12-5-2018

Fractals and the Geometry of Matrix Operators

Xiaona Zhou

How does access to this work benefit you? Let us know!
Follow this and additional works at: https://academicworks.cuny.edu/ny_pubs

Recommended Citation
https://academicworks.cuny.edu/ny_pubs/333

This Poster is brought to you for free and open access by the New York City College of Technology at CUNY Academic Works. It has been accepted for inclusion in Publications and Research by an authorized administrator of CUNY Academic Works. For more information, please contact AcademicWorks@cuny.edu.
Fractals and the Geometry of Matrix Operators

Xiaona Zhou
Honors Program
Mentor: Professor Samar El Hitti, Mathematics Department

Abstract
In this project, we are interested in how certain fractals can be generated by using introductory linear algebra techniques. A fractal is an object or a structure that is self-similar in all length scales. Sierpinski Carpet and Sierpinski Triangle are two classic examples of fractals. Fractals have many applications in real life, for instance, in imaging, art, and biological sciences. In this project, we apply classes of linear transformation to describe and generate fractals in the Euclidean plane, and we generate different examples of fractal using MATLAB.

Introduction
There are twelve 2×2 matrices that perform specific transformations like reflection, rotation, compression, expansion, and shear. We introduce the effects of such matrix operators in \( \mathbb{R}^2 \) by giving a few examples:

Hausdorff dimension for self-similar objects
Hausdorff dimension (\(dH(X)\)) is concerned with different aspects of self-similar objects as opposed to other kinds of dimensions (topological dimension \(dT(X)\)).

\[ X \text{ is a unit cube. Let } K \text{ represent the number of smaller cubes inside } X \text{ of side } \frac{1}{3}. \]

\[ K = p^3 = \frac{1}{27}. \quad (\text{The exponent is Hausdorff dimension}) \]

Notice how \(dH(X) = dT(X) = 3\) (integer)

More generally:

\[ K = \frac{1}{S^p} = \left( \frac{1}{3} \right)^d \]

\[ \ln(K) = \ln \left( \frac{1}{3} \right)^d = dH(X) = \left( \frac{\ln(K)}{\ln(3)} \right) \]

Use the formula above to find the dimension of snowflake:

Scaling factor \(S = \frac{1}{3}\), numbers of self-similar snow flakes \(K = 4\)

\[dH(Sf) = \frac{\ln(K)}{\ln(3)} = \frac{\ln(4)}{\ln(3)} \approx 1.261859\]

One can think of a fractal as an object with non-integer Hausdorff dimension, but, a more accurate definition is: \(X\) is a fractal if its Hausdorff dimension is strictly greater than its topological dimension.

\[dH(X) > dT(X)\]

Snowflake is a fractal as seen from \(dT(Sf) = 1 < dH(Sf) = 1.26\)

Generate fractals in MATLAB
We used similitudes to generate examples of fractals in \(\mathbb{R}^2\).

Definition: A similitude with scale factor \(s\) is a mapping of \(\mathbb{R}^2\) into \(\mathbb{R}^2\) of the form

\[ T \left( \begin{bmatrix} x \\ y \end{bmatrix} \right) = s \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \]

where \(s, e, f\) are scalars. \(\theta\) is the degree of rotation for the transformation. Vector \(\begin{bmatrix} e \\ f \end{bmatrix}\) contains one initial point’s position and the positions of that initial point after each transformations.

The figure below is an example of fractal. We generated this Sierpinski Carpet in MATLAB.

\[ dH(X) = 1.892789 \]

In MATLAB, the similitudes would be written as:

\[ x_n = \frac{1}{3} x + 1/3; \quad \% \text{an example as } (e,f) = \left( \frac{1}{3}, \frac{1}{3} \right) \]

\[ y_n = \frac{1}{3} y; \quad \% \text{an example as } (e,f) = \left( \frac{1}{3}, \frac{1}{3} \right) \]

A part of the MATLAB code used:

\[ \text{NumOfPts} = 1000000; \quad \% \text{Numbers of points in figure} \]
\[ \text{iterations} = 5; \quad \% \text{Run the algorithm 5 times before a point is defined} \]
\[ \text{pts} = \text{zeros}(\text{NumOfPts},2); \quad \% \text{pts is a } 1000000 	imes 2 \text{ matrix with only 0s} \]
\[ \text{for } j = 1: \text{NumOfPts} \quad \% j = 1, 2, 3, \ldots 1000000 \]
\[ x = \text{rand}(1); \quad \% \text{x is a random number in } (0-1) \]
\[ y = \text{rand}(1); \quad \% \text{y is a random number in } (0-1) \]
\[ \text{for } i = 1: \text{iterations} \quad \% i = 1, 2, 3, 4, 5 \]
\[ p = \text{rand}(1); \quad \% p is a random number in } (0-1) \]
\[ \text{if } p < 1/8 \quad \% \text{One out of every eight iterations this} \]
\[ \text{y} = \text{ones}(1,5); \quad \% \text{The y values used in iteration i=2} \]
\[ \text{y} = \text{rand}(1); \quad \% \text{y is a random number in } (0-1) \]
\[ \text{end} \]
\[ \text{end} \]

By using similar techniques in MATLAB, we generated other interesting fractals.

Selected references