Chebyshev interpolation

In this section it will be convenient to use $x_1, \ldots, x_n$ rather than $x_0, x_1, \ldots, x_n$.

Review. If $P_{n-1}(x)$ is the interpolating polynomial for $f(x)$ at $x_1, \ldots, x_n$, then

$$f(x) - P_{n-1}(x) = \underbrace{\frac{f^{(n)}(\xi)}{n!}(x - x_1)\cdots(x - x_n)}_{\text{interpolation error}}$$

for some ξ between the x_i and x.

Suppose we wish to minimize the maximal error on some interval $[a, b]$. After shifting and scaling, we can normalize this interval to the interval $[-1, 1]$.

It therefore is natural to choose $x_1, \ldots, x_n$ such that $\max_{x \in [-1,1]} |(x - x_1)\cdots(x - x_n)|$ is minimized.

Amazingly, in Theorem 91, we will be able to say exactly for which choice of x_i this happens!

Example 89. For small n, choose $x_1, x_2, \ldots, x_n$ such that $\max_{x \in [-1,1]} |(x - x_1)\cdots(x - x_n)|$ is minimal.

Solution. In the cases below, we will appeal to symmetry and assume that the optimal nodes must be such that $x_1 = -x_n$, $x_2 = -x_{n-1}$, $\ldots$. As such, the arguments only prove that our choices are optimal if that assumption is correct. In hindsight, from our general proof in Theorem 91, this will prove to be correct.

- $n = 1$: By symmetry, the optimal choice should be $x_1 = 0$.
- $n = 2$: By symmetry, $x_1 = -x_2$. Write $c = x_2$ and let $f(x) = (x + c)(x - c) = x^2 - c^2$.
  Since $f'(x) = 2x = 0$ only if $x = 0$, it follows that $\max_{x \in [-1,1]} |f(x)|$ has to occur at $x = 0$ or at the endpoints $x = \pm 1$. The corresponding values are $|f(0)| = c^2$, $|f(\pm 1)| = 1 - c^2$.
  From a plot of $m(c) = \max(c^2, 1 - c^2)$ it is clear that the minimum of $m(c)$ is achieved when $c^2 = 1 - c^2$. This latter equation has the unique positive solution $c = \frac{\sqrt{2}}{2} = \cos\left(\frac{\pi}{4}\right)$.
  Note that the x_i are $\underbrace{\cos\left(\frac{\pi}{4}\right)}_{\sqrt{2}/2}, \underbrace{\cos\left(\frac{3\pi}{4}\right)}_{-\sqrt{2}/2}$.
- $n = 3$: By symmetry, $x_1 = -x_3$ and $x_2 = 0$. Write $c = x_3$ and let $f(x) = (x + c)x(x - c) = x(x^2 - c^2)$.
  Since $f'(x) = 3x^2 - c^2 = 0$ only if $x = \pm\frac{c}{\sqrt{3}}$, it follows that $\max_{x \in [-1,1]} |f(x)|$ has to occur at $x = \pm\frac{c}{\sqrt{3}}$ or at the endpoints $x = \pm 1$. The corresponding values are $\left|f\left(\frac{c}{\sqrt{3}}\right)\right| = \frac{|c|}{\sqrt{3}}\left(\frac{2c^2}{3}\right) = \frac{2|c|^3}{3\sqrt{3}}$, $|f(\pm 1)| = 1 - c^2$.
  From a plot of $m(c) = \max\left(\frac{2|c|^3}{3\sqrt{3}}, 1 - c^2\right)$ it is clear that the minimum of $m(c)$ is achieved when $\frac{2|c|^3}{3\sqrt{3}} = 1 - c^2$. This latter equation has the unique positive solution $c = \frac{\sqrt{3}}{2} = \cos\left(\frac{\pi}{6}\right)$.
  Note that the x_i are $\underbrace{\cos\left(\frac{\pi}{6}\right)}_{\sqrt{3}/2}, \underbrace{\cos\left(\frac{3\pi}{6}\right)}_{0}, \underbrace{\cos\left(\frac{5\pi}{6}\right)}_{-\sqrt{3}/2}$.
- $n = 4$: The pattern continues and the x_i turn out to be $\cos\left(\frac{\pi}{8}\right), \cos\left(\frac{3\pi}{8}\right), \cos\left(\frac{5\pi}{8}\right), \cos\left(\frac{7\pi}{8}\right)$.

