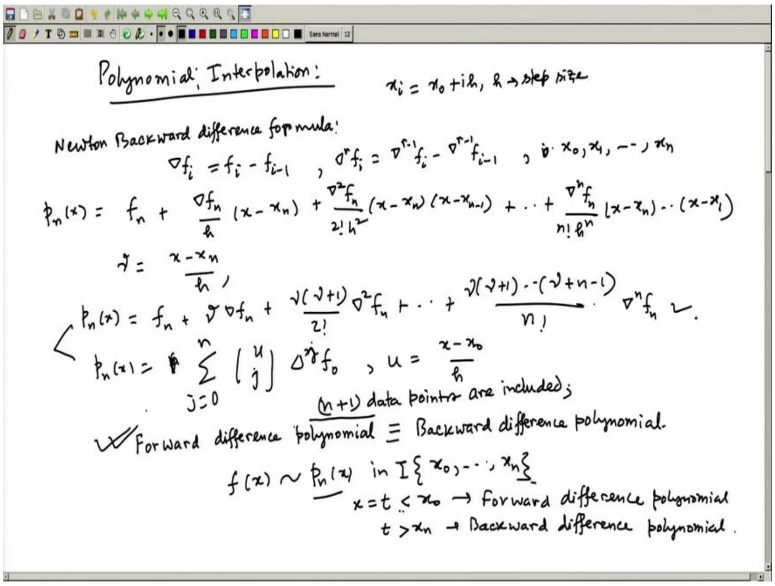
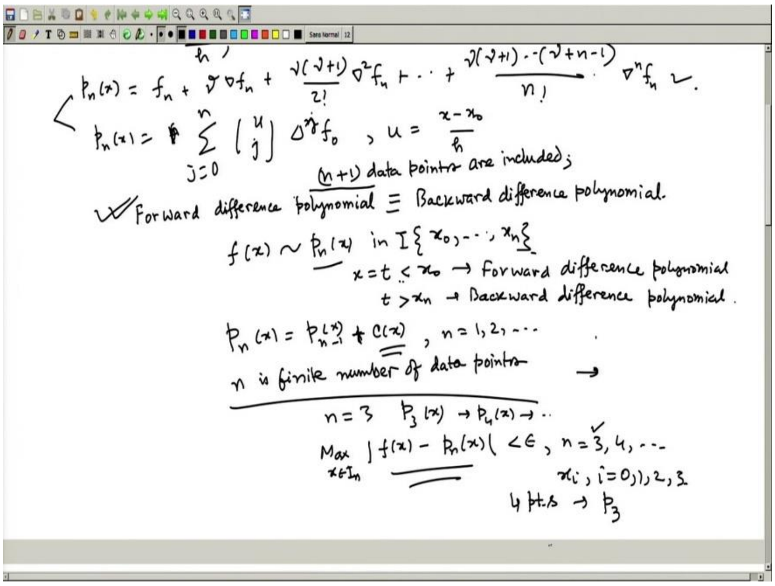
Advanced Computational Techniques
Professor Somnath Bhattacharyya
Department of Mathematics
Indian Institute of Technology, Kharagpur
Lecture 03
Polynomial Interpolation (Contd.)

(Refer Slide Time: 0:30)





So, we will continue with this polynomial interpolation. So, last class we have derived the interpolation polynomial for Lagrange that is the general situation for any choice of arbitrary set of points data points where the points may not be equispaced or arbitrarily set a discrete set of points are considered. So now, we will continue with the interpolation polynomial. So, now we are specific that we have equispaced points. So, that means the node points as we said is

$x_i = x_0 + ih$, h is referred as the step size. So, for this kind of things, we have already derived Newton's forward difference formula and the backward difference formula whichever we talk about.

