Goodness of fit test for the multifractal model of asset returns

Bell, Peter

University of Victoria

23 April 2012

Goodness of fit test for the Multifractal Model of Asset Returns

Peter Bell[1]

University of Victoria Economics Term Paper[2]

Abstract. This paper explores extensions to the random walk model for time series in finance. There is some disagreement about the suitability of multifractal probability models, but they have compelling attributes. Research that has found no evidence to support the multifractal model has used testing procedures that do not have known statistical power. Therefore, there is an opportunity for new methodology. This paper presents a testing procedure to determine if data follows a multifractal or monofractal process. Using simulation, the paper derives the power of the test. Although the power is low, the test suggests that some observed prices do follow multifractal behaviour. This is a strong result. Further, this work suggests there will be further disagreement in the literature going forward due to the difficulty of identifying multifractal data.

[1] - PhD student, pnbell@uvic.ca.

[2] -Econ 546 Econometrics II, April 2012, Dr David Giles.

1. Introduction:

This paper develops a formal criteria to choose between two models for time series in finance. Both models are extensions of the random walk model, where percentage changes follow a known distribution. One model is the monofractal, where percentage changes follow a normal distribution. The other model is a multifractal, where percentage changes are governed by a distribution that is determined by a combination of fractal distributions. To determine if data has evidence of multifractal behaviour, a researcher must determine if a particular curve is linear or nonlinear. Although this seems simple enough, it has not been studied before using formal statistical procedures. This paper explores the decision using a goodness of fit test.

The paper works towards several goals. First, I explore relevant literature that reveals disagreement about the use of multifractals in finance. By using simulation, this paper adds structure to the debate around testing for presence of multifractals. Next, the paper shows how the multifractal model can be calibrated to observed price data. Finally, theory tells us that a particular curve will be linear for a monofractal and nonlinear for a multifractal; this motivates a test statistic based on the second derivative of the curve. Lacking theoretical guidance on properties of the test statistic, the paper simulates monofractal data to identify a critical value for the test statistic. By simulating multifractals and using the critical value, it is possible to construct the power of the test. The results show that it is very difficult to correctly distinguish between mono and multifractals using the test statistic. This suggests that literature on the topic will continue to be unsettled due to the difficulty of this question at the core of the model.

2. Literature Review

The Multifractal Model of Asset Returns (MMAR) is an important model that was proposed in the seminal paper by Mandelbrot, Calvet and Fisher (2001). The model builds on fractal geometry and has compelling strengths. The interested reader should start by reviewing Mandelbrot, Calvet, Fisher (2001) and subsequently review articles that use the MMAR for empirical purposes, such as Matteo (2010). Together, these empirical and theoretical papers give a clear picture of the model and how to use it; alone, however, they can be daunting.

In contrast to papers that focus on empirical use of the MMAR, several papers focus on theoretical properties of the model. Calvet and Fisher (2004) provide a comprehensive review of the model and, in passing, provide a test to distinguish between mono and multifractals. The test is motivated by simulating monofractals to calculate the standard deviation of points on a particular curve; the test is calculated by counting the number of times that an empirically-based curve is different outside the error bounds of the monofractal one. Since the test is not the focus of the paper, the authors do not explore the power of the test. However, I suspect that their test does not focus on the linear-nonlinear tradeoff, which motivates my use of the second derivative.

Testing for the presence of multifractals is either a passing concern or the central concern of a researcher. Amongst those who are concerned with testing for the presence of multifractals, two separate papers deduce that there is no evidence for the MMAR in observed price data. Focusing on high frequency, intraday data, Jianga and Zhoua (2008) find evidence that observed data does not have multifractal attributes. This result is found using a box counting method which is different from the estimation procedure I will use. At this time, it is not clear to me why the authors find no evidence for MMAR, however, this negative result is troubling for the MMAR because it is supposed to be broadly applicable.