  Comment. We have the less familiar trig values $\cos\left(\frac{\pi}{8}\right) = \frac{1}{2}\sqrt{2 + \sqrt{2}}$ and $\cos\left(\frac{3\pi}{8}\right) = \frac{1}{2}\sqrt{2 - \sqrt{2}}$.

Armin Straub
straub@southalabama.edu

Example 90. (bonus!) Suppose we are doing interpolation on the interval $[-1,1]$ and we want the endpoints to be interpolation nodes; that is, $x_1 = -1$ and $x_n = 1$. Choose the remaining nodes such that $\max_{x\in[-1,1]} |(x-x_1)\cdots(x-x_n)|$ is minimal.

Do this for $n = 1, 2, 3$ to collect a bonus point.

An extra bonus point if you can figure out what happens for any n? (*Hint*: compare with the Chebyshev case.)

Theorem 91. (Chebyshev's theorem) For the **Chebyshev nodes**

$$x_j = \cos\left(\frac{(2j-1)}{2n}\pi\right), \quad j = 1, \ldots, n,$$

we have

$$\max_{x\in[-1,1]} |(x-x_1)\cdots(x-x_n)| = \frac{1}{2^{n-1}}.$$

That value is the minimal value for any choice of roots $x_1, \ldots, x_n$.

The corresponding polynomials $T_n(x) = 2^{n-1}(x-x_1)\cdots(x-x_n)$ are known as the **Chebyshev polynomials** of the first kind.

Note that these are scaled by 2^{n-1} so that the maximum is 1.

Proof. We will show below that $T_n(\cos(\theta)) = \cos(n\theta)$, which implies that $|T_n(x)| \leqslant 1$ for all $x \in [-1,1]$. Moreover, at $x = \cos\left(k\frac{\pi}{n}\right)$ for $k = 0, 1, \ldots, n$ the values of $T_n(x)$ alternate between 1 and -1.

Write $P_n(x) = (x-x_1)\cdots(x-x_n)$. It follows that $\max_{x\in[-1,1]} |P_n(x)| = \frac{1}{2^{n-1}}$ as claimed.

Suppose that there is a polynomial $Q_n(x) = (x-r_1)\cdots(x-r_n)$ for which $\max_{x\in[-1,1]} |Q_n(x)| < \frac{1}{2^{n-1}}$.

Note that $d(x) := P_n(x) - Q_n(x)$ has the following properties:

- $d(x)$ is of degree at most $n-1$ (because the x^n terms cancel).
- At $x = \cos\left(k\frac{\pi}{n}\right)$ for $k = 0, 1, \ldots, n$ the values of $d(x)$ alternate between $+$ and $-$.
  (Because $P_n(x) = \pm\frac{1}{2^{n-1}}$ while $|Q_n(x)| < \frac{1}{2^{n-1}}$.)
- Hence, between these $n+1$ values, there must be n zeros. That is impossible because $d(x)$ has degree less than n.

This contradiction shows that no such polynomial $Q_n(x)$ can exist. □

The following is Theorem 83 combined with Chebyshev's Theorem 91.

Theorem 92. (interpolation error using Chebyshev nodes) If P_{n-1} is the interpolating polynomial for f at n Chebyshev nodes, then the interpolation error can be bounded as

$$\max_{x\in[-,1,1]} |f(x) - P_{n-1}(x)| \leqslant \frac{1}{2^{n-1}n!} \max_{\xi\in[-,1,1]} |f^{(n)}(\xi)|.$$

Fine print. As in Theorem 83, we need that f is n times continuously differentiable.

Comment. Theorem 92 guarantees convergence, as $n \to \infty$, of the interpolating polynomials P_n to f provided that the derivatives of f don't grow too fast. On the other hand, one can show that, for certain functions f, no sequence of interpolating polynomials will converge to f.

