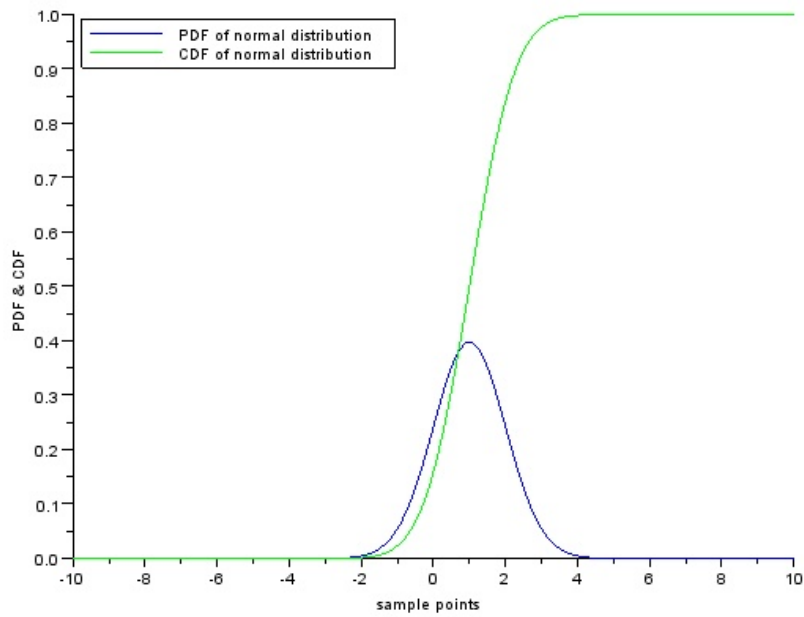


Figure 5.1: Density and Distribution plot generation for one function of Random Variable



```

11 sd_x = 1; //standard deviation
12 vari_x = sd_x.^2;
13 lgd = [];
14 //PDF and CDF of Gaussian Random Variable X
15 x = linspace(-10,10,100);
16 plot2d(x,((1/(sqrt(2*%pi*vari_x)))*exp(-0.5*(x-
    mean_x).^2/vari_x)),2); //plots pdf of X
17 set(gca(),"auto_clear","off")
18 plot2d(x,cdfnor("PQ",x,mean_x*ones(x),sd_x*ones(x))
    ,3); //cdf of gaussian RV X
19 set(gca(),"auto_clear","on")
20 xlabel('sample points');
21 ylabel('PDF & CDF');
22 //title('density and distribution function for
    Gaussian function');
23 legend(['PDF of normal distribution'; 'CDF of normal
    distribution'],2);
24
25 //PDF and CDF of Y = aX + b where a = 2, b = 1
26 a = 2;
27 b = 1;
28 y = a*x+b; //Function of One Random Variable
29 mean_y=a*mean_x+b;
30 vari_y=(a*sd_x).^2;
31 figure(2,"BackgroundColor",[1,1,1]);
32 plot2d(y,((1/(sqrt(2*%pi*vari_y*a.^2)))*exp(-0.5*(y-
    mean_y).^2/vari_y)),2); //pdf of Y
33 set(gca(),"auto_clear","off")
34 plot2d(x,cdfnor("PQ",y,(a*mean_x+b)*ones(x),(a*sd_x)
    *ones(x)),3); //cdf of y
35 set(gca(),"auto_clear","on")
36 xlabel('sample points');
37 ylabel('PDF & CDF of Y = 2X + 1');
38 legend(['PDF of Y = 2X + 1'; 'CDF of Y = 2X + 1'],2);
39
40
41 //Assignment :
42 //1. Perform the operation for function Y = 5X + 1.

```

43 //2. Generate pdf and cdf of nonlinear function  
between Y and X.

---

## Experiment: 6

Estimate the mean and variance of  $Y = 2X + 1$ , where  $X$  is a gaussian random variable.

Scilab code **Solution 6.1** Statistics of Function of one random variable

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //True and Estimated value of mean and variance of
   function of one random
5 //variable having form of  $Y = aX + b$ .
6 clc;
7 clear all;
8 rand('seed',getdate('s'))
9 m = 0; //mean of random variable x
10 vari = 1; //variance of random variable x
11 m_est = 0;
12 var_est = 0;
13 for i = 1:1000
14     y(i,1)= 1 +2*rand(1,1,"normal");//  $Y = 2X + 1$ 
```