Another researcher finds that "the apparent nonlinear scaling… can be accounted for by spurious multi-scaling typically obtained with financial data after randomization of their temporal structure" (Lux 2003, 7). Using price data for stock market indices, a currency pair (USD/DM) and daily gold prices, the author finds no evidence that the data follows a multifractal process; although he notes that this result may be driven by low power of the statistical test. Lux uses a reshuffling approach, where the data is reordered and used to estimate the distribution of key properties. The author finds that the nonlinearity of c(q) is not large relative to estimates of c(q) obtained from reshuffled data. Although the author states that this is a typical technique, there is no discussion of the power this test has for distinguishing multifractals. This criticism (unknown power of test) applies to both research efforts that found no evidence of multifractal data using observed data. In fact, these shortcomings motivated my use of simulations. Simulations allow a researcher to simultaneously control and test for the presence of multifractal data, which provides new insight into the behaviour of mono and multifractal in the MMAR estimation framework.

The discord within the literature is compelling for my work. Although Calvet and Fisher (2004) simulate monofractals to identify bounds on the curve of interest, it is not clear that any researcher has simulated different types of data to study whether a particular testing procedure is any good at distinguishing between mono and multifractal data. This means there is a great opportunity to strengths and weaknesses of various procedures that are used to test for the presence of multifractals.

3. Analysis

In the first section, I show the estimation procedure for the MMAR. I use observed data for GLD, the most heavily traded gold ETF in the USA, from February 13 2012. This asset is chosen because it is highly traded but not covered in other papers. The second section explores simulation of the MMAR and presents the test statistic. The simulations use software created by Christian Wengert (2010). The third section applies the test to the observed GLD data and presents a power curve that shows the ability of the test to correctly reject the monofractal model.

1. Estimation Techniques

This section discusses some of the methods used to estimate the MMAR. The procedure starts with observations on price taken at regularly spaced intervals in time. This basic data set should have the highest frequency of interest, for example, the GLD data I use is updated every 10 seconds. This data will be denoted as $S(i)$, the i-th observation of prices.

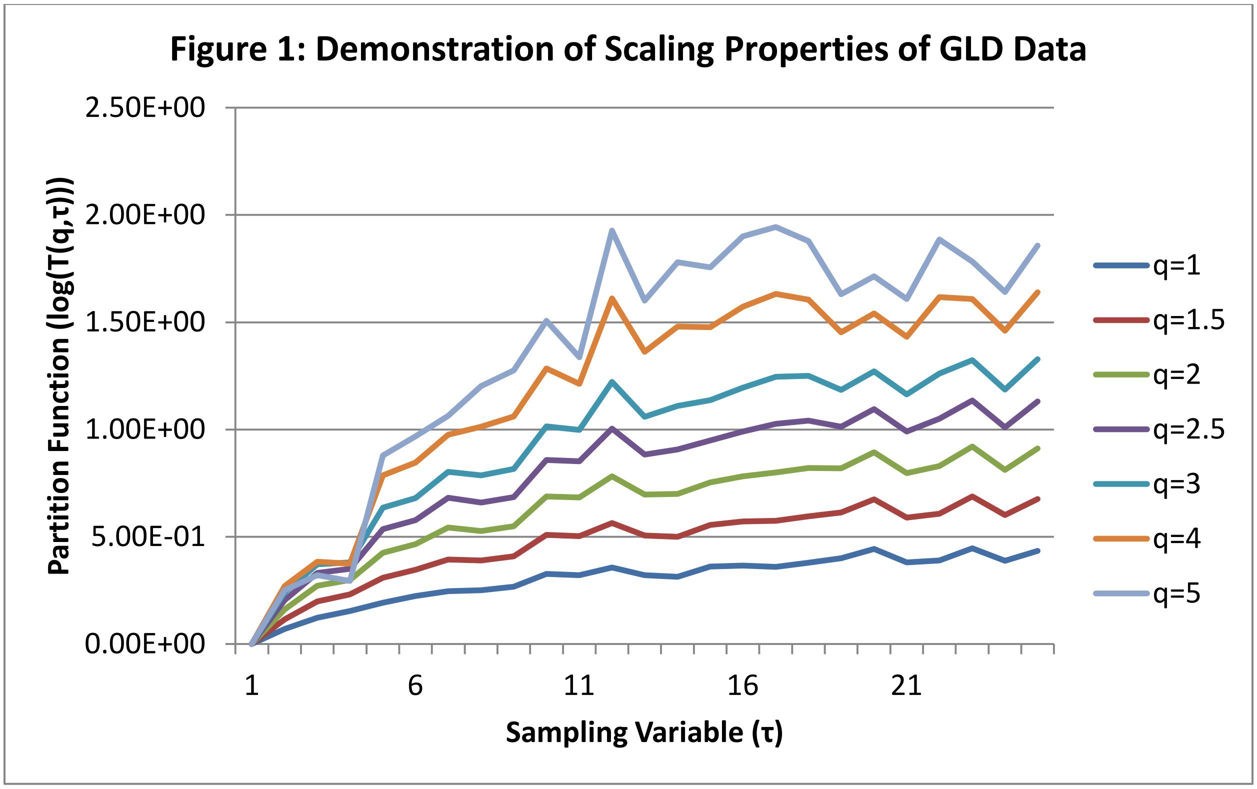
The first variable we need to define is τ (tau), an integer larger than 1 and smaller than N (number of price observations). The τ represents the rate at which the basic data set is resampled to form a new sample. If $\tau = 100$ then we form a new sample of price observations that has a new entry every time 100 observations are made in the basic data set (every time 1,000 seconds pass). At first, τ will be fixed but subsequently will be changed in a loop. I define $X(j)$ as the j-th observation of prices under the new sampling rate τ. For fixed τ, the $X(j)$ are calculated as:

