Aim : To fitting a linear curve data using the least square fitting method.

Theory:

Least Square Fitting method:

Its a mathematical technique that allows the analyst to determine the best way of fitting a curve on top of a chart of data points . And alternatively we can say that we are finding the value of the parameter of the equation that fitting the data when data have error in it.

Linear Curve fitting :

In linear curve fitting there is only two parameters those value we need to find , to plot the curve using the data. And those parameters are the slope and the intercept of the line. And this can be done by defining a square error function *E such that*

$$E(c,m)=\sum_{i=1}^{n}[y_i-(mx_i+c)]^2.$$ Here *m* is slope and *c* is the intercept

This is the error in the perfect line and in the data. And we can find the value of the *m* and *c* by making *E* small as possible and this can be done if we take partial derivative of *E* w.r.t *m* and *c* and putting it equal to zero.

$$\frac{\partial E}{\partial c}=0 \quad \text{and} \quad \frac{\partial E}{\partial m}=0,$$

$$0=\frac{\partial}{\partial c}\sum_{i=1}^{n}[y_i-(mx_i+c)]^2=2\sum_{i=1}^{n}[y_i-mx_i-c](-1) \text{ and } 0=\frac{\partial}{\partial m}\sum_{i=1}^{n}[y_i-(mx_i+c)]^2=2\sum_{i=1}^{n}[y_i-mx_i-c](-x_i)$$

by simplifying these we get these equations

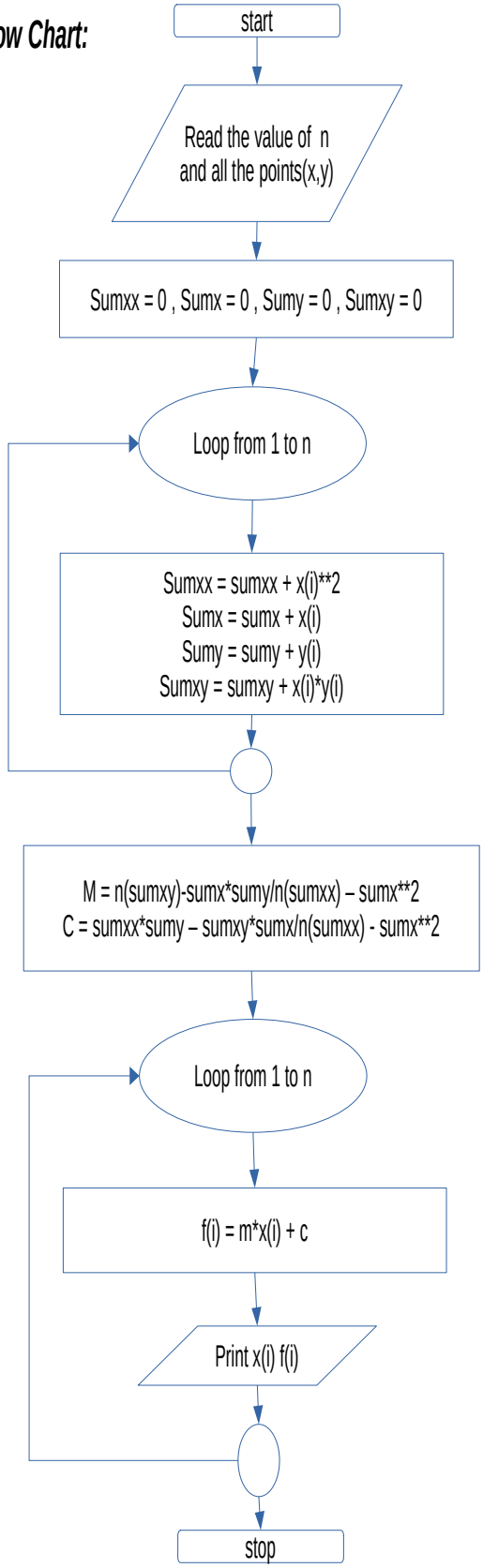
$$cn+m\sum_{i=1}^{n}x_i=\sum_{i=1}^{n}y_i \text{ and } c\sum_{i=1}^{n}x_i+m\sum_{i=1}^{n}x_i^2=\sum_{i=1}^{n}x_iy_i$$

And by solving the above equations , we get value of *m* and *c* and these are

$$c = \frac{\sum_{i=1}^{n}x_i^2\sum_{i=1}^{n}y_i-\sum_{i=1}^{n}x_iy_i\sum_{i=1}^{n}x_i}{n\sum_{i=1}^{n}x_i^2-\left(\sum_{i=1}^{n}x_i\right)^2} \quad \text{and} \quad m = \frac{m\sum_{i=1}^{n}x_iy_i-\sum_{i=1}^{n}x_i\sum_{i=1}^{n}y_i}{n\sum_{i=1}^{n}x_i^2-\left(\sum_{i=1}^{n}x_i\right)^2}$$

And these are are the solution for slope *m* and intercept *c* and in FORTRAN we need to run a loop for find the value of all the summation in equation of *c* and *m* and then use the value of *x* , *m* and *c* fitting the new curve or line that satisfied the data points at some extend.

