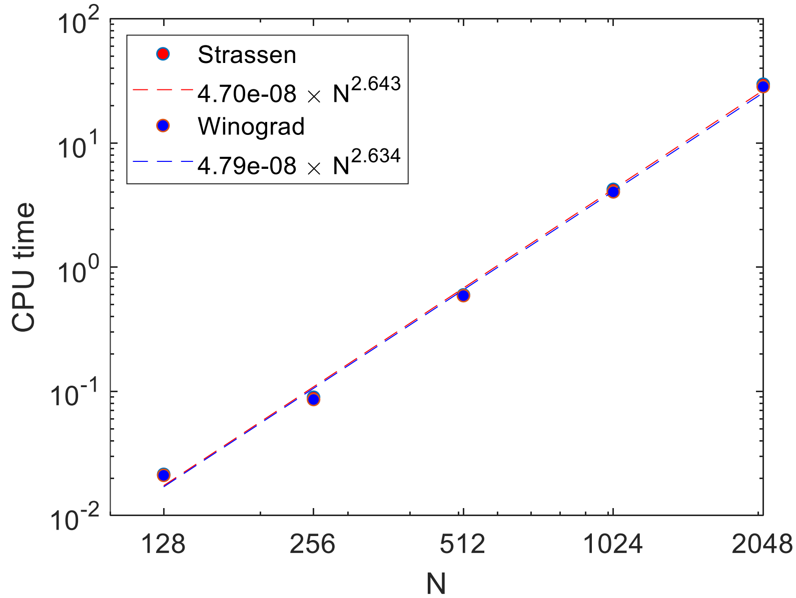
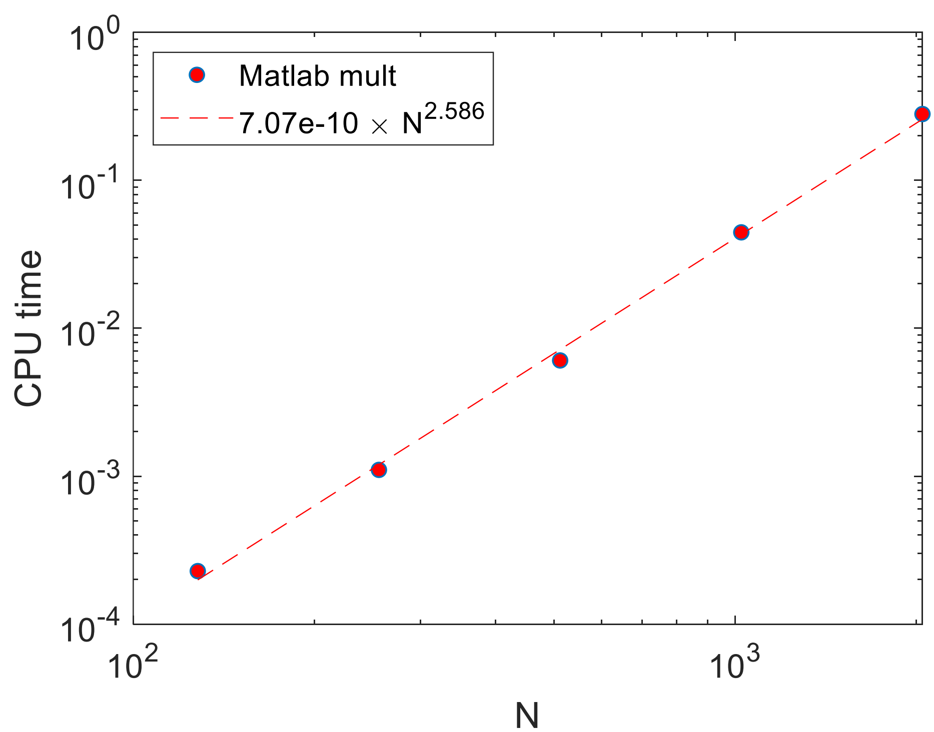
4X03: A3

1.