(1) $X(1) = S(1), X(2) = S(1+\tau), X(3) = S(1+2\tau), \ldots X(k) = S(1+(k-1)\tau).$

The next object of interest depends on a variable q, which is a positive real number. For q in a certain range, multifractal data is known to follow a power law scaling rule; generally, we consider q in $(0,5)$ (Calvet and Fisher, 2004). The central object in this scaling rule is the q-power of absolute percentage changes, which I denote as $T(q,\tau)$. This term can be calculated as follows:

(2) $$T(q,\tau) = \frac{1}{M}\sum_{i=0}^{M} \left|\frac{\mathrm{X(i+1)}}{\mathrm{X(i)}} - 1\right|^{\mathrm{q}}.$$

The object $T(q,\tau)$ is sufficiently important that it is represented graphically (Calvet and Fisher 2004, Matteo 2010). The object is related to the average percentage change of the prices., but does not have a physical interpretation because of the exponent q. When it is calculated for the GLD data, I find the following result:



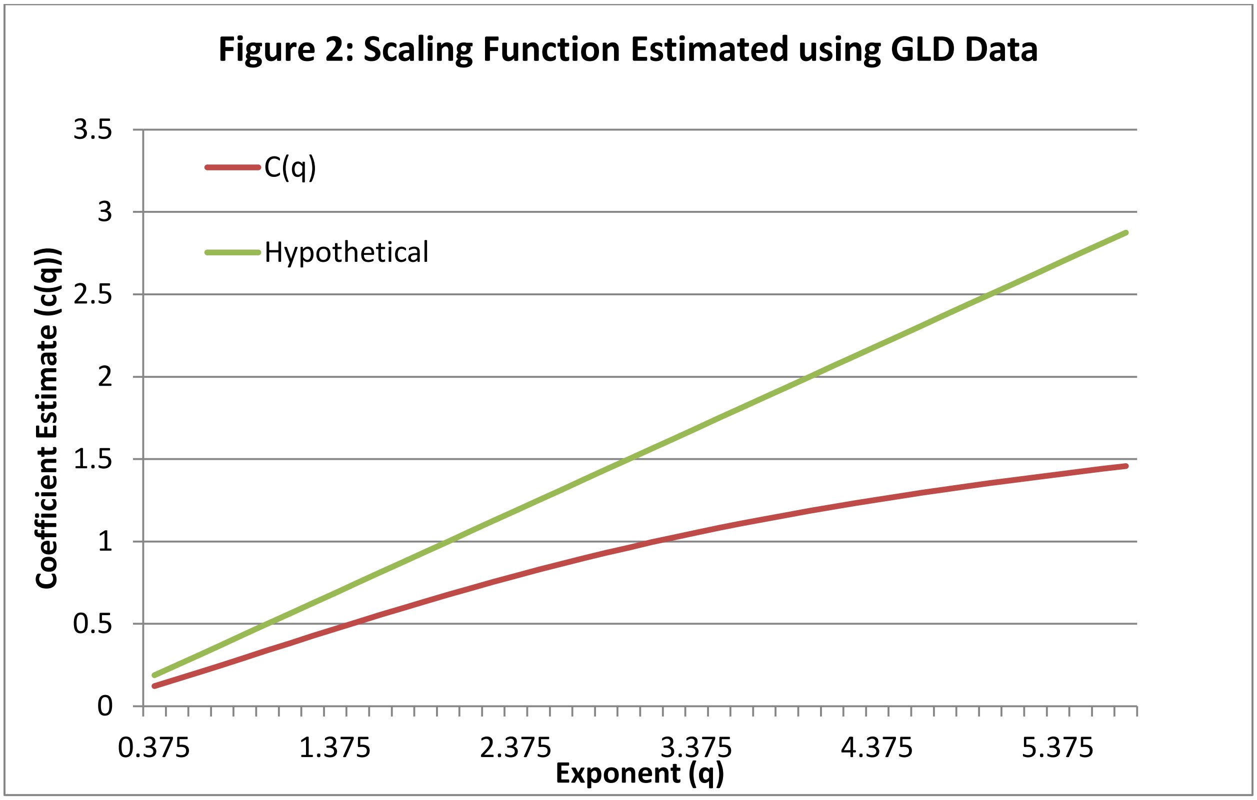
The behaviour shown in Figure 1 is typical of $T(q,\tau)$, as reported in Calvet and Fisher (2004) and Matteo (2010). There is good reason that the axes are as shown in Figure 1; the variable τ serves a mechanical purpose in calculating the scaling behaviour of the data. In contrast, q has a higher order interpretation and there is a good reason why there is a separate curve for each value of q. The physical interpretation of points in Figure 1 is secondary to the relationship between the slopes of the lines, which are the next step towards revealing the scaling rule present in fractal data.

