Scientific Computing Lab

Kevin Francis

April 1, 2021

Equations Solved

In this lab we will be comparing $\sin(x)$ to the series representation for sin. The series for $\sin(x)$ is as follows:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - ..... + \frac{(-1)^n x^{2n-1}}{(2n-1)!} = \sum_{n=1}^{N} \frac{(-1)^n x^{2n-1}}{(2n-1)!}$$

Where N is the total number of terms in the series. The current term in the series is related to the next term by the following:

$$(n+1)th \text{ term} = \frac{-x^2}{(2n+1)(2n)} * nth \text{ term}$$

Numerical Methods

Part 1

The input argument for the function is the value of x. Your function should stop when the next term in the series will be no more than 10^{-8} of the sum up to that point. I did this by using the function below.

Code

```
def series(x):
    term = x; s = x; eps = 10**(-8)
    n = 1
    while x > 2 * pi:
        x = x - 2 * pi
    if x < 0:
        x = -x
    while abs(term/s) > eps:
        term = -term*x*x/(2*n+1)/(2*n)
        s = s + term
        n += 1
    return s
```

Part 2

Calculate the series for x $\leqslant$ 1 and compare it to the built-in function $math.sin(x)$ (you may assume that the built in function is exact). What I did for this part was sum up all of the values of sin from my series function verses the built in sin function and compare the sums. See code below.

Code

```
    X = 0.0
    Y = 0.0
    for i in arange(.01,1,.001):
        X = X + series(i)
        Y = Y + sin(i)
    print "sum of series for x <= 1 -->  " + str(X)
    print "sum of sin for x <= 1 -->  " + str(Y)
```

Part 3

Examine the terms in the series for $x \approx 3\pi$ and observe the significant subtractive cancellations that occur when large terms add together to give small answers. This was done by making a function this is a copy of the series function except it does not correct for large values of x and print every term. Below is the output where you can see terms almost canceling.

Output

```
#Term 10 for x = 3pi --> 5.64370915834
#Term 11 for x = 3pi --> -0.990779074809
#Term 12 for x = 3pi --> 0.146685874087
#Term 13 for x = 3pi --> -0.0185615354328
#Term 14 for x = 3pi --> 0.00203058225795
#Term 15 for x = 3pi --> -0.000193954721599
#Term 16 for x = 3pi --> 1.63154537324e-05
#Term 17 for x = 3pi --> -1.21790920353e-06
```

Part 4

See if better precision is obtained by using trigonometric identities to keep $0 \leqslant x \leq \pi$. This is implemented in the series function, please see code for part 1.

Part 5

By progressively increasing x from 1 to 10, and then from 10 to 100, use your program to determine experimentally when the series starts to lose accuracy and when its no longer converging. Below is the code and the output for the series of *sin* from 1 to 100.

Code

```
for i in range(1,10):
    print "Series for sin(" + str(i) + ")" + "---> " + str(series(float(i)))
for i in range(10,101):
    print "Series for sin(" + str(i) + ")" + "---> " + str(series(float(i)))
```

Output

```
#Series for sin(1)---> 0.841470984809
#Series for sin(2)---> 0.90929742683
#Series for sin(3)---> 0.141120008063
#Series for sin(4)---> -0.756802495379
#Series for sin(5)---> -0.958924274684
#Series for sin(6)---> -0.279415498204
#Series for sin(7)---> 6.41575324429
#Series for sin(8)---> 4.61020400505
#Series for sin(9)---> 1.36522611452
```

Around $x = 6$ is where the series begins to lose accuracy. The larger x gets the worse it becomes.

Visualization

Here is the code and images of the relative error verses the number of terms in the sequence.

Code

```
def graf(b,e,i):
    for x in arange(b,e,i):
        #while x > pi:
            #x = x - 2 *pi
        #if x < 0:
            #x = -x
        eps = 10**(-8)
        term = float(x); n = 1; s = float(x); y = [];
        while abs(term/s) > eps:
            term = -term*x*x/(2*n+1)/(2*n)
            s = s + term
            n += 1
            w = abs((s - sin(x)))
            y.append(w)
        x = arange(1,n)
        semilogy(x,y)
    xlabel('Number of terms')
    ylabel('Relative Error')
    title('Relative Error vs. Number of Terms')
    show()
```

Figure 1: for $1 \leqslant x \leqslant 10$

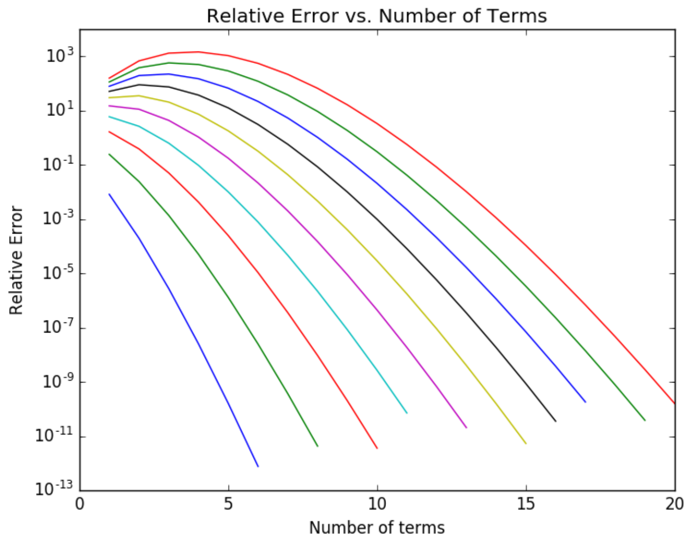
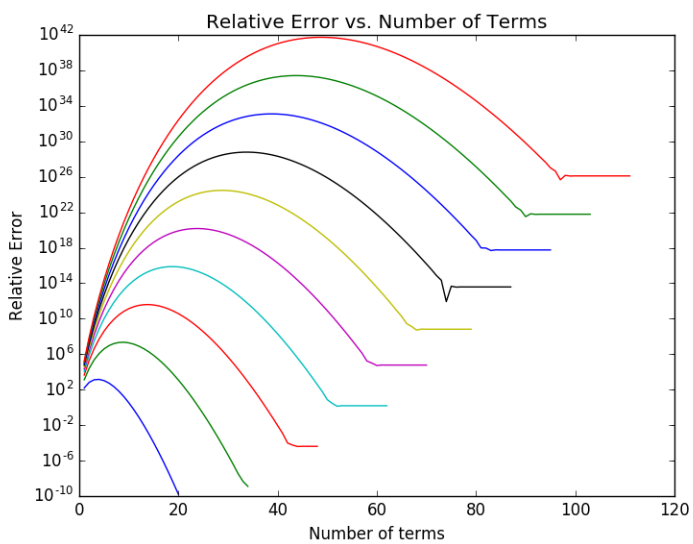


Figure 2: for $10 \leqslant x \leqslant 100$



.

Discussion

In this lab the series representing $\sin(x)$ was looked at. Figure 1 and figure 2 show the relative error verses the number of terms in the series. It can be seen that the larger x becomes the less accurate the series becomes. For these calculations the relative error used was the value of the series minus the value for the built in *sin* function.

Critique

During this lab there is something that the different parts lead to but is never explicitly stated. That is that the value of x must be small. This can be verified, especially in figure 2.