Strassen
4.70e-08 × N^2.643
Winograd
4.79e-08 × N^2.634
CPU time
N
128
256
512
1024
2048

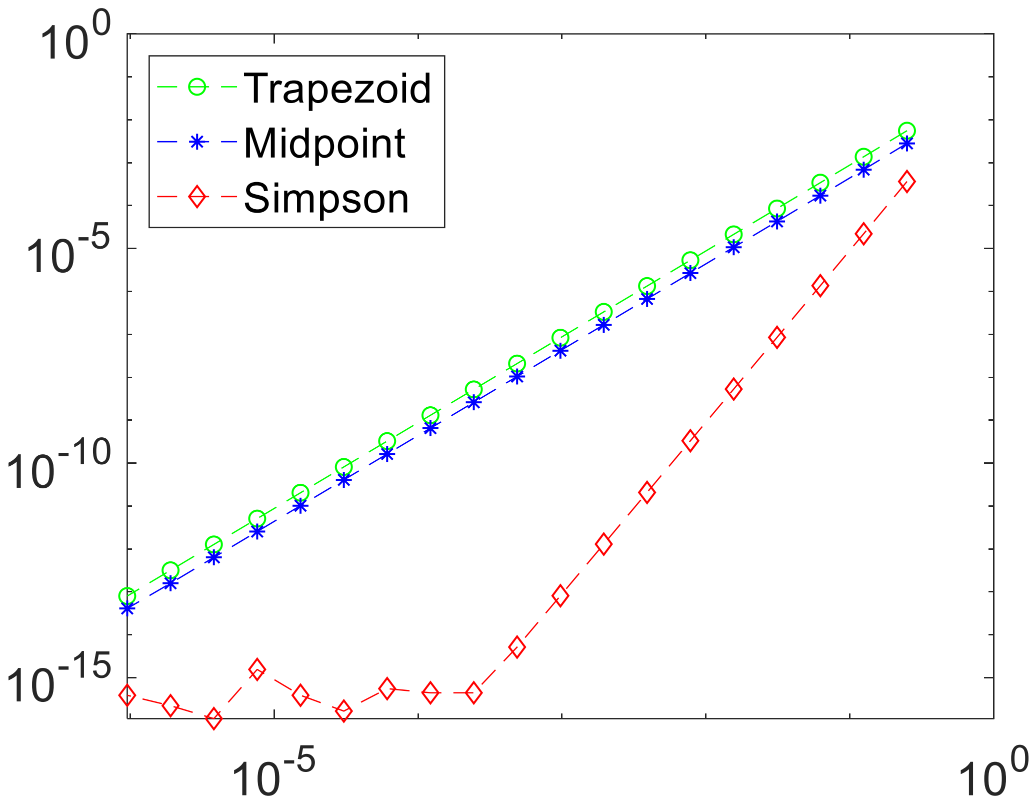

Matlab mult
7.07e-10 × N^2.586
CPU time
N

```
function [a,b,n,time] = bigOconstants(alg)
disp(alg)
n = 7:11;
n = 2.^n;
l = length(n);
time = zeros(1, l);
for i=1:length(n)
    A = rand(n(i));
    B = rand(n(i));
    if alg == 1
        tic
        A*B;
        time(i) = toc;

    elseif alg == 2
        tic
        strassen(A,B);
        time(i) = toc;

    elseif alg == 3
        tic
        strassenw(A,B);
        time(i) = toc;
    end

end

B = [ones(l,1), log(n)'];
coeff = B\log(time)';
a = exp(coeff(1));
b = coeff(2);
return
end
```

The a constants are much higher for the Strassen algorithms because while the overall b coefficient is lower, there is reduced numeric stability caused by the lowering of the b coefficient which is accounted for by utilizing a higher a coefficient to compensate.