Advanced comment. Theorem 92 can be interpreted as showing that, for a given function f, the Chebyshev interpolant P_n is a good approximation of f on the interval $[-1,1]$. However, that does not mean that it is the best polynomial approximation of degree n (in the sense of minimizing the maximal error). One can show that there exists a unique such best polynomial B_n. However, B_n is difficult to compute. On the other hand, the Chebyshev interpolant Q_n is close to best in the sense that

$$\max_{x\in[-,1,1]} |f(x)-Q_n(x)| \leqslant \left(4+\frac{4}{\pi^2}\log(n)\right) \max_{x\in[-,1,1]} |f(x)-B_n(x)|.$$

Example 93. Suppose we approximate a function $f(x)$ on the interval $[-1,1]$ by a polynomial interpolation $P(x)$. Suppose that we know that $|f^{(n)}(x)| \leqslant n$ for all $x\in[-1,1]$.

(a) Give an upper bound for the maximal error if we use the interpolation nodes $-1, -\frac{1}{3}, \frac{1}{3}, 1$.

(b) Give an upper bound for the maximal error if we use 4 Chebyshev nodes instead.

(c) How many Chebyshev nodes do we need to use in order to guarantee that the maximal error is at most 10^{-3}?

Solution.

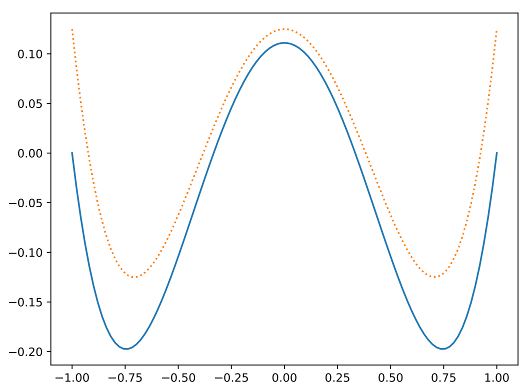
(a) This is the same problem as in the last part of Example 85.

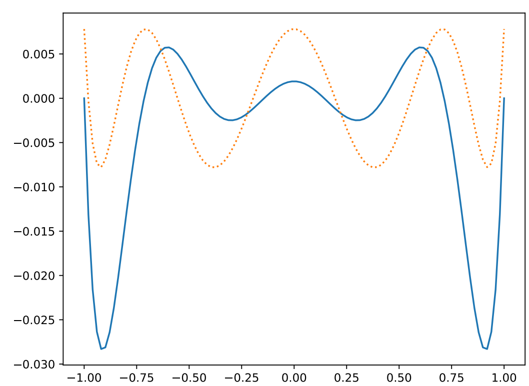
Our bound for the error was $|f(x)-P(x)| \leqslant \frac{1}{6} \max_{z\in[-1,1]} \left|(z^2-1)\left(z^2-\frac{1}{9}\right)\right| = \frac{1}{6}\cdot\frac{16}{81} \approx 0.0329$.

(b) By Theorem 91, $\max_{x\in[-1,1]} |(x-x_1)\cdots(x-x_n)| = \frac{1}{2^{n-1}}$ for Chebyshev nodes. In our case, $n=4$.

Therefore, our bound for the error is $|f(x)-P(x)| \leqslant \frac{1}{6}\frac{1}{2^{4-1}} = \frac{1}{48} \approx 0.0208$.

Comment. This bound is better than the one for the same number of equally spaced points. Indeed, for the Chebyshev nodes, this error estimate is best possible. In the plots below, we can see the difference between $(x-x_1)\cdots(x-x_n)$ in the case of equally spaced x_i and Chebyshev nodes x_i (in dotted). The first plot shows the case $n=4$ and the difference is moderate. The difference becomes very visible in the second plot which shows the case $n=8$. We can see how, for the equally spaced nodes, we get large (negative) values towards the endpoints of $[-1,1]$ while, for the Chebyshev nodes, there are no such wild swings.





(c) By Theorem 92, using n Chebyshev nodes, the error is bounded as

$\max_{x\in[-,1,1]} |f(x)-P_{n-1}(x)| \leqslant \frac{1}{2^{n-1}n!} \max_{\xi\in[-,1,1]} |f^{(n)}(\xi)| \leqslant \frac{1}{2^{n-1}(n-1)!}$.

We thus need to choose n so that $2^{n-1}(n-1)! \geqslant 10^3$.