We can introduce these backward difference formula

$\nabla f_i = f_i - f_{i-1}$, $\nabla^r f_i = \nabla^{r-1} f_i - \nabla^{r-1} f_{i-1}$

So, this polynomial formula for using all n+1 data points, considering $x_0, x_1, \dots, x_n$ remember that. Now, we can write this as

$$p_n(x) = f_n + \frac{\nabla f_n}{h}(x - x_n) + \frac{\nabla^2 f_n}{2!\,h^2}(x - x_n)(x - x_{n-1}) + \cdots + \frac{\nabla^n f_n}{n!\,h^n}(x - x_n)\dots(x - x_1)$$

Now, so that means here come to the previous one, the thing is that we are now going from the bottom of the table so, that means, nth point to 1 point and for the forward difference we are going from 0 to n. So, if we introduce a parameter say

$\vartheta = \frac{x - x_n}{h}$

In that case we can write in the form of a binomial expansion

$$p_n(x) = f_n + \vartheta \nabla f_n + \frac{\nabla^2 f_n}{2!}\vartheta(\vartheta + 1) + \cdots + \frac{\nabla^n f_n}{n!}\vartheta(\vartheta + 1)\dots(\vartheta + n - 1)$$

So, this can be introduced as a binomial form and that is the same type of things as we have derived for the forward difference formula also.

So, forward difference also the same way we have written in this manner,

$p_n(x) = \sum_{j=0}^{n} \binom{u}{j} \Delta^j f_0$, $u = \frac{x - x_0}{h}$

and so, these two when we talk about all the points that means, when n+1 data points are introduced or included. So, whether it is a forward difference formula or backward difference formula, all are identical. Forward difference formula polynomial is equivalent to the backward difference formula. Then why we need to learn is differently because, if we need to truncate if we stop somewhere instead of taking all the n+1 data point altogether instead of that, if we truncate our setup data points to a limited number of points then the advantage, whether I will go for the forward difference formula or backward difference formula.

Now, if I use a forward difference formula, when I have to approximate f(x),

$f(x) \sim p_n(x)\ in\ I\{x_0, x_1, \dots, x_n\}$

So, when a point we choose where we need to find say x=t $< x_0$ so, in that case, or around x_0 less than or equal to and equal to of course, it is identically it is overlapping the node point. So, equal to means this is the polynomial itself. So, when it is near the first point of the table then it is advisable that you use a forward difference formula. And when we choose t say around x_n close to x_n the bottom of the table so, then we choose the backward difference formula backward difference polynomial.

So, these are the situations where we have to decide whether we go for a forward difference polynomial or backward difference polynomial. Now, this will be the same when we choose all the n points. When all n points are chosen, then this forward difference and backward difference polynomial are identical. So, these kind of things are all taught in the elementary numerical analysis so, I am not going into details on that, just an overview on how the Lagrange interpolation polynomial is derived and then from there, how we get this Newton's polynomial.

So, as I said before, the advantage in using the Newton's polynomial is that if I know $p_n(x)$

$p_n(x) = p_{n-1}(x) + c(x)$

$c(x)$,that extra term so which can be obtained from the divided difference polynomial or the difference expansion and add to this lower degree polynomial to get the higher one. So, that is the way we go for n=1, 2, … and in that way we can step by step increase the degree of the polynomial and consequently we are using more number of points from the given data set.

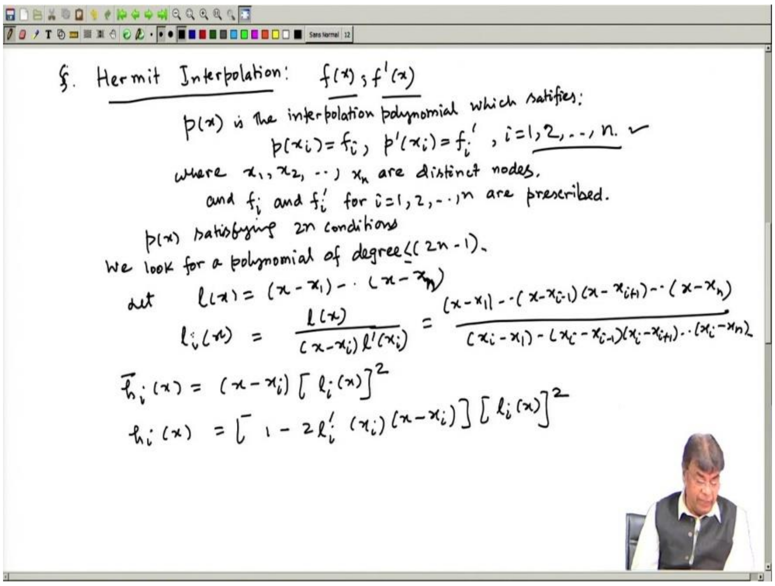
Now, when we choose again when we choose all the points if we construct a polynomial by choosing all the data points then whether it is by Lagrange polynomial or Newton's forward or Newton's backward we get the identical result, identical polynomial. So, this is about the elementary polynomial interpolation.

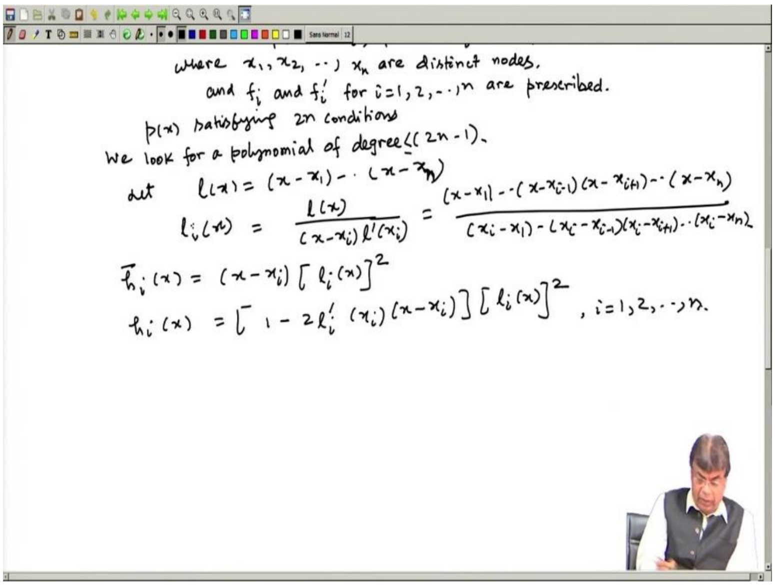
Now, there are several other situation where when instead of going from the bottom or forward or backward we can have a central position. Now, I am not much interested to talk about the central difference and all these things because there normally n is taken to be finite, finite number of data points. So, not many data points are chosen. So, that is why it does not make much difference whether you go for a central difference or that means this x=t is in between the table. So, n is restricted to say 2, 3, 4, 5 like this way.

So, normal procedure is that you choose n=3 then you find out the $p_3(x)$ so, once $p_3(x)$ is done then from there you go to $p_4(x)$ and so on, like that, and you see this degree of precision is achieved that means, whether the difference is the absolute difference between these f(x)-p(x), how it differs. So, as n is progressing 3, 4 like this way as n is progressing, how these differences is occurring some x belongs to I_n. So, that is how the interpolation polynomial goes.

So, instead of choosing all the data points provided in the table at a time so, we go step by step that is first we choose n=3, find out a polynomial of degree n =3 if we choose so that means, we are choosing four points basically, because we are considering x_i for i =0, 1, 2, 3. So, if you have four points then it will be a polynomial of degree 3, one less. So, if n+1 points are there then degree is n.

(Refer Slide Time: 12:59)





Now, we talk about another very important polynomial formula that is Hermit interpolation. So, here what we want to do is that not only the function f(x), we also interpolate the function derivative $f'(x)$ and in that process we raise the degree of the polynomial. So, the unknown is not only f(x), but $f'(x)$ also is forming an unknown, not unknown I mean the data points. So, that means, we are constructing a polynomial p(x), the interpolation polynomial which satisfy these conditions

$p(x_i) = f_i, \quad p'(x_i) = f'_i,$, i=1,2,…,n

So, that means, here exactly we are taking n data points

So, we would like to construct a polynomial p(x) such that this happening so, should not only the function is evaluated but its derivative also becoming the same, where $x_1, x_2, \dots, x_n$ are distinct node points. So, it can be real number whatever the data is given real or complex and f_i and f'_i for i=1,2,…,n are prescribed. In some cases, what we need is not only the function to be evaluated its derivative also has to be evaluated or should satisfy this condition. So, that in some case is important.