2.

a. The two lines are parallel because both trapezoid and midpoint are of the same function order. They just have different constants, but the slopes of the lines are the same.
b. The slopes of trapezoid and midpoint are h^2 while Simpsons is irregular in this cash h^2.
c. The wiggles the error of simpson having a higher order derivative (4^{th}) which can cause the function to oscillate where as both trapezoid and midpoint only utilize a second order derivative in their error term result in less pronounced oscillation.




```
function [eT, eM, eS] = error_int(f, a, b, n, ref)
eT = zeros(1,length(n));
eM = zeros(1,length(n));
eS = zeros(1,length(n));
for i=1:length(n)
    eT(i) = abs(ref - trapezoid(f,a,b,n(i)));
    eM(i) = abs(ref - midpoint(f,a,b,n(i)));
    eS(i) = abs(ref - simpson(f,a,b,n(i)));
end
end


function val = trapezoid(f, a, b, n)
h = (b-a)/n;
x = linspace(a,b,n+1);
range = x(2:n);
sum_val = sum(f(range));
val = h/2*(f(a) + 2*sum_val + f(b));
end

function val = midpoint(f, a, b, n)
h = (b-a)/n;
x = linspace(a,b,2*n+1);
range = x(2:2:2*n);
val = h * sum(f(range));
end
```

```
function val = simpson(f, a, b, n)
h = (b-a)/n;
x = linspace(a,b,n+1);
range2 = x(3:2:n-1);
range4 = x(2:2:n);
sum2 = 2 * sum(f(range2));
sum4 = 4 * sum(f(range4));
val = h/3 * (f(a) + f(b) + sum2 + sum4);
end
```

3.

$f(x) = \sin(\pi x^2/3)$

$f''(x) = \left[-4x^2\pi^2 \sin\left(\frac{\pi x^2}{3}\right) - 6\pi \cos\left(\frac{\pi x^2}{3}\right)\right]/9$

max @ 1

$|f''(1)| = 2.757614$

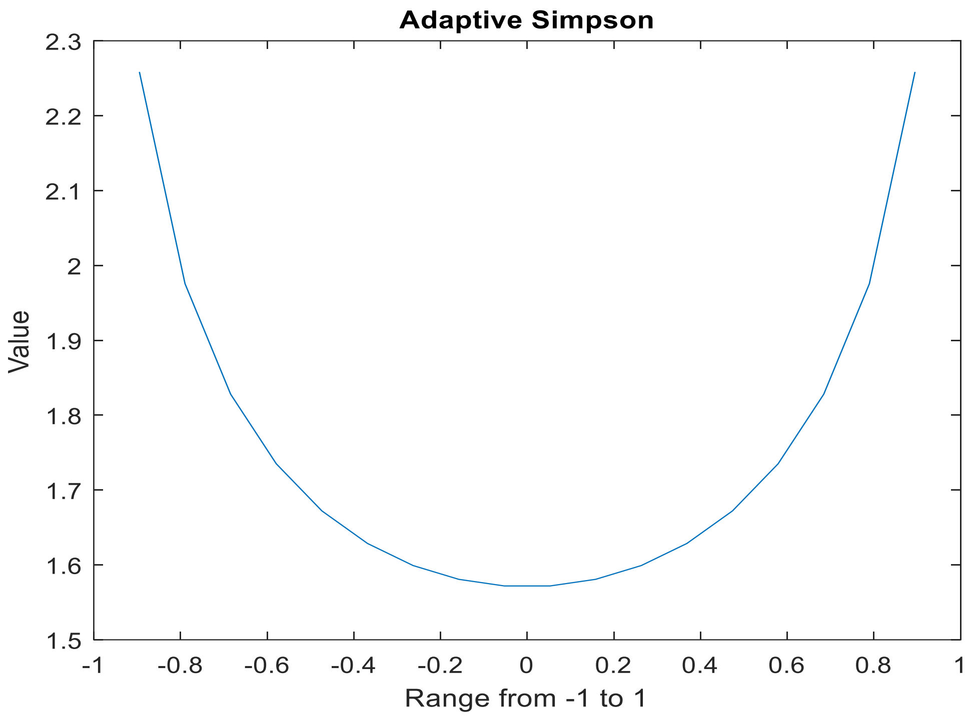
$\text{Error} = \frac{-f''(x)}{12}(b-a)h^2 \leq 10^{-10}$

$h = \frac{b-a}{n}$ $\quad h \leq 2.088 \times 10^{-5}$

$\frac{1-0}{n} \leq 2.088 \times 10^{-5}$

$47893 \leq n$

4.