```

        where x is gaussian data
15     m_est = m_est + ((1/1000)*y(i)); //estimation by
        averaging
16     var_est = var_est + ((1/1000)*(y(i)-m_est)^2);
17 end
18 printf('Estimated mean of Y(=2X + 1) is: Est_mean=%f
        ',m_est)
19 printf('\n Estimated variance of Y) is: Est_variance
        =%f',var_est)
20 //Calculation of true mean of Y
21 y_mean=integrate(' (2*x+1)*(1/sqrt(2*%pi*vari)*exp(-(
        x-m)^2/(2*vari))) ', 'x', -100,100);
22 printf('\n True mean of Y(=2X + 1) is: True_mean=%f'
        ,y_mean)
23 //Calculation of true variance of Y
24 //for a function like Y = aX + b the variance of Y
        is a^2*Variance of X.
25 gs_mean=integrate('x*(1/sqrt(2*%pi*vari)*exp(-(x-m)
        ^2/(2*vari))) ', 'x', -50,100);
26 gs_fsq=integrate('((x^2)*(1/sqrt(2*%pi*vari)*exp(-(x
        -m)^2/(2*vari)))) ', 'x', -50,100);
27 gs_var = gs_fsq - (gs_mean).^2;
28 var_y = 2^2*gs_var; //here a = 2
29 printf('\n True variance of Y(=2X + 1) is:
        True_variance=%f',var_y)
30
31 //Expectation of Y is E(Y)=E(2X+1)=2E(X)+1. That's
        why answer is 1.
32 //True variance of Y in this format is equal to a^2*
        variance of X.
33 //Assignment:
34 //1. Assume X is uniform random variable between a
        to b. find mean and variance.

```

---

## Experiment: 7

Plot Joint density and distribution function of sum of two Gaussian random variable ( $Z = X + Y$ ).

**Scilab code Solution 7.1** Joint Density and Distribution of Function of two random variable

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4
5 //Plot Joint density and distribution function of
   sum of two Gaussian random variable ( $Z = X + Y$ ).
6 clc;
7 clear all;
8 clf();
9
```

