AMS 325 Homework #4 Python Mini-projects

Skyler Aliya 112491424

06/21/2022

Introduction:

This project contains two parts: Mandelbrot Sets and Markov Chain. Each part is required to completion of a python function.

URL of GitHub repository:

https://github.com/Allliiiya/AMS-325

Algorithm:

For part A, I first used *numpy.linspace* to creat *x* and *y* array of *c*, the complex value. Then I used a for loop form 0 to *N_max* value to iterate $f(z) = z^2 + c$. I formed 2-D array *mask* by limit $|z| < threshoold$. Lastly, given the commend in the instruction, the formed image resulted.

For part B, I first used *numpy.random.rand* to creat vector *p* and matrix *P*. Normalize the vector to the sum = 1 by divide each number by the sum. Then evaluate eigenvector and eigenvalue of the transpose of matrx *P* by *numpy.linalg.eig*. Since we want to find the p_stationary, I used *numpy.argmax* to find the max value in each row of eigenvector and normalize to sum 1 with the same algorithm. After that, I intialized a N-zero vector *norm* to store each value in the *for loop* of *p - p_stationary*. Plotting the image with matplotlib.pyplot package. In the end, I tested the function with three different *n* and *N* value.

Discussion:

Images of the Mandelbrot set exhibit an elaborate and infinitely complicated boundary that reveals progressively ever-finer recursive detail at increasing magnifications; mathematically, one would say that the boundary of the Mandelbrot set is a fractal curve.

A Markov chain or Markov process is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event. With different value of n and N in test, different x-y scatter it shows.