At this point, the estimation procedure is not complete. Next, the researcher holds q fixed and performs linear regression as follows:

(3) $$\log\big(T(q,\tau)\big) = b + c\log(\tau) + \varepsilon.$$

The key variable of interest in Equation (3) is the coefficient estimate c. Since this regression depends on q, this will be referred to as $c(q)$. In other research, this is referred to as ς - the multiscaling exponent (Calvet and Fisher, 2004). For simplicity, I refer to it as $c(q)$.

To proceed, the procedure requires that we graph $c(q)$ vs q. If the data is monofractal then, in theory, the curve will be linear with slope 0.5. If the data is multifractal then the curve will be nonlinear. This graph is the central concern for the rest of this paper.



The results presented in Figure 2 are typical of research using empirical data. Researchers use rules of thumb (at best) to determine whether a given curve is sufficiently nonlinear that they can infer the presence of multifractal data. It is clear that $c(q)$ estimated using GLD data is nonlinear, but it is not clear if it is sufficiently nonlinear to say it is multifractal data; this requires that we know something about the variability in the estimates of a monofractal. This unsatisfying state of affairs motivates the need for simulations.

2. Simulation Procedure

To begin, I simulate monofractal price data and calculate the $c(q)$ curve. Then, I simulate 1,000 monofractal paths and calculate the variability of the $c(q)$ curve. This is important to confirm that my results are comparable to those reported in Calvet and Fisher; I find that standard deviation of $c(q)$ increases with q and achieve similar levels as Calvet and Fisher (2004). The results are:

Table 1: Standard Deviation of c(q) from Monofractal Simulation

q	0.5	1	2	3	4	5
s.d. c(q)	0.024	0.046	0.089	0.137	0.190	0.248

Although it is possible to use Table 1 to generate a test to distinguish mono and multifractals, I suspect this would not test for linearity of $c(q)$ precisely; rather, it would test if the $c(q)$ curve is different from the line with slope 0.5. Recall that we are concerned with testing if $c(q)$ is linear or not, because a monofractal gives a linear $c(q)$ curve with slope 0.5. Thus, it is important to test whether $c(q)$ has different curvature than a line with slope 0.5 . If the curve is a monofractal, then the second derivative will equal to zero because the slope is constant.

The curvature of $c(q)$ can be calculated by applying a numerical derivative formula twice. I calculate the second derivative $c''(q)$ for the observed GLD data and show the results as the thick line in Figure 3. For comparison, I show $c''(q)$ as estimated using several paths of monofractal data. The results are shown in Figure 3.

