Multifractal Modeling of the Stock Market

Christian Duerk, Eric Mier, Max Morgan, Dustin Sieh

Introduction

Our research was in the field of fractals and, more specifically, multifractals. Fractals can be useful in modeling many real world systems often producing results more accurate than classical, Euclidean based mathematics. In this project, fractals are used to simulate a collection of stock markets. Similar to fractals, stock markets are subject to large fluctuations following seemingly unpredictable trends. Our group proposed a method of using fractal geometry in order to simulate the many rises and falls of the real world markets. In order to apply this method, our team has put together a portfolio of stocks from various well known companies using a common investment strategy known as Modern Portfolio Theory. Using least squares data fitting and data from each of the selected stocks, market trend lines are generated over a span of 60 days. A fractal generator is then repeatedly applied to these trend lines in order to construct market simulations of each selected stock. Due to their volatile nature, these simulated markets can be used to stress test a portfolio and verify if it is capable of withstanding large market and industry fluctuations such as those created during the simulations.

Background

A fractal is a geometric shape that if separated into parts, each part is a reduced scale version of the whole, which is known as self-similarity. Fractals allow mathematicians to go beyond classical mathematics, or Euclidian geometry, where everything is composed (in essence) of straight lines or edges. Using the nature of fractals allows mathematicians to investigate the patterns of the natural world, which, unlike seen in classical mathematics, are composed of rough and jagged edges and can be often characterized by self-similarity. An example found in nature is a coastline such as the one shown in Figure 1. Fractals are used in many applications such as describing the shapes of coastlines, signal and image compression, and computer graphics. Multifractals are fractals with multiple scaling rules and are often used in dealing with irregular or erratic data with geometric or probabilistic objects. An example of such irregular and erratic data can be seen in the real world stock market.

Figure 1

Using common investment strategies, even professional investors can suffer huge losses as a result of unpredicted, large fluctuations in stock or industry value. This is because many investment strategies are based upon gaining the most returns from a given level of risk, or simply, the amount of money investors are willing to risk. One investment strategy we decided to focus on is the Modern Portfolio Theory. Users of Modern Portfolio Theory carefully select proportions of varying assets that will minimize their risk at a chosen level of expected returns. Unfortunately, this method is not always effective because it does not take into account very large and unpredictable market fluctuations. As a result, investing strategies are rarely able to plan for these large fluctuations at all. Ignoring the possibility of an extreme situation could ruin a portfolio entirely if such a situation were to occur. Multifractal simulations of stocks do not ignore the unpredictable and random fluctuations of stock markets, rather they create their own based upon the underlying trend of the stock market's historical data. As a result, multifractal simulations can be used to stress test portfolios created using investment strategies such as Modern Portfolio Theory.

Matrix and Linear Algebra in Fractals

Matrix algebra is important in the creation of fractal patterns. In constructing fractals, you begin at a point in the image of the fractal and use transformations to map to a new point in the image of the fractal. Some of these transformations are linear, while others are affine. Affine transformations are a combination of linear transformations and translations. Barnsley's Fern, seen in Figure 2, is a wonderful example of a simple fractal that is created using matrix algebra. The following four transformations are used iteratively to produce the fern.

$$(1)\quad f(x,y) = \begin{bmatrix} 0.00 & 0.00 \\ 0.00 & 0.16 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

$$(2)\quad f(x,y) = \begin{bmatrix} 0.85 & 0.04 \\ -0.04 & 0.85 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 1.60 \end{bmatrix}$$

$$(3)\quad f(x,y) = \begin{bmatrix} 0.20 & -0.26 \\ 0.23 & 0.22 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 1.60 \end{bmatrix}$$

$$(4)\quad f(x,y) = \begin{bmatrix} -0.15 & 0.28 \\ 0.26 & 0.24 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 0.44 \end{bmatrix}$$

Figure 2

Transformation (1) above is a linear transformation that will map points to the stem using a scaling and a projection onto a line parallel to the y-axis. Transformation (2) is an affine transformation that will map points to create successively smaller leaflets. Transformation (3) is also an affine transformation, it will make the largest left-hand leaflet. Transformation (4) is another affine transformation that will create the largest right-hand leaflet.