Computing $2^{n-1}(n-1)!$ for $n=1,2,\ldots$, we obtain $1, 2, 8, 48, 384, 3840$. Thus, for $n=6$ Chebyshev nodes the maximal error is guaranteed to be less than 10^{-3}.

Example 94. Suppose we approximate $\sin(x)$ on the interval $[-1,1]$ by a polynomial interpolation $P(x)$. How many Chebyshev nodes do we need to use in order to guarantee that the maximal error is at most 10^{-16}?

Solution. In the case $f(x)=\sin(x)$ we know that $|f^{(n)}(x)|\leqslant 1$ for all $x\in[-1,1]$ and all n.

Therefore, by Theorem 92, using n Chebyshev nodes, the error is bounded as

$$\max_{x\in[-,1,1]}|f(x)-P_{n-1}(x)|\leqslant\frac{1}{2^{n-1}n!}\max_{\xi\in[-,1,1]}|f^{(n)}(\xi)|\leqslant\frac{1}{2^{n-1}n!}.$$

We need to choose n so that $2^{n-1}n!\geqslant 10^{16}$. We compute $2^{n-1}n!$ for $n=1,2,\ldots$ and find that this first happens when $n=15$ (see the next Python example).

Thus, for $n=15$ Chebyshev nodes the maximal error is guaranteed to be less than 10^{-16}.

Comment. Recall that double precision floats have a precision of slightly less than 16 decimal digits. Thus we could, in theory, implement an accurate $\sin(x)$ function by using a degree 14 polynomial.

Advanced comment. Note that some care is required to translate this result into practice. For instance, if we proceed as in Example 87 then the resulting maximal error using 15 Chebyshev nodes is about $4.7\cdot 10^{-11}$, which is considerably larger than 10^{-16} (even if we take into account that we are limited by using double precision floats). The cause for this is that the computed interpolating polynomial is not accurate to full precision. Indeed, in the documentation for the function `interpolate.lagrange`, we find the following statement: *Warning: This implementation is numerically unstable. Do not expect to be able to use more than about 20 points even if they are chosen optimally.*

Example 95. Python A function for computing $n!$ is available in the Python `math` library (as well as in `numpy` and `scipy`). However, here is a possible quick implementation from scratch:

```
>>> def factorial(n):
        f = 1
        for k in range(1,n+1):
            f = f*k
        return f

>>> factorial(3)

    6
```

For the computation in the previous example, we now increase n until $2^{n-1}n!\geqslant 10^{16}$.

```
>>> n = 1

>>> while 2**(n-1) * factorial(n) < 10**16:
        n = n+1

>>> n

    15

>>> 2**(n-1) * factorial(n)

    21424936845312000

>>> 2.**(n-1) * factorial(n)

    2.1424936845312e+16
```

Example 96. `Python` Let us redo Example 88 but with Chebyshev nodes instead of equally spaced interpolation nodes.

```
>>> def chebyshev_nodes(n):
        return [cos((2*j+1)*pi/(2*n)) for j in range(n)]
>>> chebyshev_nodes(3)

    [0.8660254037844387, 6.123233995736766e-17, -0.8660254037844387]
```

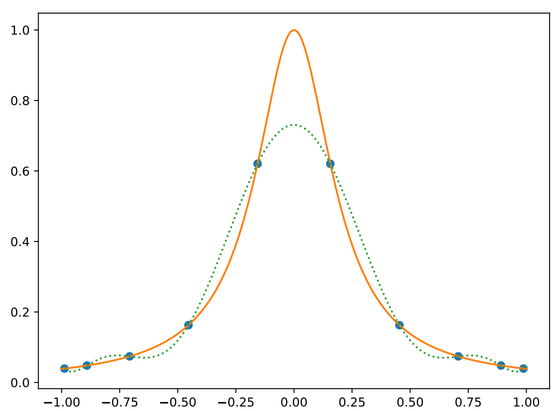
We observed in Example 88 that the maximal interpolation error for the Runge function $f(x) = 1/(1+25x^2)$ did not go down as we increased the number of (equally spaced) interpolation nodes.

