Chebyshev Nodes

Definitions and Basics

In this notebook I will do an example where the nodes of an interpolating polynomial are determined by using the zeros of a Chebyshev Polynomial.

The Chebyshev Polynomials are defined for x in the interval [-1, 1] and are defined as:

$$T_n(x) = \cos\left(n \cos^{-1}(x)\right)$$

Using the angle sum angle difference formula cos(a±b)=cos(a)cos(b)∓sin(a)sin(b) one can write

$$T_{n+1}(x) = \cos\left((n+1)\cos^{-1}(x)\right) = \cos\left(n\cos^{-1}(x)\right)\cos\left(\cos^{-1}(x)\right) - \sin\left(n\cos^{-1}(x)\right)\sin\left(\cos^{-1}(x)\right)$$
$$T_{n-1}(x) = \cos\left((n-1)\cos^{-1}(x)\right) = \cos\left(n\cos^{-1}(x)\right)\cos\left(\cos^{-1}(x)\right) + \sin\left(n\cos^{-1}(x)\right)\sin\left(\cos^{-1}(x)\right)$$

If we add these together one obtains

$$T_{n+1}(x) + T_{n-1}(x) = 2\cos\left(n\cos^{-1}(x)\right)\cos\left(\cos^{-1}(x)\right)$$

or

$$T_{n+1}(x) + T_{n-1}(x) = 2\,T_n(x)\,x$$

Solving for $T_{n+1}(x)$ gives

$$T_{n+1}(x) = 2\,x\,T_n(x) - T_{n-1}(x)$$

This is the **recurrence relationship** for the Chebyshev polynomials. It allows you to find any of the new polynomials from the two previous ones. However, in order to do this you need to determine the first 2 (n=0 and n=1). These two are rather easy to compute though as

$$T_0(x) = \cos\left(0\cos^{-1}(x)\right) = 1$$
$$T_1(x) = \cos\left(1\cos^{-1}(x)\right) = x$$

Starting with these you can build up the Chebyshev Polynomials for any n. For instance using the recurrence relationship with n=1 we have that,

$$T_2(x) \;=\; 2\,x\,T_1(x) - T_0(x) = 2\,x(x) - 1 = 2\,x^2 - 1$$

Mathematica has built in formulas for the Chebyshev polynomials. ChebyshevT[n, x] will create $T_n(x)$.

```
ChebyshevT[0, x]
ChebyshevT[1, x]
ChebyshevT[2, x]
ChebyshevT[3, x]
ChebyshevT[4, x]
ChebyshevT[5, x]
```

```
1
```

```
x
```

$-1+2x^2$

$-3x+4x^3$

$1-8x^2+8x^4$

$5x-20x^3+16x^5$

One interesting observation is that the polynomials are all either even or odd and that the power of $x^n$is 2^{n-1}(except for T_0). One of your homework problems is to verify that these polynomials are also an *orthogonal set* of polynomials.

Zeros and Extrema

Using the definition of the Chebyshev Polynomials, we can use the properties of cos(x) to determine the zeros of the Chebyshev Polynomials. Recall that cos(θ) is equal to zero at odd multiples of $\pi/2$. Thus we need to have $n\cos^{-1}(x) = (2j-1)\pi/2$ where j is any integer. If we solve this for x we obtain

$$x = \cos\left(\frac{(2j+1)\pi}{2n}\right)$$

These will of course repeat their values (since cos(x) is periodic) so we take j=0,1,2, 3,...n-1 to get exactly one complete set of the roots. The extreme values occur when cos(θ)=±1, which occur at multiples of π. If we then set $n\cos^{-1}(x) = j\pi$.

$$x = \cos\left(\frac{j\pi}{n}\right)$$

Again if we take j=0,1, 2, ... n-1 we will have a complete set of the extrema occurring in the interval of definition.

Monic Polynomials

A monic polynomial is a polynomial where the highest power of x has the coefficient equal to 1. Earlier we made the observation that the coefficient of $T_n(x)$ was 2^{n-1} except for $T_0(x) = 1$. If we wanted the *monic* Chebyshev polynomials then we just need to divide by this value. We define these polynomials by using a ~ to denote that they are monic.

$$\tilde{T}_n(x) = \begin{cases} T_0(x) = 1 & n = 0 \\ \frac{1}{2^{n-1}} T_n(x) & n \neq 0 \end{cases}$$

The roots of the Monic Polynomials are the same (if one is zero, the other must be also). The extrema occur at the same *locations* but the instead of having maximum and minimum values of 1. The maximum and minimum values are now $\pm 1 / (2^{n-1})$, which gets smaller and smaller as n increases.

Best Nodes

It turns out that the Monic Chebyshev Polynomials satisfy a nice Theorem.