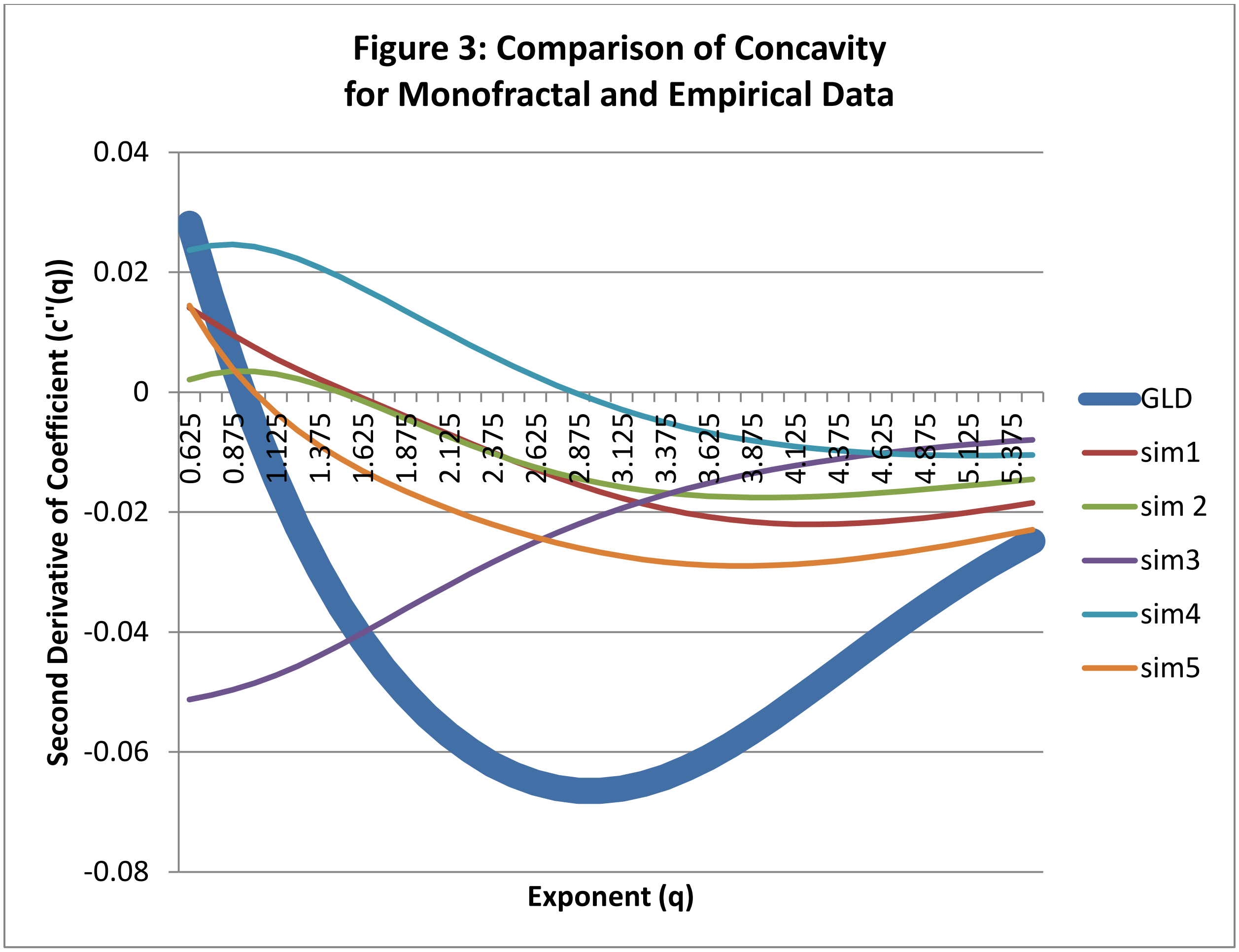