Flow Chart:


start
Read the value of n
and all the points(x,y)
Sumxx = 0 , Sumx = 0 , Sumy = 0 , Sumxy = 0
Loop from 1 to n
Sumxx = sumxx + x(i)**2
Sumx = sumx + x(i)
Sumy = sumy + y(i)
Sumxy = sumxy + x(i)*y(i)
M = n(sumxy)-sumx*sumy/n(sumxx) – sumx**2
C = sumxx*sumy – sumxy*sumx/n(sumxx) - sumx**2
Loop from 1 to n
f(i) = m*x(i) + c
Print x(i) f(i)
stop

Program in FORTRAN 95

```
program lsf

   implicit none

   ! declaration of the variables

   real , dimension(:) , allocatable :: x,y,f
   real :: sumx , sumy , sumxx , sumxy , c , m
   integer ::  i ,n
   logical :: f1 , f2

   ! Inzillizating the value of the variable

   sumx = 0
   sumy = 0
   sumxx = 0
   sumxy = 0

   ! check exiting file and open it and if file doesn't exit then create the files

   inquire(file="allpoints.dat",exist=f1)     ! for checking the existance of the file
   inquire(file="fittedpoints.dat",exist=f2)
   if (f1) then
      open(1,file="allpoints.dat",status="replace")
   else
      open(1,file="allpoints.dat",status="new",action="write")
   endif
   if (f1) then
      open(2,file="fittedpoints.dat",status="replace")
   else
      open(2,file="fittedpoints.dat",status="new",action="write")
   endif

   ! getting number  of the points given

   print *, "Enter the number of the entry required :: "
   read *, n

   ! allocate the array for store the value of the x and y coordinates and getting from the user

   allocate(x(n))
   allocate(y(n))

   print *, "Enter the value of x and y :: "
   print *, "x      y"
   read *, (x(i),y(i),i=1,n)

   ! find the required terms with summations over the input
```

```
    do i = 1,n
       write(1,*)x(i),y(i)
       sumx = sumx + x(i)
       sumy = sumy + y(i)
       sumxx = sumxx + (x(i))**2
       sumxy = sumxy + x(i)*y(i)
    enddo

    ! deallocate the arrays to free the memory

    deallocate(y)

    ! calculating the intersection and the slope of the line


    c = (sumxx*sumy - sumxy*sumx)/(n*(sumxx)-(sumx)**2)
    m = (n*sumxy - sumx*sumy)/(n*sumxx - (sumx)**2)

    ! storing the fitted points

    allocate(f(n)) ! allocating the vector for fitting points

    ! giving the output and fitting the line .

    print *, "The value of slope is ",m," and the value of y axis intersection is ",c
    print *, "The equation of the line is :: y = ",m,"x +",c
    print *, "       x            fitted points "

    do i = 1,n ! calculated the fitting points
       f(i) = m*x(i) + c
       write(2,*)x(i),f(i)
       print *, x(i),f(i)
    enddo

    deallocate(f)
    deallocate(x)

    close(1)
    close(2)
    stop
end program
```

Output of Program:

```
 Enter the number of the entry required ::
10
 Enter the value of x and y ::
 x      y
1       1.3
2       3.5
3       4.2
4       5
5       7
6       8.8
7       10.1
8       12.5
9       13
10      15.6
 The value of slope is    1.53818178      and the value of y axis intersection is  -0.360002369
 The equation of the line is :: y =    1.53818178     x + -0.360002369
       x           fitted points
  1.00000000      1.17817938
  2.00000000      2.71636128
  3.00000000      4.25454283
  4.00000000      5.79272461
  5.00000000      7.33090639
  6.00000000      8.86908817
  7.00000000      10.4072704
  8.00000000      11.9454517
  9.00000000      13.4836330
  10.0000000      15.0218153
```

Result in Graph :