In addition, we will derive a numerical integration formula where we need a polynomial which is using n node points, but the degree of the polynomial now will become 2n-1 because we have now p(x) satisfying 2n conditions. So, we look for a polynomial of degree 2n-1 because 1 constant so, degree at most 2n-1.

Now let us define

$l(x) =(x - x_1) \ldots (x - x_n)$

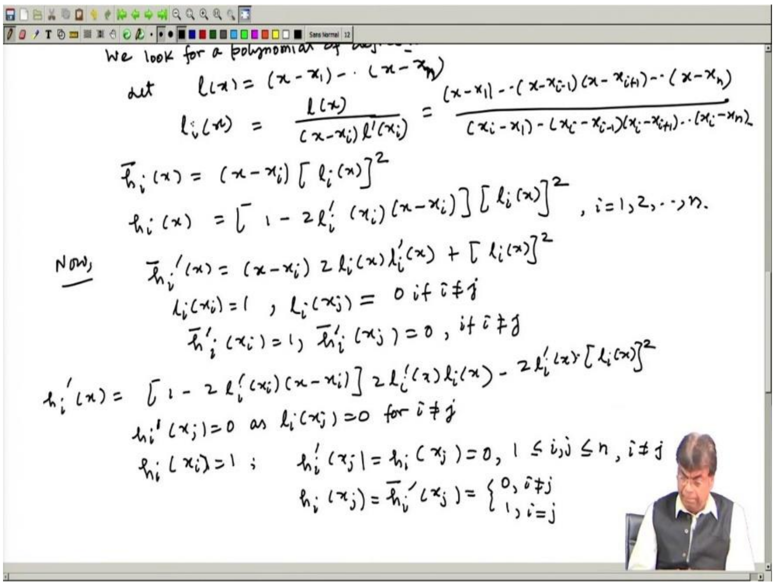
$l_i(x)=\frac{l(x)}{(x-x_i)\, l'(x_i)} = \frac{(x-x_0)\ldots(x-x_{i-1})(x-x_{i+1})\ldots(x-x_n)}{(x_i-x_0)\ldots(x_i-x_{i-1})\ldots(x_i-x_n)}$

$$\overline{h_\iota}(x) = (x - x_i)[l_i(x)]^2$$

$$h_i(x) = [1 - 2l_i'(x_i)(x - x_i)][l_i(x)]^2, i = 1,2, \ldots , n$$

Now, we construct two polynomials. So, $l(x)$ is a polynomial of degree n. So, $\overline{h_\iota}(x)$ is a polynomial of degree 2n-1 and $h_i(x)$, both $h_i(x)$ and $\overline{h_\iota}(x)$ are polynomial of degree 2n-1.

(Refer Slide Time: 19:17)



So, now observe this

$$\overline{h_i'}(\mathrm{x}) = (\mathrm{x} - x_i)^2 \; l_i(\mathrm{x}) l_i'(\mathrm{x}) + [l_i(\mathrm{x})]^2$$

Now, by definition

$l_i(x_i) = 1, \quad l_i(x_j) = 0$ if i$\neq j$

So,

$$\overline{h_i'}(x_i) = 1 \,, \overline{h_i'}(x_j) = 0 \text{ if } \mathrm{i} \neq \mathrm{j}$$

$$h_i'(\mathrm{x}) = [1 - 2l_i'(x_i)(\mathrm{x} - x_i)]2l_i'(\mathrm{x})l_i(\mathrm{x}) - 2l_i'(\mathrm{x})(l_i(\mathrm{x}))^2)$$

Now, with that also we have

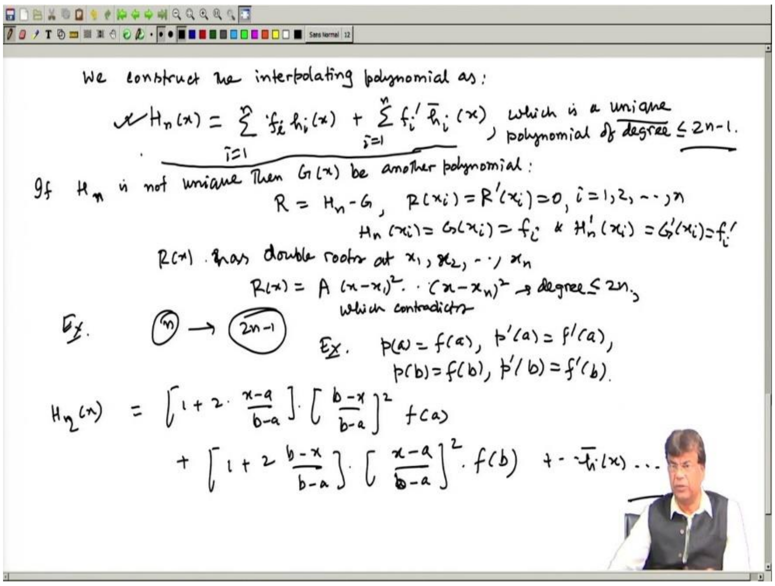
$h_i'(x_j) = 0$ as $l_i(x_j) = 0$ for i$\neq j$

$h_i(x_i)$=1, $\quad h_i'(x_j) = h_i(x_j)$=0, 1$\leq i, j \leq n, \;\; i \neq j$

$$h_i(x_j) = \overline{h_i'}(x_j) \begin{cases} 0, i \neq j \\ 1, i = j \end{cases}$$

So, this setup relation we can see from here from the construction.

(Refer Slide Time: 23:30)



Now, we construct the interpolation polynomial as

$$H_n(x) = \sum_{i=1}^{n} f_i h_i(x) + \sum_{i=1}^{n} f_i{}'\overline{h_\imath}\,(x)$$

which is a unique polynomial of degree≤ 2n -1. How it is unique? It is so, obviously what we find that under this condition this $h_n(x)$ is becoming the function value at all the node points and also the function derivative values at those node points for i =1 to n.

To check that uniqueness so, let G(x) be another polynomial if $H_n(x)$ is not unique. So, in that case

$$R = H_n - G$$

So, what do we find that

$$R(x_i) = R'(x_i) = 0, i = 1,2, \dots, n$$

Because

$H_n(x_i) = G(x_i) = f_i$

And, $H_n{}'(x_i) = G'(x_i) = f_i{}'$

So, this implies R is vanished at, so that means, there R(x) has double root at $x_1, x_2, \dots, x_n$

So, in that case

$R(x) = A(x - x_1)^2 \ldots (x - x_n)^2$

so degree≤2n, which contradicts.

So, that means we cannot have a polynomial which satisfying these two conditions, a different polynomial other than this. So, a polynomial which is becoming function value at these designated points, node points and also its derivative is becoming the function derivative values at those node points will have degree 2n-1 and it is unique.

So, for example, we can have using n number of node points we are constructing a degree of the polynomial as 2n-1, so if we increase the degree of the polynomial, we get the precision is much higher. So, an example

$p(a) = f(a), p'(a) = f'(a)$

$p(b) = f(b), p'(b) = f'(b)$

So, one can find out the polynomial as

$$H_n(x) = \left[1 + 2\frac{x-a}{b-a}\right]\left[\frac{b-x}{b-a}\right]^2 f(a) + \left[1 + 2\frac{b-x}{b-a}\right]\left[\frac{x-a}{b-a}\right]^2 f(b) + \cdots \bar{h}(x) \ldots$$

So, this is one part.

Another part, one can construct it the same way. So, only thing is that we have to construct this first part that is the function part and the other part is $\bar{h}(x)$ that can be constructed in the same manner. So, we stop it here and proceed to the next one.