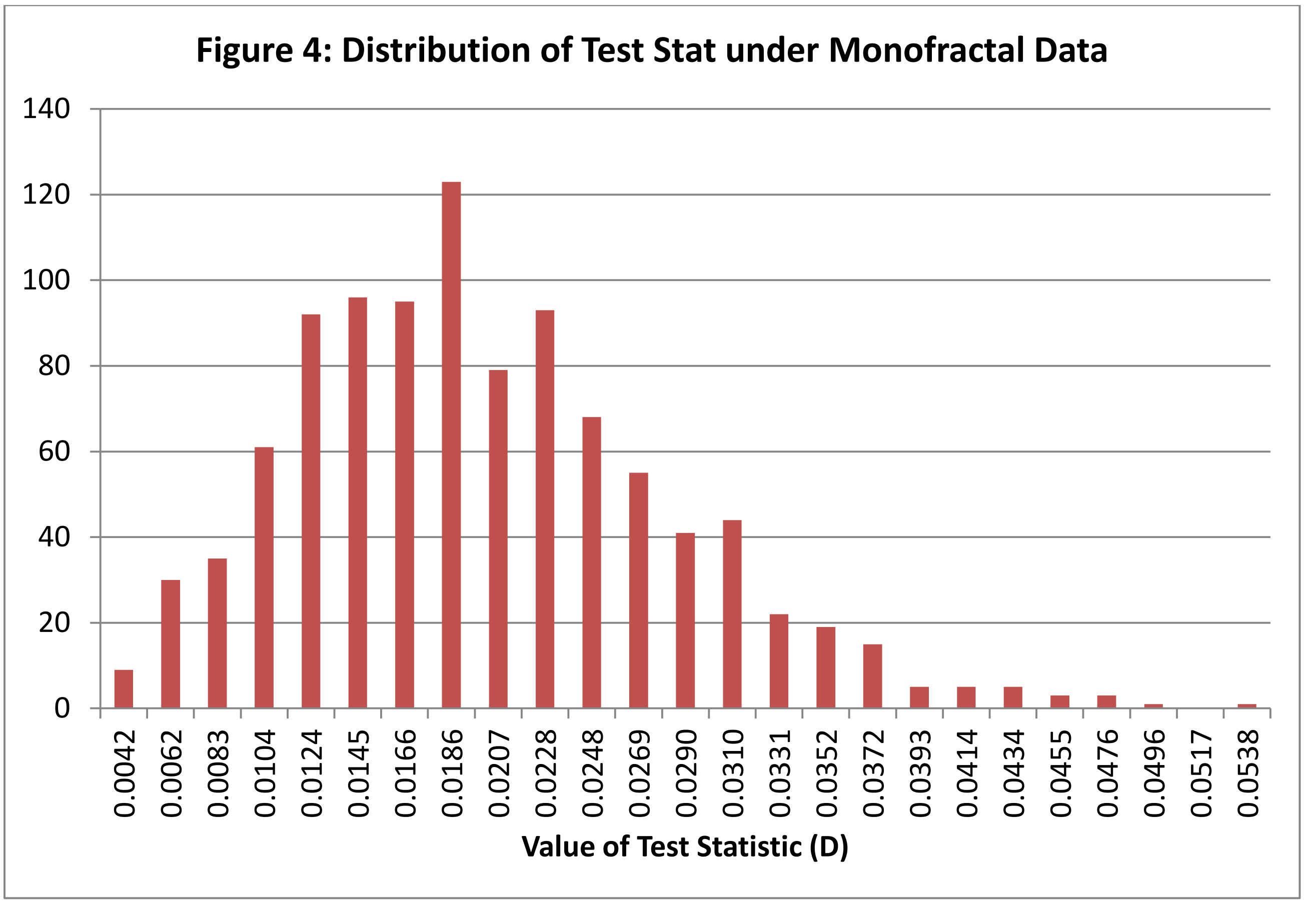