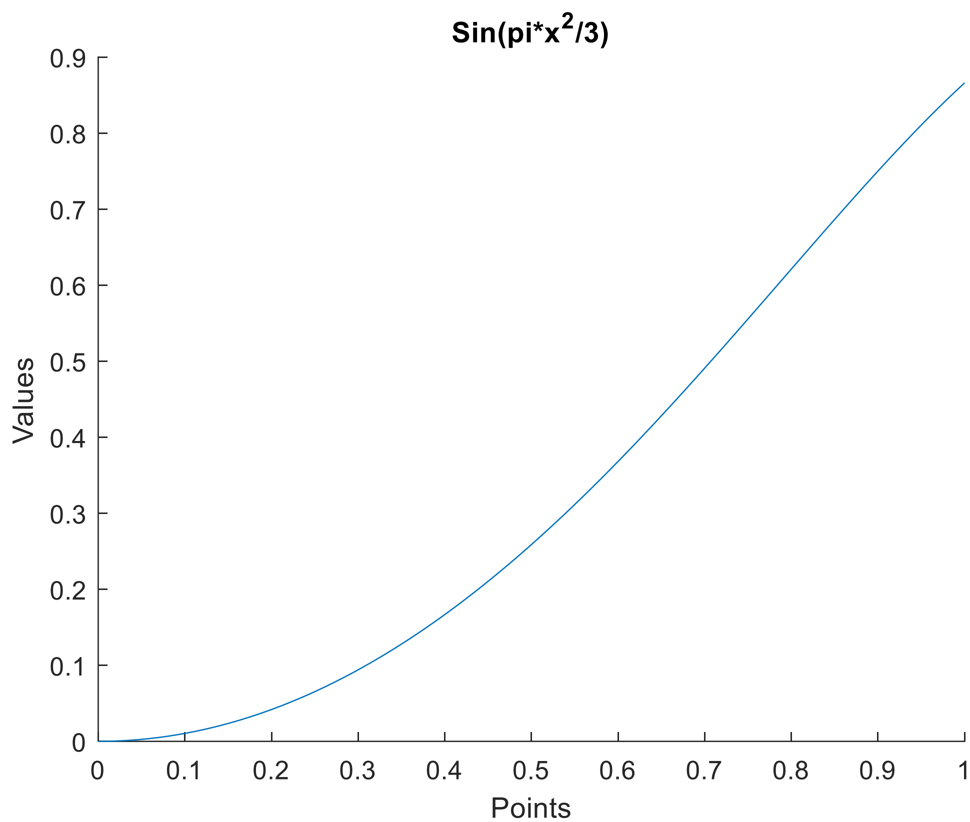
5.

```
function [S, count] = inner_integral(f, a, b, tol, count, count_max)
global xpoints
count = count + 1;
h = b - a;
c = (a + b) / 2;
d = (a + c) / 2;
e = (c + b) / 2;
syms x
fa = f(x,a);
fb = f(x,b);
fc = f(x,c);
fd = f(x,d);
fe = f(x,e);
S1 = h/6*(fa + 4*fc + fb);
S2 = h/12 * (fa + 4*fd + 2*fc + 4*fe + fb);
E2 = (S2 - S1)/15;
xpoints = unique([xpoints, a, b, c, d, e]);
if count >= count_max
S = S2 + E2;
else
    [Q1, count] = inner_integral(f, a, c, tol/2, count, count_max);
    [Q2, count] = inner_integral(f, c, b, tol/2, count, count_max);
    S = Q1 + Q2;
end
end
```

```
function Q = myquad2d(fun, a, b, c, d, tol)
f = matlabFunction(inner_integral(fun,c,d,tol,0,20));
Q = adaptive_simpson(f,a,b,tol, 0, 20);
end
```

Error from script = 1.1277e-07

6.