Fractal Generator and Simulation

To begin the computation, a trend line was created from historical market data using the least squares method. To do this, a function of the form $f(t) = c_0 + c_1 t$ must be found

$$\text{such that}\quad \begin{vmatrix} f(x_1) & = & y_1 \\ f(x_2) & = & y_2 \\ f(x_3) & = & y_3 \\ f(x_4) & = & y_4 \end{vmatrix} \qquad \text{or} \qquad \begin{vmatrix} c_0 & + & x_1c_1 & = & y_1 \\ c_0 & + & x_2c_1 & = & y_2 \\ c_0 & + & x_3c_1 & = & y_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_0 & + & x_nc_1 & = & y_n \end{vmatrix}$$

where $x_1, x_2, x_3, \ldots, x_n$ and $y_1, y_2, y_3, \ldots, y_n$ are the corresponding x and y values of the data points taken from the historical market data. To find the constants represented by c_0 and c_1 we used MATLAB to calculate $\overrightarrow{C^*} = (A^TA)^{-1}A^T\vec{b}$ where

$$A = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \qquad \text{and} \qquad \vec{b} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

After finding constants c_0 and c_1, they were applied to the equation $f(t) = c_0 + c_1 t$ in order to construct the trend line of each stock used.

Next, the trend line is put through a fractal generator which splits the trend line into three segments of equal length. The first segment will be put through an affine transformation composed of a rotation and translation while the third segment will be put through a linear transformation composed of only a rotation. The angle of rotation for both segment one and segment three is determined using a pseudorandom number generator that generates angles between an upper and lower bound. The upper and lower bound for the angle of rotation is dependent upon the original slope of the line and can be calculated for a positive slope as shown below where theta is the angle of the original line with respect to the horizontal.

$$\Phi_{lower} = 0^\circ$$
$$\Phi_{upper} = 90^\circ - \theta$$

Similarly, for a negative slope the upper bound is found by subtracting theta from -90°. These upper bounds are chosen in order to avoid the segments from being rotated past the vertical. The translation of the first segment along the x-axis is determined using a pseudorandom number generator that generates values between 0 and one tenth the length of the line segment. The first segment is rotated about its leftmost x-y coordinate and shifted left along the x-axis as seen in Figure 3 and described by the system of equations below where phi is the angle of rotation and epsilon is the shift applied in the x-direction.

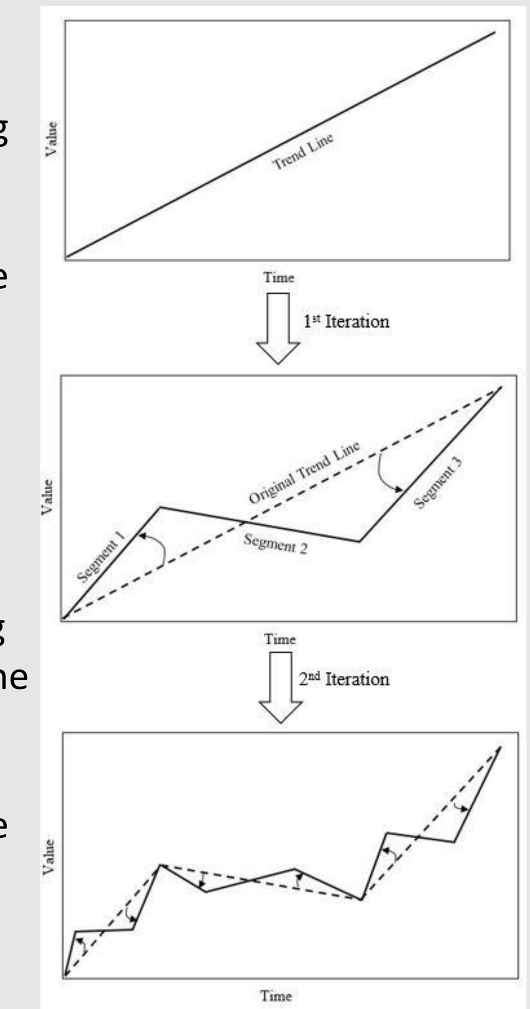

Figure 3

$$\begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \varepsilon \\ 0 \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

The third segment is rotated about its rightmost x-y coordinate by the same angle as the first. This transformation can be seen in Figure 3 and is described by the system of equations below.

$$\begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

The second segment is created by connecting the first and third.

By performing the rotation and translation on the first segment, the value of the stock either increases or decreases and occurs in a shorter time period, thus increasing the volatility of the stock simulation. This process described above is performed on each individual line segment within the simulation for a number of iterations. The number of iterations can be specified by the user. An example of the first two iterations can be seen in Figure 3.

In our experiment, a portfolio was constructed containing ten industry stocks and five mutual funds. The simulation was then performed on each of the stocks in the portfolio. These simulations are not intended to accurately predict behavior of the stock they represent but rather to display similar activity, such as the large, unpredictable fluctuation of value often found in real world stocks.

Simulation Results and Analysis

One stock used in the experimental portfolio was Kellogg Company (K). Stock data was obtained ranging from January 4th 2010 to March 5th 2010. The original stock data can be seen in Figure 4. The simulation was performed using this raw data and yielded results shown in Figure 5.

