John Sipala

AMS 595 Assignment 4

https://github.com/jdsipala/AMS595Project4

For the first part of this assignment, I wrote a Python script that creates the Mandelbrot set. Using the formula z(n+1) = z(n)^2 + c where z0 = 0 and c is a complex number. I made an array of complex values between [-2,1] and [-1.5,1.5], and for each point I repeated the formula up to a specified number of times. One of the conditions in the Mandelbrot script was to check if the absolute value of z stayed below a threshold of 50, if so, it was part of the Mandelbrot set. Then, I plotted the results using Matplotlib to get the fractal image. The image and process of writing this script were familiar based on what we did on MATLAB earlier in the semester.

In the second part of the assignment, I worked with a Markov Chain that had five states. I started by making a random 5 by 5 transition matrix and normalized each row so that the probabilities in each row added up to 1. I also created a random probability vector and applied the transition rule 50 times to see how it changed. Then I used numpy eigenvalue functions to find the stationary distribution, which is the eigenvector of P^T that corresponds to eigenvalue 1. The final distribution matched closely with the result after many transitions.

For the last part of the assignment, I wrote a function to approximate functions that are infinitely differentiable using its Taylor series. I used sympy functions to calculate the derivatives and numpy to evaluate them over a range of x values. Using the function given in the assignment f(x) = x*sin2(x) + cos(x) over the interval [-10,10], I compared the true values to the approximated values. I also measured the total error and runtime for different degrees and saved the results in a csv file. As shown in Figure 2 from the assignment, the approximated values were very close to the actual values.