N-Step Factorial - Journal

Adam Lawson

April 2022

1 Week 01: 4/17-4/23

1.1 Description of Work

(BC) What is the "original" definition of the NSF(x, n) function?

I have worked out a more algebraic expression of the NSFs using what we already know which is:

$$\text{NSF}(x, 1) = x!$$

We can then express the function as

$$\text{NSF}(x, n) = x * (x - n) * (x - 2n) * (x - 3n)...(x - m * n)$$
$$\text{where it satisfies } x - m * n > 0 \text{ and } x - (m + 1) * n \leq 0$$

The number of polynomials used is determined by this formula:

$$s = \lceil x/n \rceil$$

Is this m? i.e. Is $m = s$?

Date	Hours
4/18	1.00
4/20	4.00
4/21	2.00

2 Week 02: 4/24-4/30

2.1 Description of Work

Complex adaptation of NSF
Other ways to express NSF letting m=x/n

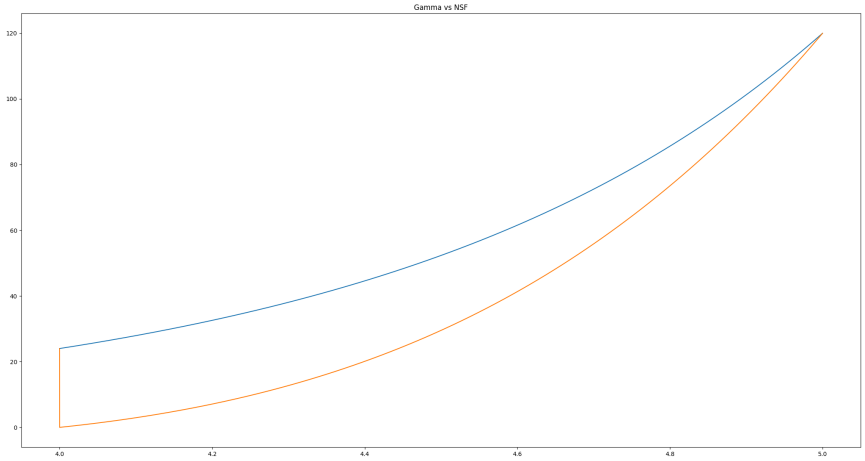
$$\begin{aligned} \text{NSF}(x,n) &= n^m m! \\ &= n^{\lceil x/n \rceil} * \text{NSF}(x/n, 1) \\ \text{If x/n is an integer, NSF}(x,n) &= n^{\lceil x/n \rceil}(x/n)! \\ &= n^{\lceil x/n \rceil} * \Gamma(x/n+1) \end{aligned}$$

Date	Hours
4/24	4.00
4/25	2.00
4/27	3.00
4/30	2.00

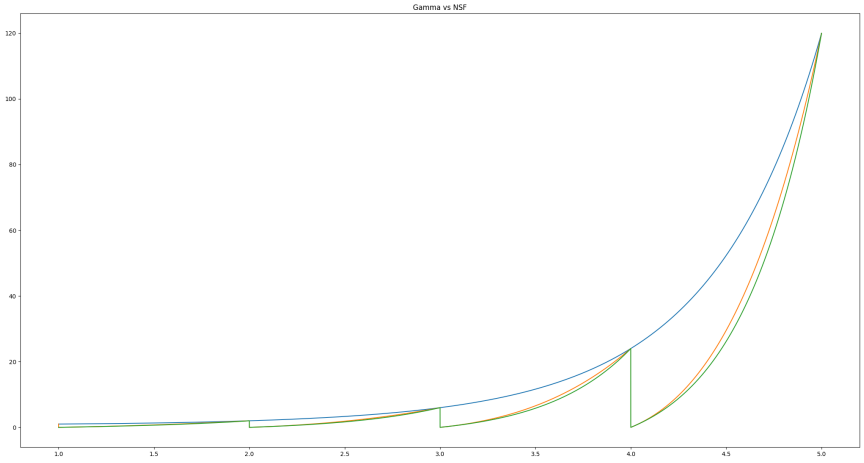
3 Week 03: 5/1-5/7

3.1 Description of Work

Comparing the gamma function and NSF of step 1.



Blue is Gamma and Orange is NSF

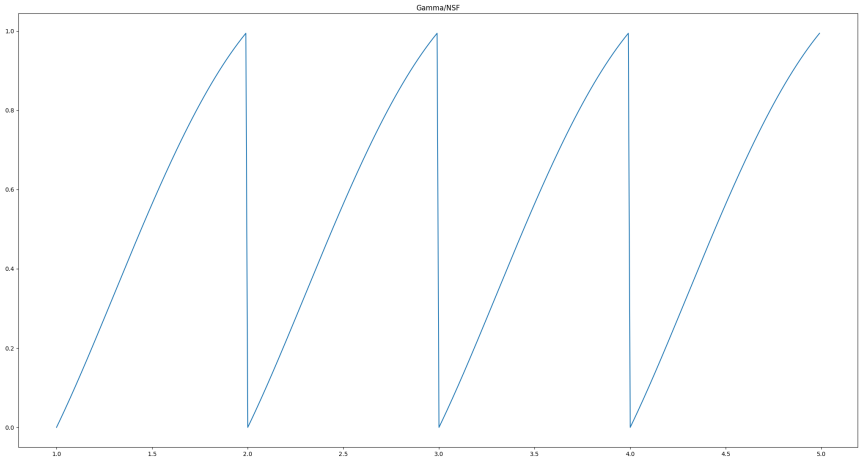


Approximating the solution as $(x/n - \lfloor x/n \rfloor)\Gamma(x+1)$
Blue is Gamma, Orange is NSF and Green is the approximation
The approximation can also be represented as $((x/n) \mod 1) * \Gamma(x+1)$

Date	Hours
5/2	2.00
5/3	2.00
5/4	3.00
5/6	2.00

4 Week 04: 5/8-5/14

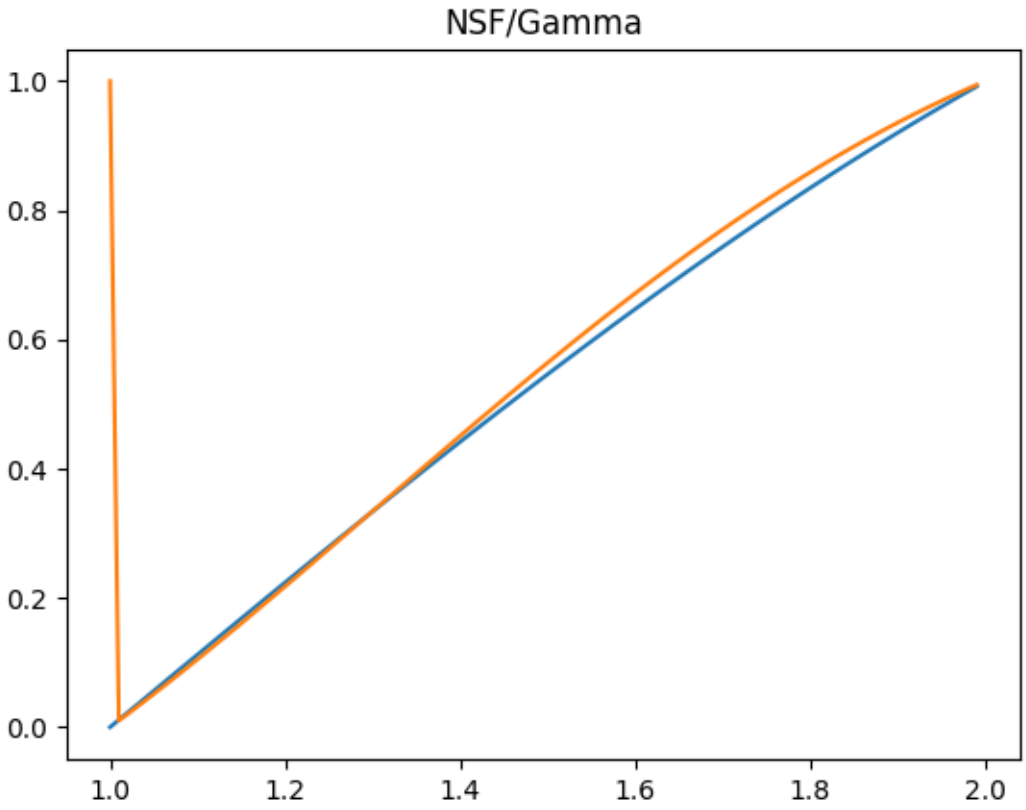
Solving for $g(x)$ in $g(x)\Gamma(x) = \text{NSF}(x, 1)$
We know $g(x)$ is periodic and is in a form x^a.
Note that a is not a constant nor is it linear!



Date	Hours
5/9	4.00
5/10	2.00
5/14	3.00

5 Week 05: 5/15-5/21

Proving that g(x) is periodic in Paper.tex.
Approximating the curve with Sine functions.
Creating a g(x) matrix to approximating g(x).
The Python program is not complete to correctly approximate g(x). Working out bugs.



Orange is NSF blue is $1.4 * Sin(x * \pi/3.2)$

Date	Hours
5/16	4.00
5/17	1.00
5/20	4.00

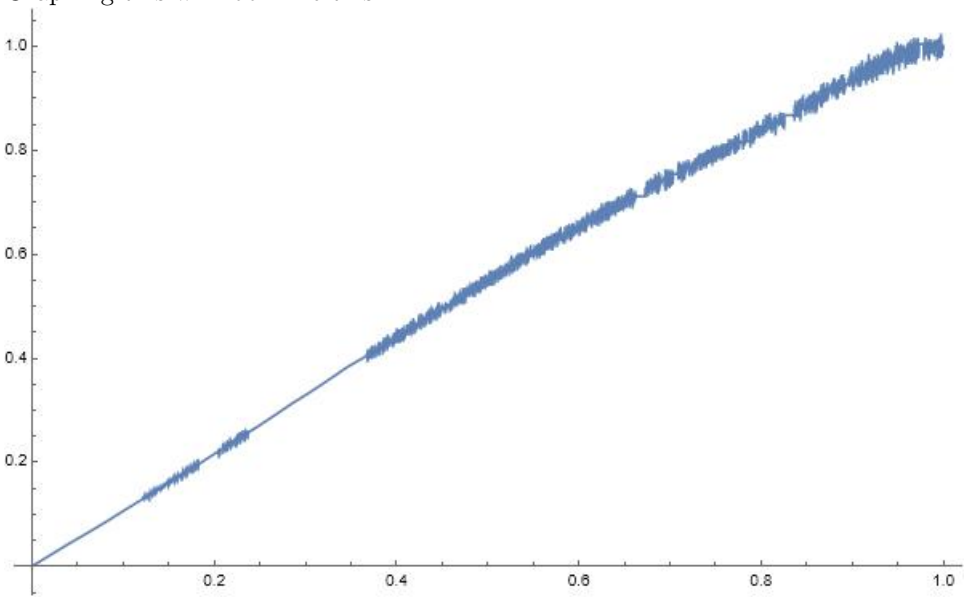
6 Week 06: 5/22-5/28

Debugging the python code for Calculating g(x) and the approximation of the curve.

7 Week 07: 5/29-6/4

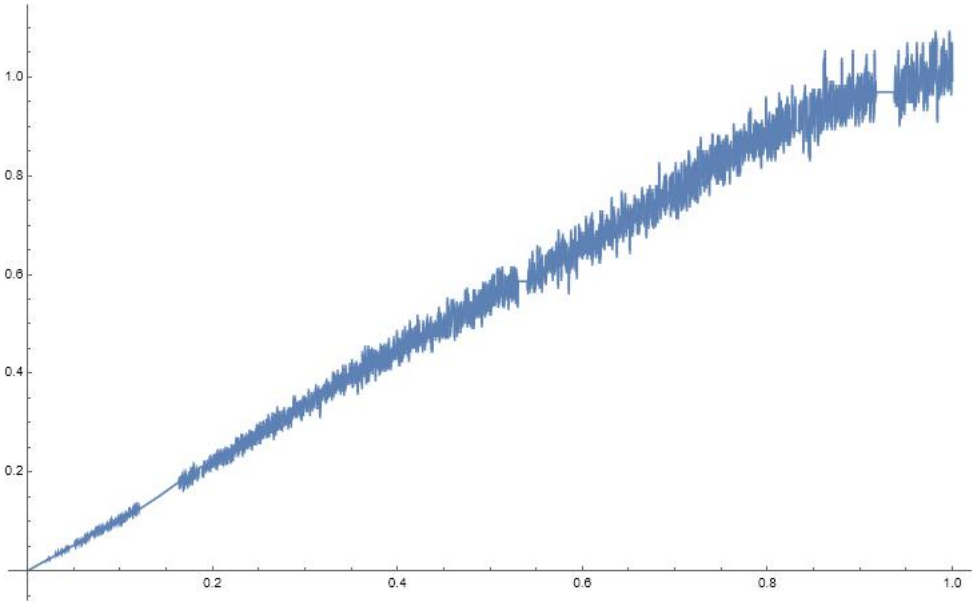
Fourier Approximation for 10 terms is complete, however, the main issue is that it is off by around 0.02 on average.
$C_1 = 6.617703335740171 * 10^{12}$
$C_2 = -2.77132145301574 * 10^{13}$
$C_3 = 3.962243416540127 * 10^{13}$
$C_4 = -1.054928489449636 * 10^{13}$
$C_5 = -3.364981577515149 * 10^{13}$
$C_6 = 4.483340969148211 * 10^{13}$
$C_7 = -2.564196403297839 * 10^{13}$
$C_8 = 7.342270972984537 * 10^{12}$
$C_9 = -8.61538932823401 * 10^{11}$
$C_{10} = -0.03664057375873128$

Graphing this will look like this:



8 Week 08: 6/5-6/11

Fourier Approximation for 20 terms. Looking at how it progresses, it seems that we have the wrong function to approximate the curve or the Fourier Series created is losing its accuracy.


1.0
0.8
0.6
0.4
0.2
0.2
0.4
0.6
0.8
1.0