**BLOOD PRESSURE**

**AIM:**

In a group of 40 people, 10 are healthy and every person the of the remaining 30 has either high blood pressure, a high level of cholesterol or both. If 15 have high blood pressure and 25 have high level of cholesterol,

a) how many people have high blood pressure and a high level of cholesterol?

If a person is selected randomly from this group, what is the probability that he/she

b) has high blood pressure (event A)?

c) has high level of cholesterol (event B)?

d) has high blood pressure and high level of cholesterol (event A and B)?

e) has either high blood pressure or high level of cholesterol (event A or B)?

f) Use the above to check the probability formula: P(A or B) = P(A) + P(B) - P(A and B).

**CODE:**

clc;

n=30;

a=15;

b=25;

//P(A AND B)

a\_and\_b=a+b-n;

mprintf("\n(A AND B) = %d",a\_and\_b)

//P(A)

p\_a=a/n;

mprintf("\nP(A) = %f",p\_a)

//P(B)

p\_b=b/n;

mprintf("\nP(B) = %f",p\_b)

//P(A AND B)

p\_and\_b=a\_and\_b/n;

mprintf("\nP(A AND B) = %f",p\_and\_b)

//P(A OR B)

p\_or\_b=p\_a+p\_b-p\_and\_b;

mprintf("\nP(A or B) = %d",p\_or\_b)

//P(A OR B) = P(A) + P(B) - P(A AND B)

k=p\_a+p\_b-p\_and\_b

if k==p\_or\_b then mprintf("\n\nk=%d,p\_or\_b=%d,\n%d=%d\nP(A OR B) = P(A) + P(B) - P(A AND B)",k,p\_or\_b,k,p\_or\_b)

end

**OUTPUT:**

(A AND B) = 10

P(A) = 0.500000

P(B) = 0.833333

P(A AND B) = 0.333333

P(A or B) = 1

k=1,p\_or\_b=1,

1=1

P(A OR B) = P(A) + P(B) - P(A AND B)

**COMMITTEE**

**AIM:**

A committee of 5 people is to be formed randomly from a group of 10 women and 6 men. Find theprobability that the committee has

a) 3 women and 2 men.

b) 4 women and 1 men.

c) 5 women.

d) at least 3 women.

**CODE:**

clc;

n=16;

r=5;

men=6;

women=10;

//3 WOMEN AND 2 MEN

p=(nchoosek(women,3)\*nchoosek(men,2))/nchoosek(n,r)

mprintf("\n\n2 WOMEN AND 3 MEN = %f",p);

//4 WOMEN AND 1 MEN

p=(nchoosek(women,4)\*nchoosek(men,1))/nchoosek(n,r)

mprintf("\n\n4 WOMEN AND 1 MEN = %f",p);

//ALL ARE WOMEN

p=(nchoosek(women,5)\*nchoosek(men,0))/nchoosek(n,r);

mprintf("\n\nALL ARE WOMEN = %f",p);

//ATLEAST 3 WOMEN

p=0;

for i=3:5

for j=0:2

if(i+j==5)

temp=(nchoosek(women,i)\*nchoosek(men,j))/nchoosek(n,r);

end

end

p=p+temp;

end

mprintf("\n\nATLEAST 3 WOMENS = %f",p);

**OUTPUT:**

2 WOMEN AND 3 MEN = 0.412088

4 WOMEN AND 1 MEN = 0.288462

ALL ARE WOMEN = 0.057692

ATLEAST 3 WOMENS = 0.758242

**STUDENT’S INTERNET**

**AIM:**

In a school, 60% of pupils have access to the internet at home. A group of 8 students is chosen at

random. Find the probability that

1. exactly 5 have access to the internet.
2. at least 6 students have access to the internet.

**CODE:**

clc;

p=0.6

q=1-p;

n=8;

//EXACTLY 5 STUDENTS

r=5;

mprintf("EXACTILY 5 STUDENTS = %f",nchoosek(n,r)\*(p\*\*r)\*(q\*\*(n-r)))

//ATLEAST 6 STUDENTS

temp=0;

for r=6:8

temp=temp+nchoosek(n,r)\*(p\*\*r)\*(q\*\*(n-r));

end

mprintf("\nATLEAST 6 STUDENTS = %f",temp);

**OUTPUT:**

EXACTILY 5 STUDENTS = 0.278692

ATLEAST 6 STUDENTS = 0.315395

**IRON & STEEL**

**AIM:**

In a certain country last year a total of 500 million tons of trash was recycled. The chart below shows the

distribution, in millions of tons, for the different types of trashes.

a) How many tons of Iron/Steel was recycled?

b) What percent of the recycled trash was glass

**CODE:**

clc;

n=500;

p=170;

al=90;

gl=60;

ot=50;

//IRON/STEEL

i\_s=n-(p+al+gl+ot);

mprintf("\nIRON\STEEL RECYCLED = %d",i\_s);

mprintf("\nGLASS WAS RECYCELDE BY %f",(60/500)\*100);

**OUTPUT:**

IRONSTEEL RECYCLED = 130

GLASS WAS RECYCELDE BY 12.000000

**CHI-SQUARE**

**AIM:**

A random variable, C, has a Chi-square distribution with 9 degrees of freedoms. Use SCILAB’s cdfchi function to compute

i)Pr(C < 5.1) ii) Pr( C > 6.2)       iii) Pr (4.1< C < 7.5)

**CODE:**

x=5.1;

f=9;

[P,Q]=cdfchi("PQ",x,f)

disp("cdfchi(x<5.1):",[P,Q])

x=6.2;

f=9;

[P,Q]=cdfchi("PQ",x,f)

disp("cdfchi(x>6.2):",[P,Q])

x=3.4;

f=9;

[P,Q]=cdfchi("PQ",x,f)

disp("cdfchi(x<7.5):",[P,Q])

**OUTPUT:**

"cdfchi(x<5.1):"

0.1744947 0.8255053

"cdfchi(x>6.2):"

0.2802534 0.7197466

"cdfchi(x<7.5):"

0.0536923 0.9463077

**NEWTON’S METHOD**

**AIM:**

Use newton’s method to determine X2X2 for the given function and the given value of X0X0

1. F(x)=X3-7X2+8X-3f(x)= X3-7X2+8X-3,X0=5,X0=5

**CODE:**

import math as m

def func( x ):

return x\*x\*x-7\*x\*x+8\*x-3

def derivFunc( x ):

return 3\*x\*x-14\*x+8

def newtonRaphson( x ):

h = func(x) / derivFunc(x)

while abs(h) >= 0.0001:

h = func(x)/derivFunc(x)

x = x - h

print("The value of the root is : ","%.4f"% x)

x0 =5

newtonRaphson(x0)

**OUTPUT:**

The value of the root is : 5.6858

**NEWTON’S METHOD**

**AIM:**

Use newton’s method to determine X2X2 for the given function and the given value of X0X0

1. F(X)=xcos(x)—x2f(x)=xcosx-x2,x0=1 x0=1

**CODE:**

**OUTPUT:**

**NEWTON’S METHOD**

**AIM:**

Use newton’s method to find all the roots of the given equation accurate to 6 decimal places, that lies in the given interval.

1. X4-5X3+9X+3=0x4-5X3+9X+3=0 in [4,6][4,6]

**CODE:**

def func( x ):

return x\*x\*x\*x -5\*x\*x\*x+9\*x+3

def derivFunc( x ):

return 4\*x\*x\*x-15\*x\*x+9

def newtonRaphson( x ):

h = func(x) / derivFunc(x)

while abs(h) >= 0.0001:

h = func(x)/derivFunc(x)

x = x - h

print("The value of the root is : ","%.4f"% x)

x0 =5

newtonRaphson(x0)

**OUTPUT:**

The value of the root is: 4.5289

**NEWTON’S METHOD**

**AIM:**

Use newton’s method to find all the roots of the given equation accurate to 6 decimal places, that lies in the given interval.

1. 2X2+5=eX2X2+5=ex in [3,4] [3,4]

**CODE:**

import math as m

def func( x ):

return 2\*x\*x+5-m.exp(x)

def derivFunc( x ):

return 4\*x-m.exp(x)

def newtonRaphson( x ):

h = func(x) / derivFunc(x)

while abs(h) >= 0.0001:

h = func(x)/derivFunc(x)

x = x - h

print("The value of the root is : ","%.4f"% x)

x0 =3.5

newtonRaphson(x0)

**OUTPUT:**

The value of the root is : 3.2756

**NEWTON’S METHOD**

**AIM:**

Use newton’s method to find all the roots of the given equation accurate to 6 decimal places

1. X3-X2-15X+1 = 0X3-X2-15X+1=0

**CODE:**

def func( x ):

return x\*x\*x-x\*x+15\*x+1

def derivFunc( x ):

return 3\*x\*x-2\*x+15

def newtonRaphson( x ):

h = func(x) / derivFunc(x)

while abs(h) >= 0.0001:

h = func(x)/derivFunc(x)

x = x - h

print("The value of the root is : ","%.4f"% x)

x0 =-0.5

newtonRaphson(x0)

**OUTPUT:**

The value of the root is : -0.0664

**NEWTON’S METHOD**

**AIM:**

Use newton’s method to find all the roots of the given equation accurate to 6 decimal places

1. 2-X2=sin(X)

**CODE:**

import math as m

def func( x ):

return 2 - x\*x - m.sin(x)

def derivFunc( x ):

return -2 \* x -m.cos(x)

def newtonRaphson( x ):

h = func(x) / derivFunc(x)

while abs(h) >= 0.0001:

h = func(x)/derivFunc(x)

x = x - h

print("The value of the root is : ","%.4f"% x)

x0 =1.5

newtonRaphson(x0)

**OUTPUT:**

The value of the root is : 1.0615

**LAGRANGE’S INTERPOLATION**

**AIM:**

1. Find Solution using Lagrange’s Interpolation formula

|  |  |
| --- | --- |
| x | F(x) |
| 300 | 2.4771 |
| 304 | 2.4829 |
| 305 | 2.4843 |
| 307 | 2.4871 |

X=301

**CODE:**

def interpolation(d, x):

output = d[0][1] + (x - d[0][0]) \* ((d[1][1] - d[0][1])/(d[1][0] - d[0][0]))

return output

data=[[300,2.4771],[304,2.4829]]

x=301

print("x=301=>f(x)=".format(x),interpolation(data,x))

**OUTPUT:**

x=301=>f(x)= 2.4785500000000003

**LAGRANGE’S INTERPOLATION**

**AIM:**

Use Lagrange’s interpolation formula to find y(10) using the following table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 5 | 6 | 9 | 11 |
| Y | 12 | 13 | 14 | 16 |

**CODE:**

def interpolation (d, x):

output = d[0][1] + (x - d[0][0]) \* ((d[1][1] - d[0][1])/(d[1][0] - d[0][0]))

return output

data=[[9,14],[11,16]]

x=10

print("x=10=>y=".format(x),interpolation(data, x))

**OUTPUT:**

x=10=>y= 15.0