C1 = 16651

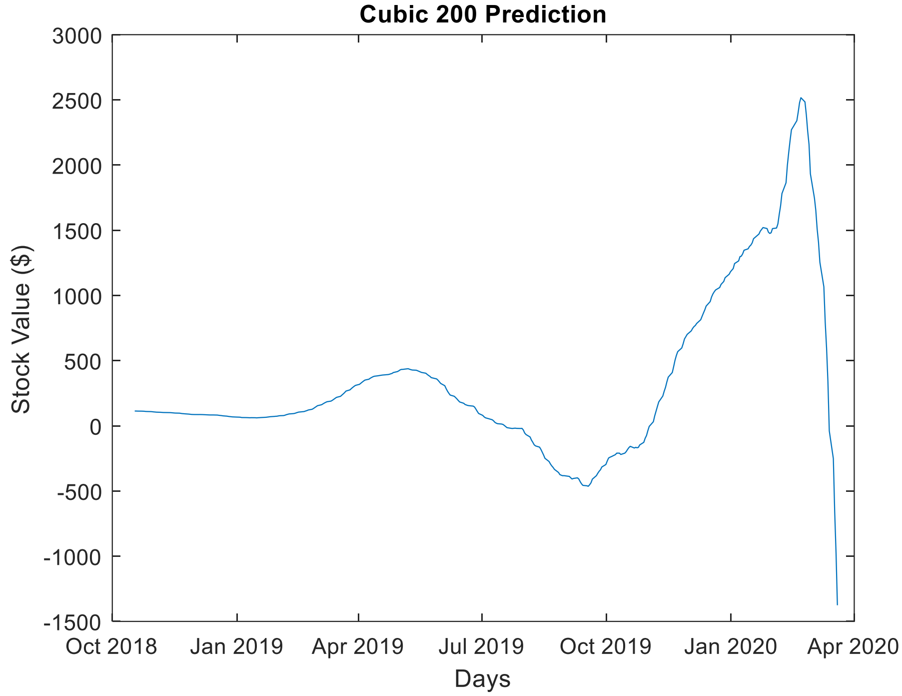
C2 = 165

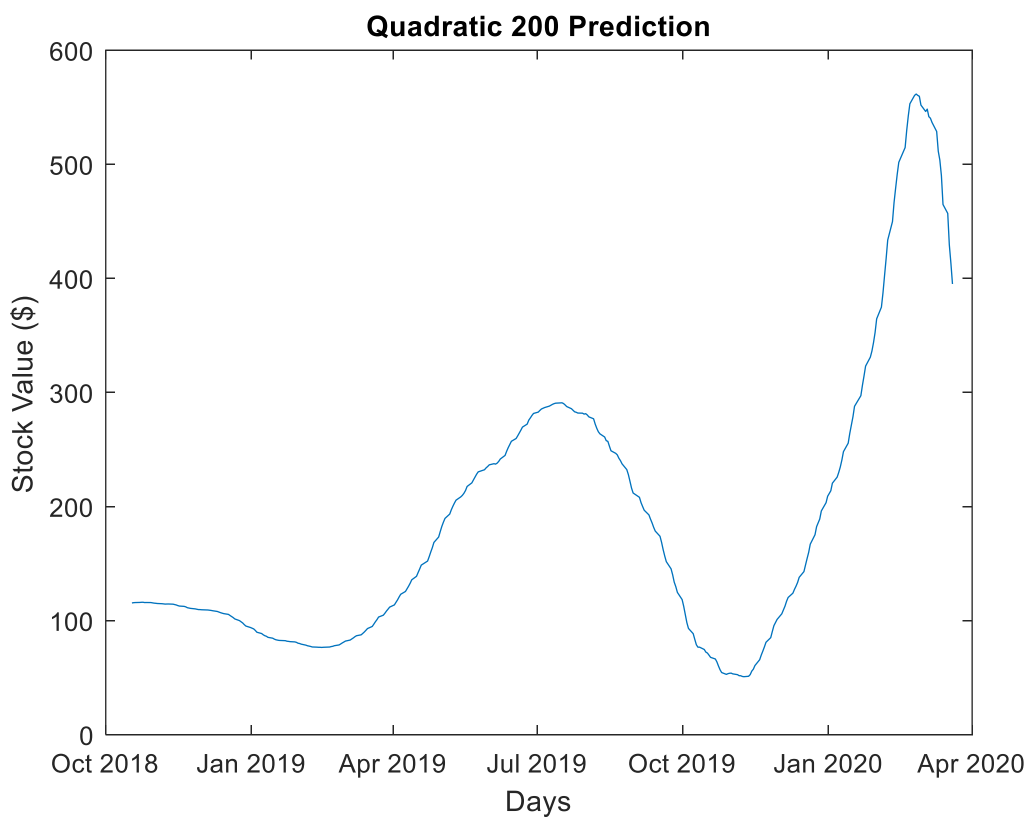