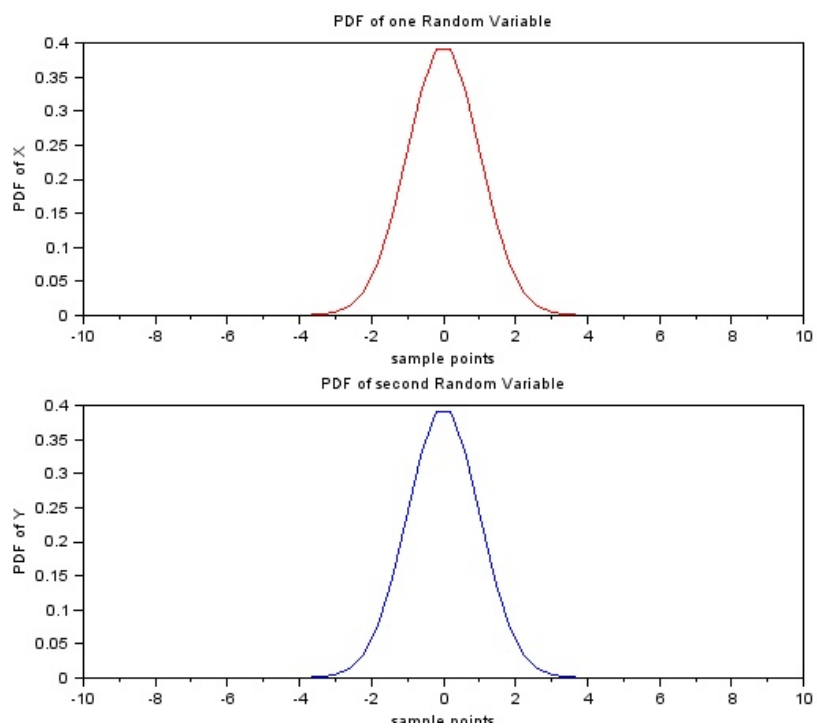


Figure 7.1: Joint Density and Distribution of Function of two random variable

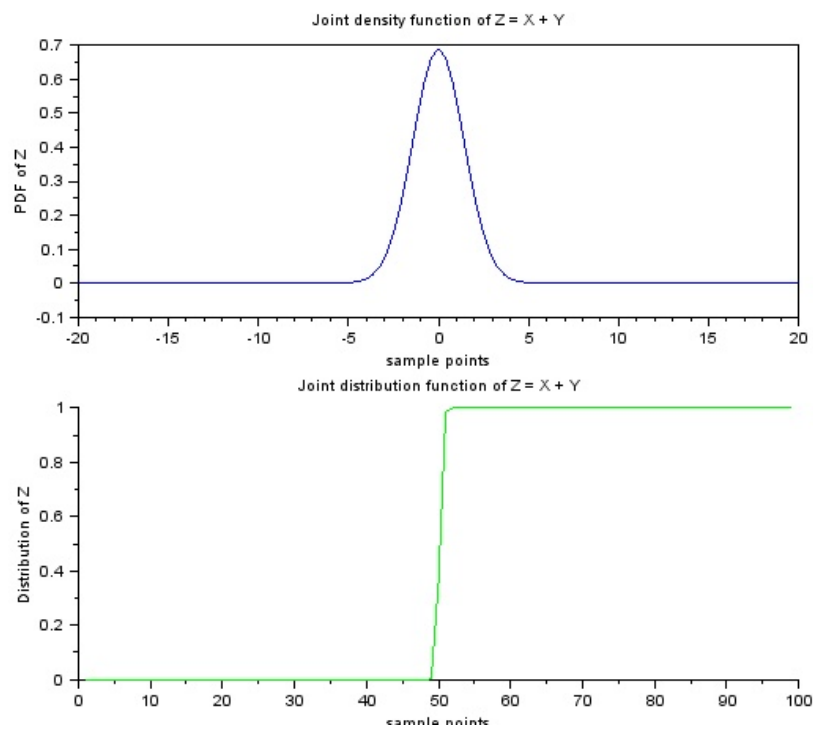


Figure 7.2: Joint Density and Distribution of Function of two random variable

```

10 //PDF of Gaussian Random Variable X
11 mean_x = 0; //mean of first gaussian RV
12 sd_x = 1; //standard deviation of first gaussian RV
13 vari_x = sd_x.^2;
14
15 x = linspace(-10,10,50); //generating linearly spaced
    data as Random output
16 X = ((1/(sqrt(2*%pi*vari_x)))*exp(-0.5.*(x-mean_x)
    .^2/vari_x)); //finding gaussian pdf of above data
17 subplot(2,1,1);
18 plot(x,X, 'r')
19 xlabel('sample points');
20 ylabel('PDF of X');
21 title('PDF of one Random Variable')
22
23
24 //PDF of Gaussian Random Variable Y
25 mean_y = 0;
26 sd_y = 1;
27 vari_y = sd_y.^2;
28
29 y = linspace(-10,10,50);
30 Y = ((1/(sqrt(2*%pi*vari_y)))*exp(-0.5.*(y-mean_y)
    .^2/vari_y));
31 subplot(2,1,2);
32 plot(y,Y, 'b')
33 xlabel('sample points');
34 ylabel('PDF of Y');
35 title('PDF of second Random Variable')
36
37 // Joint pdf of sum of random variable X & Y
38 // When two IID random variable are summen up, their
    Joint PDF is convolution between individual pdfs
    of Random variables
39 z = convol(X,Y);
40 figure(2," BackgroundColor",[1,1,1]);
41 subplot(2,1,1); plot(linspace(-20,20,99),z) // Joint
    PDF

```



```

42 xlabel('sample points');
43 ylabel('PDF of Z');
44 title('Joint density function of  $Z = X + Y$ ')
45 P = cdfnor("PQ", linspace(-20,20,99), mean(z)*ones(z),
         sqrt(variance(z))*ones(z));
46 set(gca(),"auto_clear","off")
47 subplot(2,1,2); plot2d(1:length(P),P,3);
48 xlabel('sample points');
49 ylabel('Distribution of Z');
50 title('Joint distribution function of  $Z = X + Y$ ')
51 set(gca(),"auto_clear","on")
52 // Assignment:
53 // Change mean and variance of individual random
   variable X and Y and observe output

```

---

## Experiment: 8

Estimate the mean and variance of a R.V.  $Z = X + Y$ . Where  $X$  and  $Y$  are also random variables.

**Scilab code Solution 8.1** Estimation of mean and variance of sum of two random variable

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4 //Concept : Estimation of mean and variance of sum of
      two random variable  $Z = X + Y$ , where  $X$  and  $Y$  are
      random variable .
5 // Above concept is explained with example as
      follows .
6 //Example: A large circular dartboard is set up with
      a "bullseye" at the center of the circle ,which
      is at the coordinate (0,0). A dart is thrown at
      the center but lands at (X,Y) are two different
      Gaussian random variables . What is average
      distance of the dart from the bullseye? What is
```