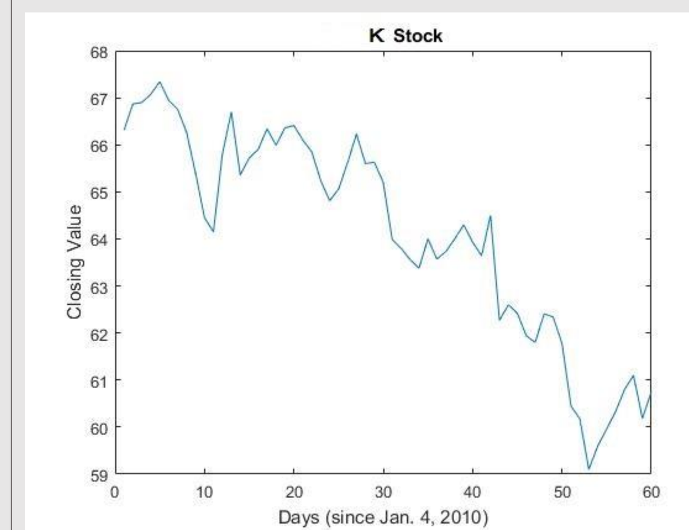

Figure 4

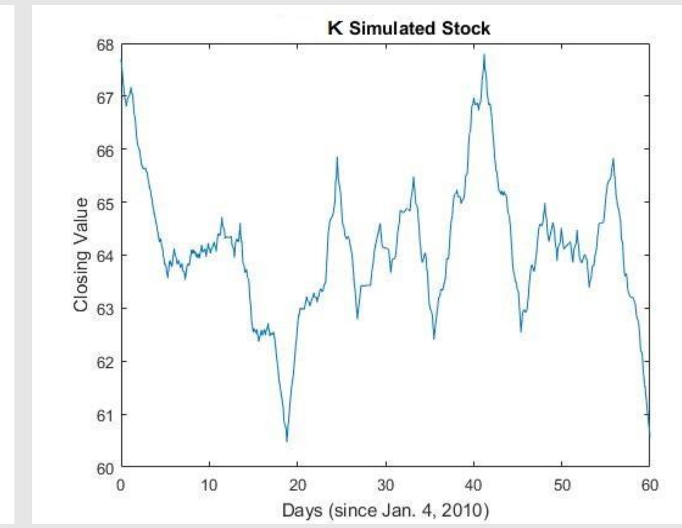

Figure 5

The results obtained from the simulation on the Kellogg Company stock data represent our experiment the most effectively. The simulation yielded magnitudes of fluctuations that are quite comparable to those seen in the real stock data. The range of the actual stock data is about $8.50 per share and the range of the simulated stock data is about $7.00 per share. This simulation is a good example of how our fractal generator works in general. It is not necessarily a good predictor of the specific behavior of the actual stock, however it does model the potential fluctuations in price that actually occurred for this stock, which is the overall purpose of our simulation.

Extension of Multifractal Modeling

To summarize, the fractal generator simulation can be used to test the "safety" of a given portfolio. For a safe portfolio, it should spread its capital over a large and diverse span of industries and investment opportunities. That way, no large losses in a certain industry or investment will ruin the portfolio as a whole. For example, the portfolio should be able to balance out and handle the large, severe dips of any one stock with the general rises of all the other ones. While the simulation tests a portfolio, it should not be used as a stand alone tool to create a portfolio. The simulation should instead be used as an additional tool to indicate the effectiveness of the investment strategy, such as Modern Portfolio Theory, that was used to initially construct the portfolio.

As an overall test, one can run the simulation on all of the stocks within the portfolio and plot all of the resulting fractals on one plot to see the overall gain/loss of the portfolio at each day in the generation. Using this plot, the "safety" of the portfolio can then be analyzed. Looking at the plot, if there are regions with severe overall losses within the portfolio, one can conclude that the investments made could be manipulated to yield better results. This is a way that the simulation can be used to exploit the flaws in a certain portfolio.

It is important to understand that the simulation created from the fractal generator has the possibility to be extremely volatile and far more random than what the Modern Portfolio Theory accounts for. This increased level of volatility is the aspect that makes the fractal simulations a stress test. This is beneficial because it prepares the portfolio for a potentially more chaotic and random market than what the Modern Portfolio Theory would predict. Using the generator, one can conclude that if the portfolio performs well under these circumstances, they can be confident that the portfolio will perform well in the real world market.

The use of multifractals to model the stock market is still very much a new field of research, only arising within the last decade. Based on this fact and our research, we believe this to be a promising way to help people build their portfolios in the future. Out simulation in particular is only being used on a handful of stocks and with limited investment knowledge and experience. With more research into this topic, we are sure to discover even more efficient ways to model stocks and using linear algebra concepts that we have learned in this class, as well as other advanced mathematical topics we learned over the course of the project.