

exercise_02

January 30, 2021

1 Polynomial interpolation

Stefano Decio

Date: **Feb 8, 2021** Submission deadline: **Jan 25, 2021**

If you want to have a nicer theme for your jupyter notebook, download the [cascade stylesheet file tma4125.css](#) and execute the next cell:

```
[1]: # from IPython.core.display import HTML
def css_styling():
    try:
        with open("tma4125.css", "r") as f:
            styles = f.read()
            return HTML(styles)
    except FileNotFoundError:
        pass #Do nothing

# Comment out next line and execute this cell to restore the default notebook
→style
css_styling()
```

We import the required modules, and run the lectures notebooks to import the functions used to construct interpolation polynomials.

```
[2]: import numpy as np
import matplotlib.pyplot as plt
from math import factorial
newparams = {'figure.figsize': (8.0, 4.0), 'axes.grid': True,
             'lines.markersize': 8, 'lines.linewidth': 2,
             'font.size': 14}
plt.rcParams.update(newparams)

from polynomialinterpolation import *
```

2 Obligatory exercises

2.1 1)

Consider the data points

x_i	-2	-1	1	2
$f(x_i)$	-3	3/4	-3/2	3/2

a) Find both the Lagrange and Newton form of the minimal degree interpolating polynomial at these points. Verify that the polynomials you get coincide. Use this to find an approximation to $f(0)$.

b) Confirm your results numerically, e.g. using the functions above.

3 2)

The population of Norway in the period from 1993 to 2018 is, according to SSB

year	1994	1999	2004	2009	2014	2019
population	4,341,615	4,474,004	4,598,214	4,826,848	5,142,265	5,378,857

Use the interpolation polynomial, through the functions cardinal and lagrange, to estimate the population in the years 2001 and 2011. Predict the population in 2030 and 2040. Comment on the results. (The population in 2001 was 4,523,145, in 2011 it was 4,948,330).

3.1 3)

Consider the function $f(x) = x^2 \sin(x)$.

a) Calculate analytically the polynomial of degree 3 that interpolates $f(x)$ in the interval $[0, 3]$ at 4 equally distributed nodes, where 2 of the nodes are the interval endpoints.

b) Find the 4 Chebyshev nodes in the interval $[0, 3]$ (you do not need to calculate the interpolating polynomial by hand in this case).

c) Find by hand a bound for the maximal interpolation error in the interval $[0, 3]$ in both cases, equidistributed nodes and Chebyshev.

d) Confirm your tests numerically. More specifically, plot the function and the interpolation polynomial, measure the interpolation error and compare it with your theoretical results.

Hint for part c):

$$\frac{d^n}{dx^n} x^2 \sin(x) = \begin{cases} (-1)^{n/2} (x^2 \sin(x) - 2nx \cos(x) - n(n-1) \sin(x)) & n \text{ even.} \\ (-1)^{(n-1)/2} (x^2 \cos(x) + 2nx \sin(x) - n(n-1) \cos(x)) & n \text{ odd.} \end{cases}$$

4 Recommended exercises

4.1 4)

Inverse interpolation. Given a sufficiently continuous function $f(x)$ with a root r in some interval $[a, b]$. Choose this sufficiently small to ensure f to be strict monotonically increasing or decreasing in $[a, b]$. We will use the fact that

$$f(r) = 0 \quad \Rightarrow \quad r = f^{-1}(0).$$

Choose $n + 1$ distinct data points (x_i, y_i) in the interval $[a, b]$, and find the interpolation polynomial $p_n(y) \approx f^{-1}(y)$ and $r \approx p_n(0)$.

a) Let $f(x) = x^3 - 7$ and $[a, b] = [1.5, 2.0]$. As nodes, choose $(x_i) = (1.5, 1.75, 2.0)$ and use the idea above to find an approximation to the root. How close to the exact solution is the approximation?

b) Repeat the example above, but now with $n + 1$ uniformly distributed nodes over the interval $[1.5, 2.0]$. Use the functions **cardinal** and **lagrange** to calculate the value of the polynomial $p_n(0)$. Choose $n = 2$ (to check your hand calculations), 4, 8, and 16. Find the approximation in each case, as well as the error.

NB! (Python): Even if you only want to calculate the value of the cardinal functions in one point, 0, it still has to be given as an array (`array([0])`) in the function **cardinal**.

[]: