

# Calculating Functions with Taylor Polynomials

Thomas Marshall

March 2023

## Contents

<b>1</b>	<b>Trigonometric Functions</b>	<b>2</b>
1.1	Sine and Cosine . . . . .	2
1.2	Inverse Tangent . . . . .	3
1.3	Other Trig Functions . . . . .	4
<b>2</b>	<b>Logarithms</b>	<b>4</b>
2.1	Natural Log . . . . .	4
2.2	Other Bases . . . . .	5
<b>3</b>	<b>Exponential</b>	<b>5</b>
3.1	Natural Exponential . . . . .	5
3.2	Other Bases . . . . .	5
<b>4</b>	<b>Implementation</b>	<b>5</b>

# 1 Trigonometric Functions

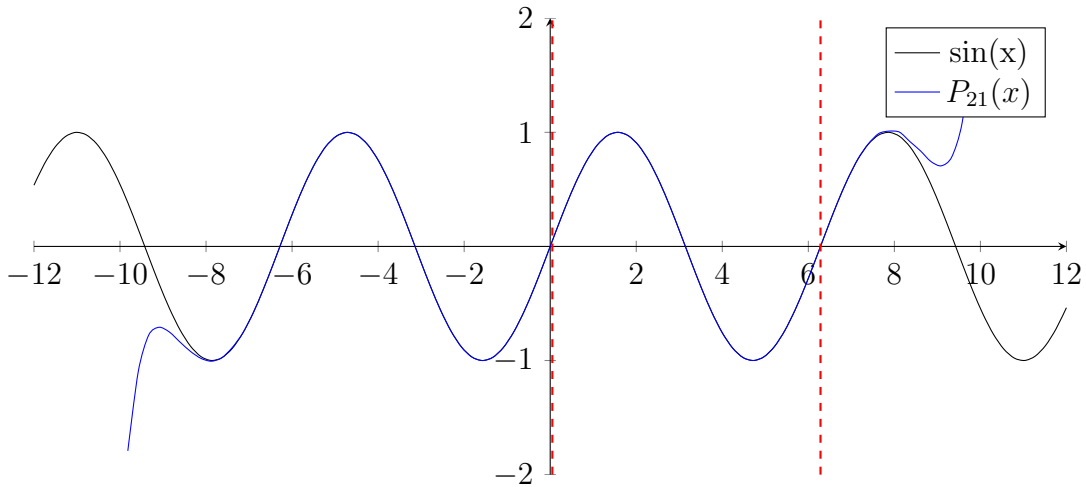
## 1.1 Sine and Cosine

We can use the Maclaurin Series for sine and cosine and calculate several terms of the series using a for loop. The series are shown below.

$$\sin(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!}$$

$$\cos(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}$$

With just 11 terms (21st degree Taylor polynomial), the sums above will be extremely accurate for small inputs. However, the margin of error increase quickly as we use larger inputs.



For better accuracy with larger inputs, we need to transform all inputs to be in the range  $0 \leq x < 2\pi$  (shown by the dotted lines). Because sine and cosine have a period of  $2\pi$ , the below is true.

$$x_m = x \bmod 2\pi$$

$$\sin(x) = \sin(x_m)$$

$$\cos(x) = \cos(x_m)$$

Instead of calculating  $\sin(x)$  and  $\cos(x)$  directly, we can calculate  $\sin(x_m)$  and  $\cos(x_m)$ .

## 1.2 Inverse Tangent

It is possible to construct an infinite series that converges to  $\arctan(x)$  and use it to accurately calculate values of  $\arctan(x)$ . First take the derivative and then rewrite.

$$\frac{d}{dx} \arctan(x) = \frac{1}{1+x^2} = \frac{1}{1-(-x^2)}$$

Using the formula for the sum of an infinite geometric series gives us the following.

$$\frac{1}{1-(-x^2)} = 1 - x^2 + x^4 - x^6 + \dots + (-x^2)^n + \dots$$

$$\begin{aligned} \arctan(x) &= \int (1 - x^2 + x^4 - x^6 + \dots + (-x^2)^n + \dots) dx \\ &= x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{2n+1} \end{aligned}$$

We can use a for loop to evaluate  $\arctan(x)$  with 40 terms for pretty good accuracy. However, even though the interval of convergence of this series is  $-1 \leq x \leq 1$ , the accuracy greatly decreases for  $x$  values above 0.7.

The solution is to reduce the input to our  $\arctan$  function to be less than 0.7. The identity below will be extremely helpful.

$$\arctan(x) = \arctan(c) + \arctan(b)$$

$$b = \frac{x-c}{1+cx}$$

If we set  $c$  to 0.5, then  $b$  will always be between -0.5 and 0.5, if  $0.7 \leq x \leq 1$ . This is perfect because the Taylor Series is extremely accurate in the range  $-0.5 \leq x \leq 0.5$ . For our final algorithm, we will return the Taylor series approximation if  $x < 0.7$ . Otherwise, we will use the identity. The final function will look like this.

$$\begin{aligned} p(x) &= \sum_{n=0}^{40} (-1)^n \frac{x^{2n+1}}{2n+1} \\ f(x) &= \begin{cases} p(x) & \text{if } x \leq 0.7 \\ \arctan(0.5) + \arctan\left(\frac{x-0.5}{1+0.5x}\right) & \text{if } x > 0.7 \end{cases} \end{aligned}$$

## 1.3 Other Trig Functions

Since we can already calculate  $\sin(x)$  and  $\cos(x)$ , it is easy to calculate other trig functions using basic trig identities.

$$\tan(x) = \frac{\sin(x)}{\cos(x)} \quad \cot(x) = \frac{\cos(x)}{\sin(x)} \quad \csc(x) = \frac{1}{\sin(x)} \quad \sec(x) = \frac{1}{\cos(x)}$$

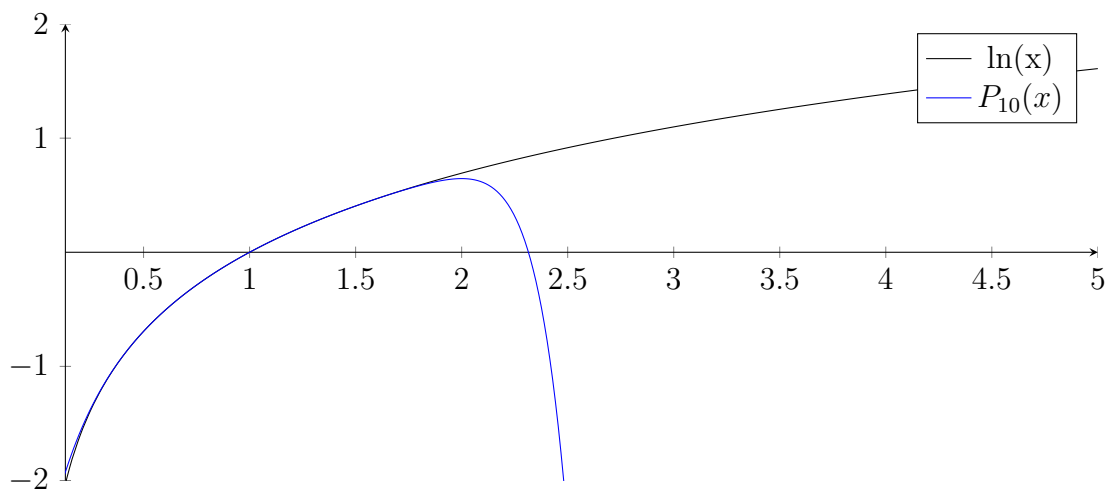
## 2 Logarithms

### 2.1 Natural Log

The Taylor series for  $\ln(x)$  centered around  $x = 1$  is shown below.

$$\ln(x) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1} (x-1)^n}{n}$$

Because the range of convergence of this series is  $0 < x < 2$ , it cannot be used to calculate values outside of that range. Additionally, since the accuracy of this series decreases as the inputs increase, we should limit the inputs to be less than about 1.5.



Since we want to be able to calculate any value for  $x > 0$ , we need a way to transform the inputs to be inside the desired interval. The below is true because of log rules.

$$\ln(ab) = \ln(a) + \ln(b)$$

$$\ln(a^b) = b \ln(a)$$

We can rewrite the  $\ln$  function using this rule.

$$\ln(x) = \ln\left(\frac{1.4^b x}{1.4^b}\right) = \ln\left(\frac{x}{1.4^b}\right) + \ln(1.4^b) = \ln\left(\frac{x}{1.4^b}\right) + b \ln(1.4)$$

Where  $b$  is the smallest positive integer such that  $1.4^b > x$ . We can easily find  $b$  in code by starting with a variable equal 1, and multiplying it by 1.4 until it's greater than  $x$ . And  $b$  is the number of times the loop was run.

## 2.2 Other Bases

Since we can calculate  $\ln(x)$  we can calculate logarithms for all bases using the change of base rule.

$$\log_b(x) = \frac{\log_d(x)}{\log_d(b)} \Rightarrow \log_b(x) = \frac{\ln(x)}{\ln(b)}$$

# 3 Exponential

## 3.1 Natural Exponential

The Maclaurin series for  $e^x$  is shown below. Using this formula with around 30 to 40 terms, it is simple to accurately calculate  $e^x$  for all  $x > 0$ .

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

## 3.2 Other Bases

To calculate  $b^x$ , we can use log rules to rewrite it using base  $e$ .

$$b^x = e^{\ln(b^x)} \Rightarrow b^x = e^{x \ln(b)}$$

Since we can already calculate  $e^x$  and  $\ln(x)$ , using this formula will only take a few lines of code.

# 4 Implementation

All the algorithms described can be seen at [this website](#). The code for these algorithms is written in TypeScript and can be seen on [GitHub](#).