** INTERPOLATION **

----

* INTRODUCTION :

THE GENERAL PROBLEM OF INTERPOLATION CONSISTS OF REPRESENTING A FUCTION WITH THE AID OF GIVEN VALUES WHICH THIS FUNCTION TAKES FOR DEFINITE VALUES OF THE INDEPENDENT VARIABLES .

SUPPOSE y = f(X) BE A FUNCTION WHICH TAKES VALUES y1,y2,y3,.......,yn FOR VALUES x1,x2,x3,.......,xn OF THE INDEPENDENT VARIABLE x,& FURTHER SUPPOSE THAT @(x) REPRESENTS AN ARBITRARY FUNCTION CONSTRUCTED IN A WAY SUCH THAT IT TAKES THE VALUES AS f(x) FOR THE VALUES x1,x2,x3,......,xn.THEN IF f(x) IS REPLACED BY @(x) OVER A GIVEN INTERVAL , THE PROCESS CONSTITUTES INTER- -POLATION, AND THE FUNCTION @(x) IS A FORMULA OF INTERPOLATION.

THE FUNCTION @(x) CAN TAKE A VARIETY OF FORMS . WHEN @(x) IS A POLYNOMIAL THE PROCESS OF REPRESENTING f(x) BY @(x) IS CALLED "PARABOLIC OR POLYNO- -MIAL INTERPOLATION" ; AND WHEN @(x) IS A FINITE TRIGONOMETRIC SERIES,THE PROCESS IS KNOWN AS "TRIGONOMETRIC INTERPOLATION" .

SIMILARLY,@(x) MAY BE A SERIES OF EXPONINTIAL FUNCTIONS,LEGENDRE POLYNO- -MIALS,BESSEL FUNCTIONS,etc. IN PRACTICAL PROBLEM WE ALWAYS CHOOSE @(x) TO BE THE SIMPLEST FUNCTION WHICH REPRESENTS THE GIVEN FUNCTION OVER THE GIVEN INTERVAL IN QUESTION. SINCE POLYNOMIAL ARE THE SIMPLEST FUNCTIONS, WE USUALLY TAKE A POLYNOMIAL FOR @(x),AND NEARLY ALL THE STANARD FORMULAS OF INTERPOLATION ARE POLYNOMIAL FORMULAS .

ANOTHER REASON FOR THE POPULARITY OF THE INTERPOLATION IS THAT IT PROVIDES THE THEORETICAL FOUNDATION FOR THE DERIVATION OF DIFFERENTIATION AND INTE- -GRATION FORMULAS AND FOR THE SOLUTION OF DIFFERENTIAL EQUATIONS,WHICH IS STILL RELEVANT WHEN COMPUTERS ARE USED.

* LANGRANGIAN INTERPOLATION *

WE CONSIDER A SECOND ORDER POLYNOMIAL OF TYPE

y(x) = a1*(x-x2)*(x-x3) + a2*(x-x1)*(x-x3) + a3*(x-x1)*(x-x2)

WHICH PASSES THROUGH THE POINTS $(x_1,y_1),(x_2,y_2)$ AND (x_3,y_3) , WHERE a_1,a_2 AND a_3 ARE UNKNOWN CONSTANTS WHOSE VALUES ARE DETERMINED AS SHOWN BELOW :

AT $x=x_1$, $y(x_1) = a_1*(x_1-x_2)*(x_2-x_3)$ or $a_1 = y_1/((x_1-x_2)*(x_1-x_3))$

AT $x=x_2$, $y(x_2) = a_2*(x_2-x_1)*(x_2-x_3)$ or $a_2 = y_2/((x_2-x_1)*(x_2-x_3))$

AT $x=x_3$, $y(x_3) = a_3*(x_3-x_1)*(x_3-x_2)$ or $a_3 = y_3/((x_3-x_1)*(x_3-x_2))$

SUBSTITUTING THE VALUES OF a_1,a_2 AND a_3 IN THE ABOVE EQUATION, WE GET

$$y(x) = y_1*(((x-x_2)*(x-x_3))/((x_1-x_2)*(x_1-x_3))) + y_2*(((x-x_1)*(x-x_3))/((x_2-x_1)*(x_2-x_3))) + y_3*(((x-x_1)*(x-x_2))/((x_3-x_1)*(x_3-x_2)))$$

USING THE PRODUCT AND SUMMATION NOTATIONS,THE ABOVE EXPRESSION CAN BE

EXPRESSED AS ,

$$y(x) = \sum_{i=1}^{3} y_i \prod_{\substack{j=1 \\ j\neq i}}^{3} (x-x_j)/(x_i-x_j)$$

WHICH IS A SECOND ORDER POLYNOMIAL PASSING THROUGH THREE POINTS.IN GENERAL

FOR "n" POINTS , WE HAVE A "(n-1)th" DEGREE POLYNOMIAL AS FOLLOWS :

$$y(x) = \sum_{i=1}^{n} y_i \prod_{\substack{j=1 \\ j\neq i}}^{n} (x-x_j)/(x_i-x_j)$$

THIS POLYNOMIAL IS KNOWN AS THE "LAGRANGIAN INTERPOLATION FORMULA" AND IS

VERY SIMPLE TO IMPLEMENT ON COMPUTER .

* NOTE THAT LANGRANGIAN FORMULA IS MERELY RELATION BETWEEN TWO VARIABLES,SO BY TAKING "y" AS THE INDEPENDENT VARIABLE WE CAN WRITE A FORMULA GIVING "x" AS A FUNCTION OF "y" AS FOLLOWS :

$$x(y) = \sum_{i=1}^{n} x_i \prod_{\substack{j=1 \\ j \neq i}}^{n} (y-y_j)/(y_i-y_j)$$

THIS POLYNOMIAL IS KNOWN AS THE "LAGRANGIAL INVERSE INTERPOLATION FORMULA" .

* THE MAIN USE OF "LAGRANGIAN INTERPOLATION FORMULA" ARE :

<I>TO FIND ANY VALUE OF A FUNCTION WHEN THE GIVEN VALUE OF INDEPENDENT VAR- -IABLE ARE NOT EQUIDISTANT , AND

<II> TO FIND THE VALUE OF THE INDEPENDENT VARIABLE CORRESPONDING TO A GIVEN VALUE OF A FUNCTION.

* SUPPOSE FOR A GIVEN SET OF POINTS,WE USE "LAGRANGIAL POLYNOMIAL" OF HIGHEST ORDER. USING THE HIGHEST ORDER POLYNOMIAL DOES NOT GUARANTEE THAT THE INTER- -POLATED VALUE WILL BE ACCURATE. THE INTERPOLATED VALUE WILL BE BEST,IF THE DEGREE OF POLYNOMIAL USED INTERPOLATION HAPPENS TO BE OF THE SAME DEGREE AS THE INHERENT DEGREE RELATIONSHIP AMONG THE INDEPENDENT VARIABLE & THE FUNCTION VALUES.

* THE MAIN DISADVANTAGES IN CASE OF "LAGRANGIAL INTERPOLATION",IT IS DIFFICULT

TO FIND THE ORDER OF THE POLYNOMIAL TO BE FITTED. THIS PROBLEM IS WELL TAKEN CARE OF IN "NEWTON'S METHODS OF INTERPOLATION",WHICH MAKES USE OF DIFFERENCES.

* ALGORITHM FOR LAGRANGIAN INTERPOLATION FORMULA *

TO IMPLEMENT THE "LAGRANGIAN INTERPOLATION FORMULA" , LET "xi's" AND "yi's" REPRESENT THE TABLE POINTS AND THE CORRESPONDING VALUES OF THE FUNCTION FOR A SET OF "n" POINTS .

ALGORITHM : LAGRANGIAN

---------   ----------

* INPUT :

<I>   "n" -------------> IT INDICATES THE NUMBER OF POINTS

<II>   x1,x2,...,xn ---> SET OF "n" NO OF VALUES OF x's

<III> y1,y2,...,yn ---> SET OF "n" NO OF VALUES OF y's

<IV>   x ---------------> THE VALUE OF "x" AT WHICH THE INTERPOLATED VALUE IS TO BE COMPUTED.

* OUTPUT :

<I>   Sum ---------------> IT INDICATES THE INTERPOLATED VALUE .

* THE STEPS OF THE ABOVE ALGORITHM IS DESCRIBED BELOW :

----

1)Input n , where "n" is the no of points

2)If (n<1) then goto step 1 else goto step 3

3)Input x

4)Set i=1

5) Input xi,yi

6)Set i=i+1

7)If (i<=n) then goto step 5 else goto step 8

8)Set sum=0.0

9)Set i=1

10) Set prod=1.0

11) Set j=1

12)   If (j#i) then

13)   Set prod = prod*((x-xj)/(xi-xj))

14)   Endif

15) Set j=j+1

16) If (j<=n) then goto step 12 else goto step 17

17)Set sum = sum + (yi*prod)

18)Set i=i+1

19)If (i<=n) then goto step 10 else goto step 20

20)Write sum,"as the interpolated value"

21)End

* ALGORITHM FOR LAGRANGIAN INVERSE INTERPOLATION FORMULA *

TO IMPLEMENT THE "LAGRANGIAN INVERSE INTERPOLATION FORMULA",LET "xi's" AND "yi's" REPRESENT THE TABLE POINTS AND THE CORRESPONDING VALUES OF THE FUNCTION FOR A SET OF "n" POINTS .

ALGORITHM : LAGRANGIAN_INVERSE

---------   ------------------

* INPUT :

<I>   "n" -------------> IT INDICATES THE NUMBER OF POINTS

<II>   x1,x2,...,xn ---> SET OF "n" NO OF VALUES OF x's

<III> y1,y2,...,yn ---> SET OF "n" NO OF VALUES OF y's

<IV>   y ---------------> THE VALUE OF "y" AT WHICH THE INTERPOLATED VALUE IS
TO BE COMPUTED.

* OUTPUT :

<I>   Sum ---------------> IT INDICATES THE INTERPOLATED VALUE .

* THE STEPS OF THE ABOVE ALGORITHM IS DESCRIBED BELOW :

----

1)Input n , where "n" is the no of points

2)If (n<1) then goto step 1 else goto step 3

3)Input y

4)Set i=1

5) Input xi,yi

6)Set i=i+1

7)If (i<=n) then goto step 5 else goto step 8

8)Set sum=0.0

9)Set i=1

10) Set prod=1.0

11) Set j=1

12) If (j#i) then

13) Set prod = prod*((y-yj)/(yi-yj))

14) Endif

15) Set j=j+1

16) If (j<=n) then goto step 12 else goto step 17

17)Set sum = sum + (xi*prod)

18)Set i=i+1

19)If (i<=n) then goto step 10 else goto step 20

20)Write sum,"as the interpolated value"

21)End

EX1: GIVEN THE FOLLOWING TABLE OF VALUES,FIND y(2.5) USING THE LAGRANGIAN INTERPOLATION OF ORDER 3 .

x	0	1	2	3
y(x)	0	2	8	27

ANS:

HERE THE VALUE OF n = 4. USING "LAGRANGIAN INTERPOLATION FORMULA",

WE GET,

y(x) = y1*(((x-x2)*(x-x3)*(x-x4))/((x1-x2)*(x1-x3)*(x1-x4))) +
y2*(((x-x1)*(x-x3)*(x-x4))/((x2-x1)*(x2-x3)*(x2-x4))) +
y3*(((x-x1)*(x-x2)*(x-x4))/((x3-x1)*(x3-x2)*(x3-x4))) +
y4*(((x-x1)*(x-x2)*(x-x3))/((x4-x1)*(x4-x2)*(x4-x3)))

SUBSTITUTING THE VALUES FROM THE GIVEN TABLE OF DATA AND X=2.5,

WE GET,

y(2.5) = 0*(((2.5-1)*(2.5-2)*(2.5-3))/((0-1)*(0-2)*(0-3))) +
1*(((2.5-0)*(2.5-2)*(2.5-3))/((1-0)*(1-2)*(1-3))) +
8*(((2.5-0)*(2.5-1)*(2.5-3))/((2-0)*(2-1)*(2-3))) +
27*(((2.5-0)*(2.5-1)*(2.5-2))/((3-0)*(3-1)*(3-2)))

= 15.625

THUS, y(2.5) = 15.625

EX2: GIVEN THE FOLLOWING TABLE OF VALUES,FIND THE VALUES OF "x" when y=0.390

x	20	25	30	35
y(x)	0.342	0.423	0.500	0.650

|_____|_____|_____|_____|_____|

ANS:

THIS IS A PROBLEM OF "INVERSE INTERPOLATION" . USING "LAGRANGIAN INTER-

-POLATION FORMULA",WE GET,

x(y) = x1*(((y-y2)*(y-y3)*(y-y4))/((y1-y2)*(y1-y3)*(y1-y4))) +

x2*(((y-y1)*(y-y3)*(y-y4))/((y2-y1)*(y2-y3)*(y2-y4))) +

x3*(((y-y1)*(y-y2)*(y-y4))/((y3-y1)*(y3-y2)*(y3-y4))) +

x4*(((y-y1)*(y-y2)*(y-y3))/((y4-y1)*(y4-y2)*(y4-y3)))

SUBSTITUTING THE VALUES FROM THE GIVEN TABLE OF DATA AND X=2.5,WE GET,

x(0.390) = 20*(((0.390-0.423)*(0.390-0.500)*(0.390-0.650))/((0.342-0.423)*(0.342-0.500)*(0.342-0.650))) +

25*(((0.390-0.342)*(0.390-0.500)*(0.390-0.650))/((0.423-0.342)*(0.423-0.500)*(0.423-0.650))) +

30*(((0.390-0.342)*(0.390-0.423)*(0.390-0.650))/((0.500-0.342)*(0.500-0.423)*(0.500-0.650))) +

35*(((0.390-0.342)*(0.390-0.423)*(0.390-0.500))/((0.650-0.342)*(0.650-0.423)*(0.650-0.500)))

= 22.75

HENCE, x(0.390) = 22.75