C1 /C2 = 100.9152

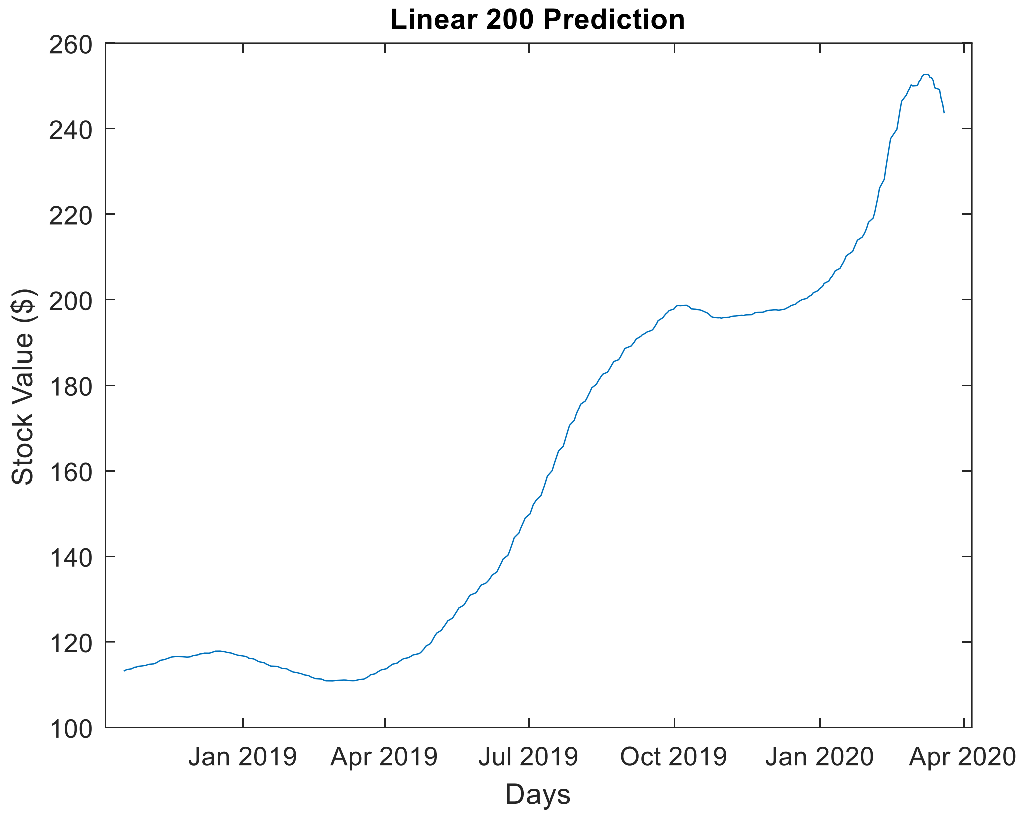
Adaptive simpson error = 3.1489 * 10^-11