Theorem: For any monic n^{th} degree polynomial p(x)

$$\frac{1}{2^{n-1}} = \max_{x\in[-1,1]} \left| \tilde{T}_n(x) \right| \leq \max_{x\in[-1,1]} \left| p(x) \right|$$

What this theorem says is that if you want to have a monic polynomial on [-1, 1] such that the maximum of its absolute value is the smallest, then you should choose a Chebyshev polynomial of the appropriate degree. In particular if we are interpolating for x∈[-1,1] and we want to find the nodes that make the maximum of $|(x - x_0)(x - x_1) \ldots (x - x_n)|$ the smallest then we should choose the $x_i$to be the zeros of the Chebyshev polynomial of degree n+1. So we should choose:

$$x_j = \cos\left(\frac{(2j+1)\pi}{2(n+1)}\right) \text{ for j=0,1, 2,... n}$$

If we do this and we are working on the interval [-1, 1] then we know that the maximum value of $|(x - x_0)(x - x_1) \ldots (x - x_n)|$ for x in [-1,1] will be $1/2^n$ (note that this is the degree n+1 Chebyshev monic polynomial).

If we want to do this on intervals other then [-1,1] then we just have to shift and stretch the nodes. In particular if interval is [a,b] then the nodes should be chosen to be:

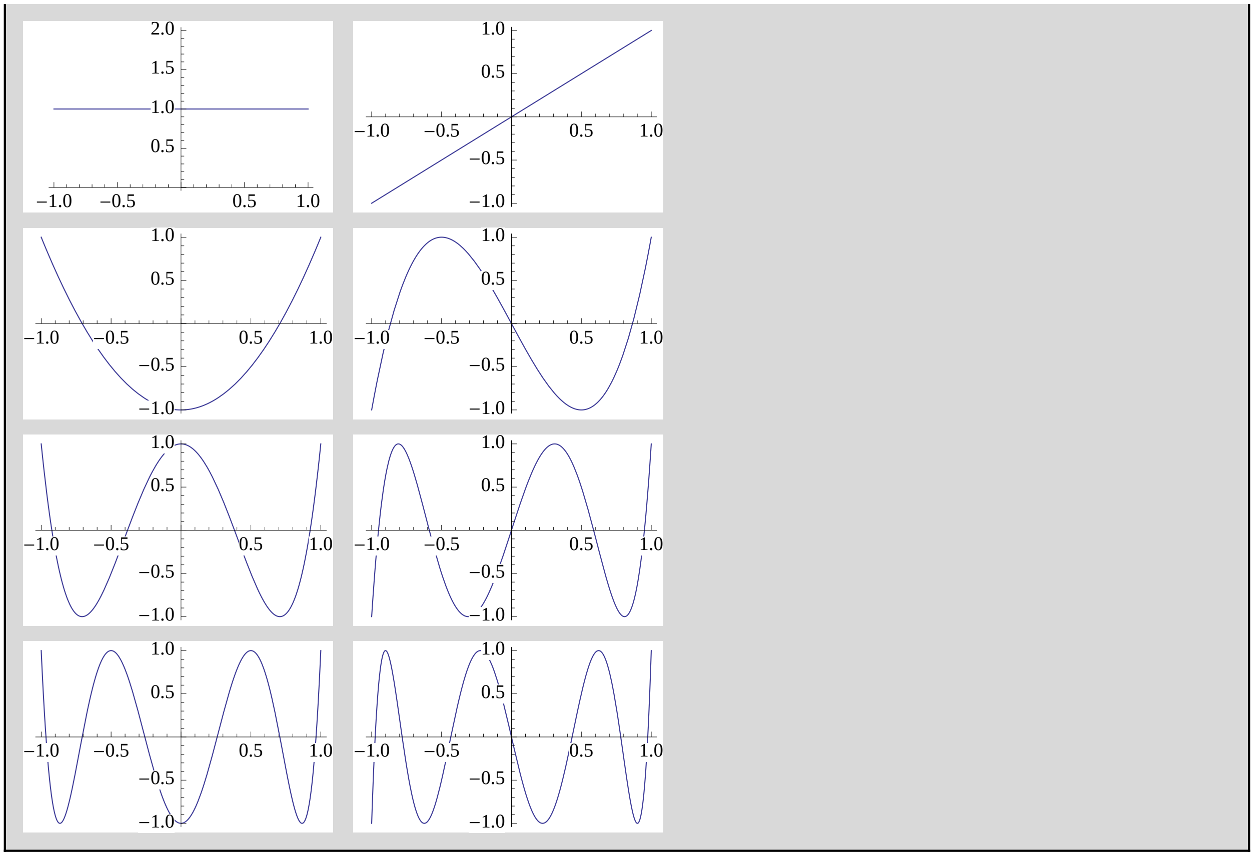
$$x_j = \left(\frac{a+b}{2}\right) + \left(\frac{b-a}{2}\right)\cos\left(\frac{(2j+1)\pi}{2(n+1)}\right) \text{ for j=0,1, 2,... n}$$

In this case the maximum also changes somewhat too. We then have that the maximum value of $|(x - x_0)(x - x_1) \ldots (x - x_n)|$ for x in [a,b] will be $(b - a)^{n+1} / 2^{2n+1}$