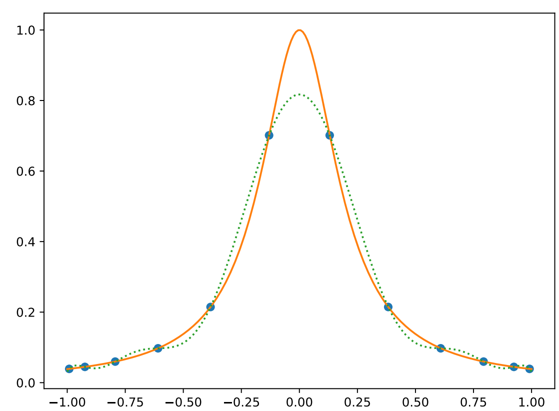
```
>>> def f(x):
        return 1/(1+25*x**2)
>>> [max_interpolation_error(f, -1, 1, chebyshev_nodes(n), 100) for n in range(2,18)]

    [0.92338165562264884, 0.60057189596611948, 0.74778034684079508, 0.40195613012685899,
    0.5534788672877784, 0.26410513077643449, 0.38946847488552683, 0.17006563147899745,
    0.26712486571968486, 0.10902564197574982, 0.1809557278548104, 0.06902642915187851,
    0.12185501126173093, 0.0460893689663045, 0.081815311151541836, 0.032580232210393967]
```

Unlike in Example 88, these values suggest that, by increasing the number of Chebyshev nodes, the maximal interpolation error will go to zero.

For comparison with Example 88, the following plots are for 10 and 12 interpolation nodes.





Comment. Note how we no longer have oscillations towards the endpoints of the interval. These plots also reveal why (as we can see from the above list of maximal errors) an even number of Chebyshev nodes leads to a relatively worse interpolation error compared to an odd number. (Namely, for an odd number of nodes, we have a node at $x = 0$, the peak of our function; while, for an even number of nodes, that peak is underestimated by the interpolation.)

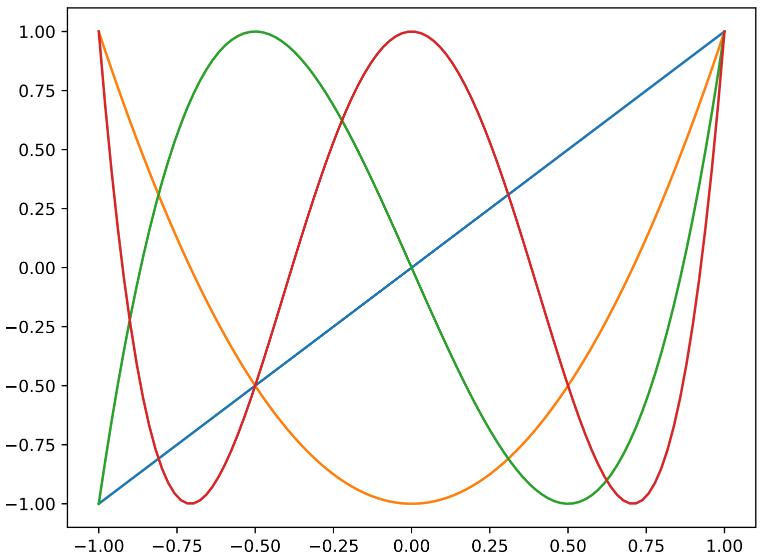
Chebyshev polynomials

As introduced after Chebyshev's Theorem 91, the **Chebyshev polynomials** of the first kind are

$$T_n(x) = 2^{n-1}(x - x_1)\cdots(x - x_n), \quad x_j = \cos\left(\frac{(2j-1)}{2n}\pi\right).$$

These are scaled by 2^{n-1} so that the maximum is 1. Indeed, $T_n(1) = 1$.

We can see in the following plot of $T_n(x)$ for $n \in \{1, 2, 3, 4\}$ that the Chebyshev polynomials alternate between the values ± 1. The goal of this section is to prove this and other properties.



Review. (trig identities through Euler) By **Euler's identity**, $e^{i\theta} = \cos(\theta) + i\sin(\theta)$. In other words, $\cos(\theta) = \mathrm{Re}(e^{i\theta})$ and $\sin(\theta) = \mathrm{Im}(e^{i\theta})$ are "parts" of the exponential function.