```

    variance of data?
7 //Distance from center is given as sqrt(X^2+Y^2)
8
9 clc;
10 clear all;
11 rand('seed',0)//setting seed of random generator to
    zero
12 m_est = 0;
13 for i = 1:1000
14     R(i,1)=sqrt(rand(1,1,'normal')^2+rand(1,1,'
        normal')^2);//calculation of distance from
        center
15     m_est=m_est+(1/1000)*R(i);//estimation of mean
        from data
16 end
17 m_est
18 mprintf('Mean of Sum of Two Random variable that is
    Mean of Z = %f',m_est)
19 v_est = variance(R)//variance calculation
20 mprintf('\\n Variance of Sum of Two Random variable
    that is Mean of Z = %f',v_est)

```

---

## Experiment: 9

# Simulation of Central Limit Theorem.

**Scilab code Solution 9.1** Summation of two random variable leads to Gaussian density function

```
1 // Operating System : Windows XP or later ,
2 // Scilab           : 5.3.3
3
4
5 //Simulation of Central Limit Theorem.
6 //(Which says that if we keep on adding independant
   Random Variables then it density function
7 //approches to gaussian distribution)
8 //here two uniform RVs are added.
9
10 clc;
11 clear all;
12 clf();
13 n = 0:0.01:1;
14 x = zeros(length(n),1);
15 i = 1:50; //length of Uniform rv 1
```

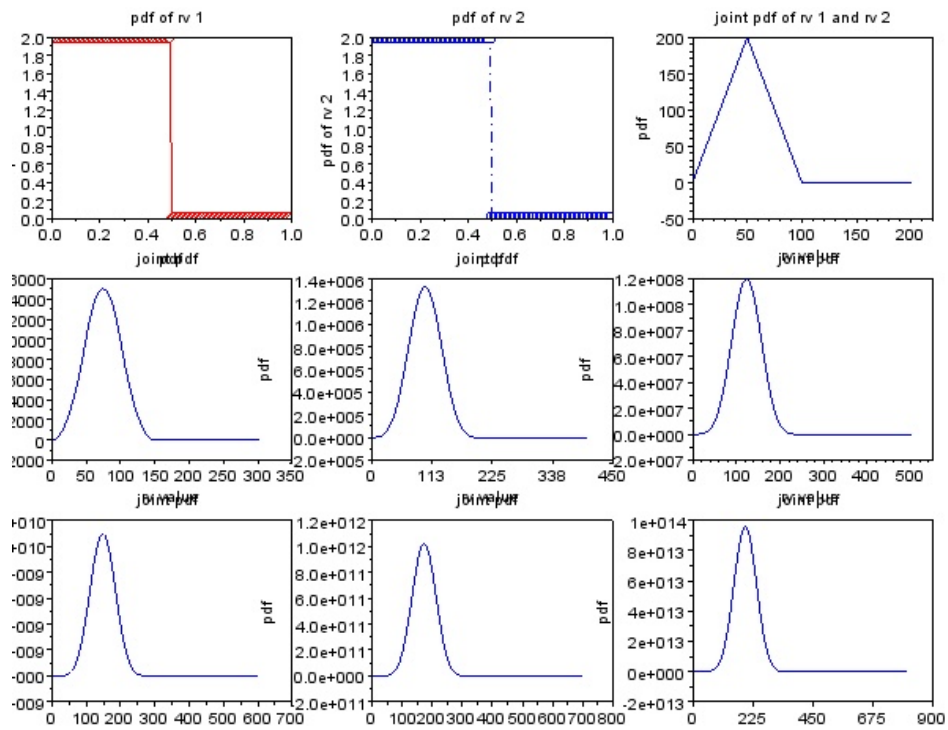


Figure 9.1: Summation of two random variable leads to Gaussian density function

```

16 x(i)= 2;
17 subplot(3,3,1);
18 plot(n,x,'r-d')
19 xlabel('pdf');ylabel('pdf of rv 1')
20 title('pdf of rv 1 ');
21
22 y = zeros(length(n),1);
23 j = 1:50;
24 y(j)= 1*2;//length of Uniform rv 2
25 subplot(3,3,2);
26 plot(n,y,'bo-.')
27 xlabel('pdf');
28 ylabel('pdf of rv 2')
29 title('pdf of rv 2');
30
31 z1 = convol(x,y);
32 subplot(3,3,3)
33 //When two independent RVs are added their joint
    density is convolution of marginal density
34 plot(z1)
35 xlabel('rv value');
36 ylabel('pdf')
37 title('joint pdf of rv 1 and rv 2');
38
39 for i = 4:9 // adding rv 9 times
40     subplot(3,3,i)
41     z1 = convol(z1,y);
42     plot(z1)
43     xlabel('rv value');
44     ylabel('pdf')
45     title('joint pdf');
46 end

```

---