Chebyshev_Polynomials

April 14, 2021

[1]:
```python
__author__ = "@Tssp"
__date__   = "14/04/21"
import sympy as sp
from sympy.functions.special.polynomials import chebyshevt
from sympy.abc import n, x
import matplotlib.pyplot as plt
import numpy as np
```

The Chebyshev polynomial of first kind, $T_k(x)$, is the polynomial of degree k defined in the range $x \in [-1, 1]$ by the relation

$$T_k(x) = cos(k\ arccos(x))$$

Therefore, $-1 \leq T_k(x) \leq 1$. By setting $x = cos(z)$, we have

$$T_k(z) = cos(kz)$$

$n = 0$

[2]:
```python
chebyshevt(0, x)
```

[2]: 1

$n = 1$

[3]:
```python
chebyshevt(1, x)
```

[3]: x

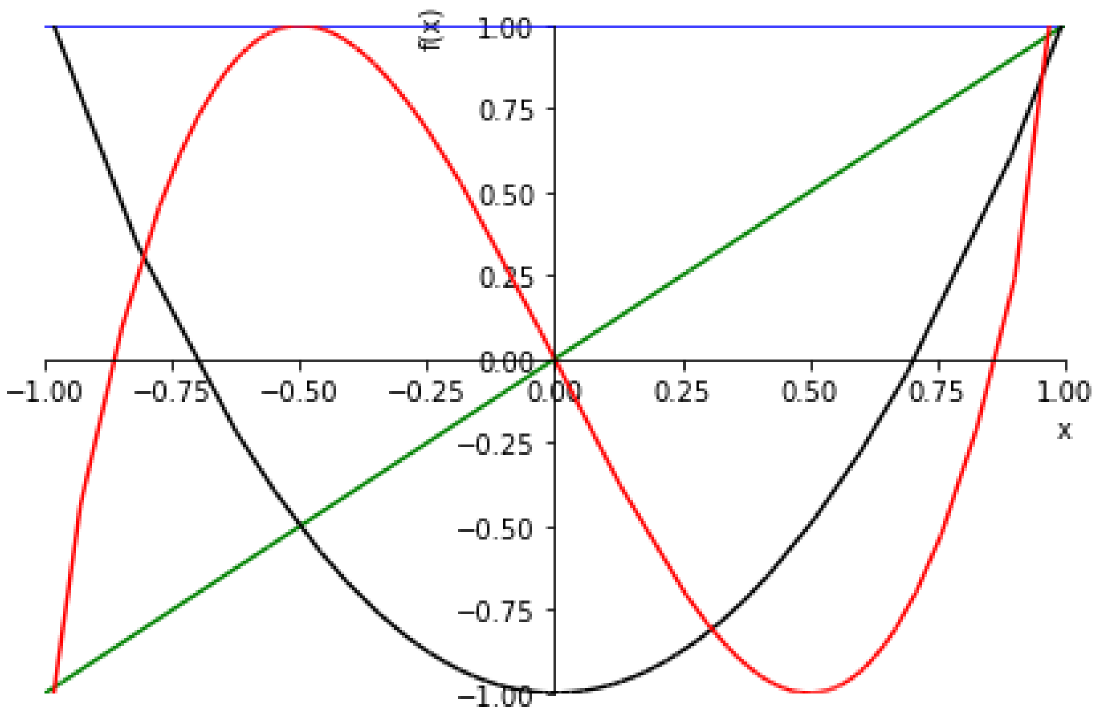
$n = 2$

[4]:
```python
chebyshevt(2, x)
```

[4]: $2x^2 - 1$

[5]:
```python
p0 = sp.plotting.plot(chebyshevt(0, x), xlim=(-1, 1), ylim=(-1,1), show=False,␣
 ↪line_color='b')
p1 = sp.plotting.plot(chebyshevt(1, x), xlim=(-1, 1), ylim=(-1,1), show=False,␣
 ↪line_color='g')
p2 = sp.plotting.plot(chebyshevt(2, x), xlim=(-1, 1), ylim=(-1,1), show=False,␣
 ↪line_color='k')
```

```
p3 = sp.plotting.plot(chebyshevt(3, x), xlim=(-1, 1), ylim=(-1,1), show=False,␣
 ↪line_color='r')
p0.append(p1[0])
p0.append(p2[0])
p0.append(p3[0])
p0.show()
```



From the trigonometric identity

$$cos(k+1)z + cos(k{-}1)z = 2cos(z)cos(kz),$$

it can be obtained the recurrence relationship

$$T_{k+1}(x) = 2xT_k(x){-}T_{k-1}(x), k\ 1$$

which allows to deduce, in particular, the expression of the polynomials T_k, $k\ 2$ from the knowledge of T_0 and T_1.

The Chebyshev polynomials constitute a complete basis of orthogonal polynomials in the interval $[-1, 1]$ with weight $\frac{1}{\sqrt{1-x^2}}$,

$$\int_{-1}^{1} dx \frac{T_i(x)T_j(x)}{\sqrt{1-x^2}} = \frac{\pi}{2} a_i \delta_{ij}$$

,

where $a_i = 2$ for $i = 0$ and 1 for $i > 0$

[6]:
```
sp.integrate(chebyshevt(0, x) * chebyshevt(0, x)/sp.sqrt(1 - x**2), (x, -1, 1))
```

[6]:

π

[7]:
```
sp.integrate(chebyshevt(1, x) * chebyshevt(1, x)/sp.sqrt(1 - x**2), (x, -1, 1))
```

[7]: $\frac{\pi}{2}$

[8]:
```
sp.integrate(chebyshevt(0, x) * chebyshevt(1, x)/sp.sqrt(1 - x**2), (x, -1, 1))
```

[8]:

0

The completeness property tells us that in general any function $f(x)$ can be represented as a linear combination of Chebyshev polynomials,

$$\sum_{m=0}^{M} c_m T_m(x)$$

Then, making use of the orthogonality of the Chebyshev polynomials the expansion coefficients cm can be evaluated from

$$c_m = \frac{2}{\pi a_m} \int_{-1}^{1} dx \frac{f(x) T_m(x)}{\sqrt{1 - x^2}}$$

[9]:
```
def Chebyshev_Expansion(f, order):
    '''
    Parameters
    ----------
    f: function to expand in terms of the Chebyshev polynomials f(x)
    m: order of expansion

    Returns
    -------
    C: Expanded function
    '''
    C = 0
    for m in range(order):
        if m == 0: am = 2
        elif m>0: am=1
        cm = 2/(np.pi*am) * sp.integrate(f*chebyshevt(m, x)/sp.sqrt(1 - x**2),␣
↪(x,-1,1)).evalf()
        C += cm * chebyshevt(m, x)
    print(C)
    return C
```

[10]:
```
C_exp = Chebyshev_Expansion(sp.exp(x), 3)
```

```
0.542990679068153*x**2 + 1.13031820798497*x + 0.994570538217932
```

[11]:
```
p1 = sp.plotting.plot(C_exp, xlim=(-1, 1), ylim=(0,4), show=False,␣
 ↪line_color='b')
p2 = sp.plotting.plot(sp.exp(x), xlim=(-1, 1), ylim=(0,4), show=False,␣
 ↪line_color='g')
p1.append(p2[0])
p1.show()
```