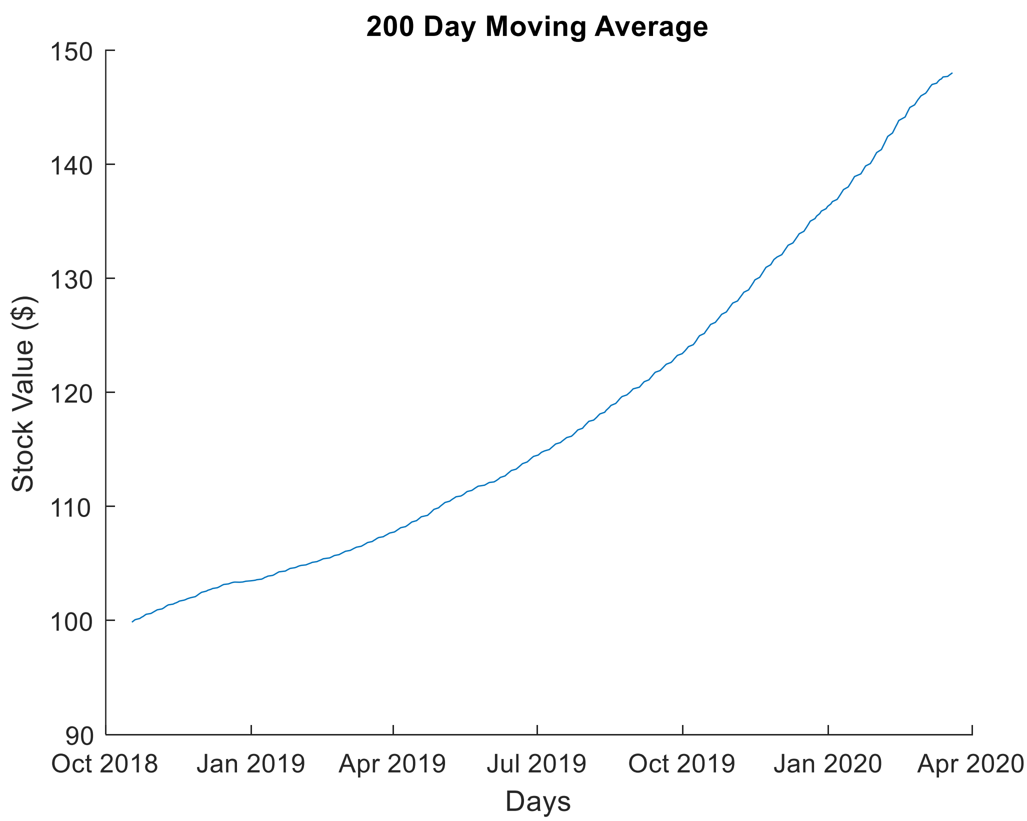
Composite simpson error = 5.5511e^-17

7.


Cubic 200 Prediction
Stock Value ($)
3000
2500
2000
1500
1000
500
0
-500
-1000
-1500
Oct 2018
Jan 2019
Apr 2019
Jul 2019
Oct 2019
Jan 2020
Apr 2020
Days


Quadratic 200 Prediction
Stock Value ($)
600
500
400
300
200
100
0
Oct 2018
Jan 2019
Apr 2019
Jul 2019
Oct 2019
Jan 2020
Apr 2020
Days


Linear 200 Prediction
Stock Value ($)
260
240
220
200
180
160
140
120
100
Jan 2019
Apr 2019
Jul 2019
Oct 2019
Jan 2020
Apr 2020
Days


200 Day Moving Average
Stock Value ($)
150
140
130
120
110
100
90
Oct 2018
Jan 2019
Apr 2019
Jul 2019
Oct 2019
Jan 2020
Apr 2020
Days