Example

First lets just take a look at some of the Chebyshev Polynomials and their plots

```
GraphicsGrid[{{Plot[ChebyshevT[0, x], {x, -1, 1}], Plot[ChebyshevT[1, x], {x, -1, 1}]},
  {Plot[ChebyshevT[2, x], {x, -1, 1}], Plot[ChebyshevT[3, x], {x, -1, 1}]},
  {Plot[ChebyshevT[4, x], {x, -1, 1}], Plot[ChebyshevT[5, x], {x, -1, 1}]},
  {Plot[ChebyshevT[6, x], {x, -1, 1}], Plot[ChebyshevT[7, x], {x, -1, 1}]}}]
```



Now let us do an example. Suppose we want to interpolate $\cos(x)$ on the interval $[2, 5]$ using a 4^{th} degree polynomial and we want to choose the nodes to minimize thee error due to $|(x-x_0)(x-x_1)(x-x_2)(x-x_3)(x-x_4)|$. We then need to choose the nodes to be the shifted zeros of the Chebyshev polynomial of degree 5. So we need

$$x_j = \left(\frac{a+b}{2}\right) + \left(\frac{b-a}{2}\right) \cos\left(\frac{(2j+1)\pi}{2(n+1)}\right) \text{ for j=0,1, 2,... n}$$

Here we have a=2, b=5 and n=4, this gives:

$$x_j = \left(\frac{7}{2}\right) + \left(\frac{4}{2}\right) \cos\left(\frac{(2j+1)\pi}{10}\right) \text{ for j=0,1, 2, 3, 4}$$

or

$$x_0 = 3.5 + (1.5)\cos(\pi/10), \quad x_1 = 3.5 + (1.5)\cos(3\pi/10), \quad x_2 = 3.5 + (1.5)\cos(\pi/2),$$
$$x_0 = 3.5 + (1.5)\cos(7\pi/10), \quad x_0 = 3.5 + (1.5)\cos(9\pi/10),$$

Below I have computed these in *Mathematica* but have called them the variables a instead of x. You could also find the values by doing Solve[ChebyshevT[5, x] == 0, x] and then multiplying and adding the correct constant.

In[9]:=
```
a0 = (7 / 2) + (3 / 2) Cos[π / 10]
a1 = (7 / 2) + (3 / 2) Cos[3 π / 10]
a2 = (7 / 2) + (3 / 2) Cos[5 π / 10]
a3 = (7 / 2) + (3 / 2) Cos[7 π / 10]
a4 = (7 / 2) + (3 / 2) Cos[9 π / 10]
```

Out[9]= $\frac{7}{2} + \frac{3}{2}\sqrt{\frac{5}{8} + \frac{\sqrt{5}}{8}}$

Out[10]= $\frac{7}{2} + \frac{3}{2}\sqrt{\frac{5}{8} - \frac{\sqrt{5}}{8}}$

Out[11]= $\frac{7}{2}$

Out[12]= $\frac{7}{2} - \frac{3}{2}\sqrt{\frac{5}{8} - \frac{\sqrt{5}}{8}}$

Out[13]= $\frac{7}{2} - \frac{3}{2}\sqrt{\frac{5}{8} + \frac{\sqrt{5}}{8}}$

Notice that a0 is not the smallest number. That is OK, the nodes do not need to be ordered from smallest to largest. Next define the function you want to interpolate.

In[1]:=
```
g[x_] = Cos[x]
```

Out[1]=
```
Cos[x]
```

What follows below is some code that computes the interpolating polynomial and then plots the polynomial and its error. The default is set to the Chebyshev nodes, but you can then move the nodes around. You will notice that when you move the nodes you can make the maximum error smaller, but it is difficult. This is because you are only controlling the maximum part of the polynomial piece of the error. The 5th derivative terms would contribute too. The red line indicates the bound on the error when using the Chebyshev nodes calculated by taking the product of the maximum of the fifth derivative of cos(x) and the maximum of the Chebyshev polynomial. It is still larger then the actual error, but if you move the nodes it is much easier to make the error larger then this.

In[6]:=
```
G[x0_, x1_, x2_, x3_, x4_] :=
 Module[{T}, T = Table[{x, N[g[x]]}, {x, {x0, x1, x2, x3, x4}}];
  T]
```

In[7]:=
```
f[x0_, x1_, x2_, x3_, x4_, x_] := InterpolatingPolynomial[G[x0, x1, x2, x3, x4], x]
```

In[16]:=

```
Manipulate[GraphicsGrid[
  {{Show[Plot[{f[x0, x1, x2, x3, x4, x], g[x]}, {x, 2, 5}, PlotRange → {-1, 1}],
      ListPlot[G[x0, x1, x2, x3, x4], PlotMarkers → {Automatic, Medium}]]},
   {Plot[{Abs[f[x0, x1, x2, x3, x4, x] - g[x]], 3^5 / (5! * 2^9)},
      {x, 2, 5}, PlotRange → {0, .005}]}}],
 {{x0, a4}, 2, 5}, {{x1, a3}, 2.1, 5}, {{x2, a2}, 2.2, 5},
 {{x3, a1}, 2.30, 5}, {{x4, a0}, 2.4, 5}]
```

Out[16]=