Figure 3 shows how $c''(q)$ compares for empirical data and several monofractal simulations. There are strong differences between most monofractal curves and the empirical one. This is a promising result that leads nicely into a testing procedure. A goodness of fit test compares an estimated curve to a benchmark to determine if the estimated one is significantly different. Here, the estimated curve will be $c''(q)$ and the benchmark will be zero. A goodness of fit test requires a metric that measures the difference between the curves; I use the absolute value of the difference. Based on this, the test statistic D can be defined as:

(4) $D = \frac{1}{L}\sum_{\mathrm{q}}|c''(q)\,|$ where q takes discrete values in $(q_1, q_2, \dots, q_L)$.

This test statistic is the average value of the second derivative of $c(q)$. I use an average so that other researchers can use the critical values for D, even if they use different numbers of q points. In theory, the value for this test statistic should be zero for a monofractal. However, this may not be true for a given sample. To investigate this, I simulate many monofractal paths and calculate the distribution of the test statistic. For 1,000 simulations, I estimate the distribution of D as follows:



Using Figure 4, it is possible to get a critical value for the test statistic D. This value will be used to test if observed data is monofractal or multifractal, and to build a power curve for the test. At 95% level of significance, the critical value $D_{0.95} = 0.0347$. This value is the fiftieth most large observation of all simulated D values. At 99% significance, the critical value is $D_{0.99} = 0.0406$. These critical values are designed in a way that allows any researchers to use them, regardless of the values number of q-points that they consider. For reference, I use $L = 40$.

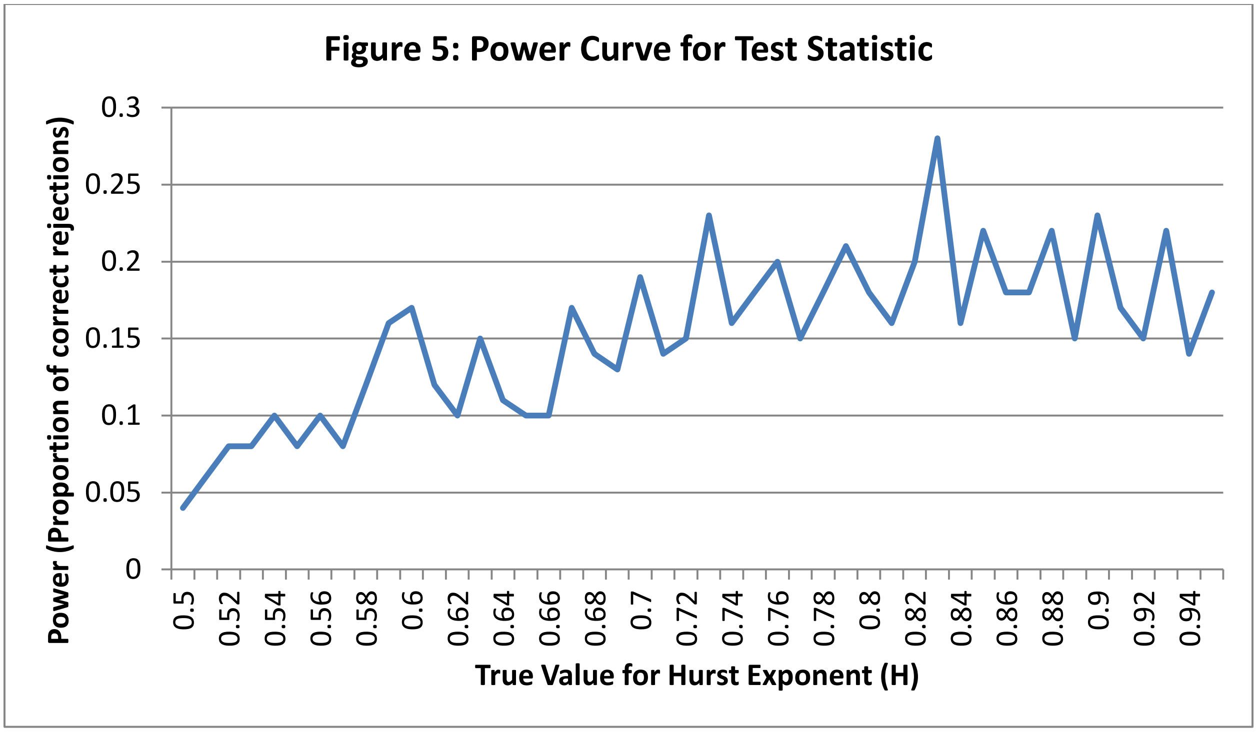
3. Results

First, I state the statistical procedure in terms of the hypothesis, test statistic, and decision criteria. This testing procedure is supposed to be valid for testing the MMAR in any situation.