All of the trig function identities can then be obtained from simpler identities of the exponential function. For instance, the exponential function satisfies $e^{A+B} = e^A e^B$. For the cosine, this relation translates into

$$\cos(\alpha + \beta) = \mathrm{Re}(e^{i\alpha}e^{i\beta}) = \mathrm{Re}(\cos(\alpha) + i\sin(\alpha)(\cos(\beta) + i\sin(\beta))) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta).$$

Theorem 97. The **Chebyshev polynomials** $T_n(x)$ of the first kind satisfy:

(a) $T_n(\cos(\theta)) = \cos(n\theta)$

Equivalently, $T_n(x) = \cos(n \arccos(x))$.

(b) $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$

This Fibonacci-like recursive relation together with $T_0(x) = 1$ and $T_1(x) = x$ characterizes $T_n(x)$.

Proof.

(a) It follows from trig identities (see the first part of the next example) that $\cos(n\theta)$ can be written as a polynomial in $\cos(\theta)$. In other words, there is a (unique) polynomial $p_n(x)$ such that $\cos(n\theta) = p_n(\cos(\theta))$. We need to show that $T_n(x) = p_n(x)$. Since both are polynomials of degree n, this follows if we can show that they agree at $n+1$ points.

By definition, for $j \in \{1, 2, \ldots, n\}$, $T_n(x)$ has a root at $x_j = \cos(\theta_j)$ where $\theta_j = \frac{(2j-1)}{2n}\pi$.

On the other hand, $p_n(x_j) = p(\cos(\theta_j)) = \cos(n\theta_j) = \cos\left(\left(j - \frac{1}{2}\right)\pi\right) = 0$.

$T_n(x)$ and $p_n(x)$ therefore have the same n roots. It follows that they are the same if they have the same leading coefficient. For $T_n(x)$ it is clear from the definition $T_n(x) = 2^{n-1}(x - x_1)\cdots(x - x_n)$ that the leading coefficient is 2^{n-1}. That the same is true for $p_n(x)$ follows from the recursive relation for $p_n(x)$ that we show in the second part.

(b) It follows from the trig identity $\cos(\alpha+\beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta)$ (which we derived above) that

$$\begin{aligned}\cos((n+1)\theta) &= \cos(n\theta+\theta) = \cos(n\theta)\cos(\theta) - \sin(n\theta)\sin(\theta),\\ \cos((n-1)\theta) &= \cos(n\theta-\theta) = \cos(n\theta)\cos(\theta) + \sin(n\theta)\sin(\theta),\end{aligned}$$

where we used that $\sin(-\theta) = -\sin(\theta)$ for the last term. Adding these two, and then writing $T_n(x) = \cos(n\theta)$ with $\theta = \arccos(x)$, we obtain

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

which is the claimed recursive relation. □

Example 98. Determine the first few Chebyshev polynomials $T_n(x)$.

Solution. (using cosines) We use $T_n(x) = \cos(n\theta)$ with $x = \cos(\theta)$ combined with Euler's identity $e^{i\theta} = \cos(\theta) + i\sin(\theta)$ as well as the trig identity $\cos(\theta)^2 + \sin(\theta)^2 = 1$.

- $e^{2i\theta} = (e^{i\theta})^2 = (\cos(\theta) + i\sin(\theta))^2$ has real part $\cos(2\theta) = \cos(\theta)^2 - \sin(\theta)^2 = 2\cos(\theta)^2 - 1$.
  Hence $T_2(x) = 2x^2 - 1$.
- $e^{3i\theta} = (\cos(\theta) + i\sin(\theta))^3$ has real part $\cos(3\theta) = \cos(\theta)^3 - 3\cos(\theta)\sin(\theta)^2 = 4\cos(\theta)^3 - 3\cos(\theta)$.
  Hence $T_3(x) = 4x^3 - 3x$.

Solution. (using recursion) Starting with $T_0(x) = 1$ and $T_1(x) = x$, we apply $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$ to compute $T_2(x), T_3(x), \ldots$

- $T_2(x) = 2xT_1(x) - T_0(x) = 2x^2 - 1$
- $T_3(x) = 2xT_2(x) - T_1(x) = 2x(2x^2-1) - x = 4x^3 - 3x$
- $T_4(x) = 2xT_3(x) - T_2(x) = 2x(4x^3-3x) - (2x^2-1) = 8x^4 - 8x^2 + 1$
- ...

Armin Straub
straub@southalabama.edu