Table 2: Goodness of fit test procedure for MMAR		
Hypotheses:	**Test Statistic:**	**Decision criteria:**
H_0: Data is generated by monofractal model (H=0.5)	$D = \frac{1}{L}\sum_{\mathrm{q}} \lvert c''(q) \rvert$ where $c(q)$ is calculated as in Section 3.1.	Reject H_0 if $D > D_{0.95} = 0.0347$
H_A: Data is generated by multifractal model (H>0.5)		

When the test statistic is applied to the data for GLD, the test statistic $D = 0.0448$. Since this is larger than $D_{0.95}$, the test suggests that the GLD data is not from a monofractal at 95% confidence. The test suggests that GLD has multifractal properties. It is encouraging that this result confirms the intuition that Figure 2 suggests GLD has presence of multifractal data.

A more involved result is the power curve for the test. This result depends specifically on the structure of the test statistic D and the critical value used. Power is the probability that the test reports that data is not monofractal when the data is, in fact, not monofractal. Power is a good thing for a statistical test since it means the test is giving a correct result. It is possible to calculate power since we can simulate multifractal data (Wengert 2010). To simulate a multifractal path the key parameter is H, the Hurst exponent. If $0.5 < H < 1$ then the data will be generated by a multifractal process. So, I establish a grid of values for H in $(0.5,1)$. For each H value, I simulate 100 paths. For each path, I calculate the test statistic D; if $D > D_{0.95}$ then this represents a correct rejection. For each value of H, the power is the proportion of correct rejections to incorrect rejections.



The power curve implies that it is difficult for this test to distinguish between mono and multifractals. For all possible parameters, the power is small and this means the test will often give false signals. In particular, the test will often suggest that data is monofractal when it is actually multifractal. This has two implications.

Figure 5 implies that if we reject the null using observed data then this is a very strong result. This is because the test proposed here has low power; even when faced with multifractal data, the test is likely to suggest that it follows a monofractal distribution. So, if the test suggests data follows a multifractal process then we can infer that the data is strongly different from a monofractal. This is adds weight to our result that GLD data is multifractal.

Figure 5 also leads me to expect that the literature on the topic will continue to be divided on suitability of the MMAR for financial data. The test shows that it is very difficult to distinguish between mono and multifractal processes; therefore, research showing that data follows monofractal process is to be expected. This disagreement may be based on diagnostic challenges from using the model, not the model itself.

3. Conclusion

This paper replicates results that show the MMAR in action. The analysis suggests that high frequency gold price data is multifractal, but the literature does not provide a tool to formally test whether this is true or not. To show that this methodological gap poses a problem, the paper uses simulation to show that mono and multifractal behave in subtly different ways in the MMAR estimation framework. Next, the paper proposes a goodness of fit test to determine if the curve of interest is linear with slope 0.5. The paper defines a test statistic and determines critical values for the statistic based on simulation of monofractals. When applied to gold price data, this test suggests that there is evidence for a multifractal model. However, the paper shows that power of this test is quite low. This makes the result that gold prices have multifractal attributes is more strong because it is a hard result to achieve.

There is much future work to do with the MMAR. The model has attributes that make it compelling for use in industrial applications and theoretical models of financial markets. However, the results presented here are not a final word on testing between mono and multifractals. In fact, there is an outstanding question around the simulation of multifractals that could motivate further research. When simulating multifractal data, the program by Christian Wengert (2010) allows a researcher to vary H (the Hurst exponent) and b (the number of subdivisions). When I simulate multifractal paths, I vary H but keep $b = 2$ fixed; it may be that larger b causes multifractal paths to become more pronounced. It follows that this change could improve the power of the test. The test statistic proposed in this paper would become more valuable if it could be shown that the power increases when the data comes from a better multifractal simulation.

References:

Calvet, Laurent and Adlai Fisher. 2002. "Multifractality in asset returns: theory and evidence." *The Review of Economics and Statistics* 84:381-406.

Di Matteo, T. 2007. "Multi-scaling in finance." *Quantitative Finance* 7:21-36.

Jianga, Zhi-Qiang and Wei-Xing Zhoua. 2008. "Multifractality in stock indexes: Fact or Fiction?" *Physica A* 387:3605–3614

Lux, Thomas. 2003. Detecting multi-fractal properties in asset returns: the failure of the scaling estimator. Economics working paper, Department of Economics, Christian-Albrechts-Universität Kiel.

Wengert, Christian. "Multifractal Model of Asset Returns (MMAR)." Last modified December 12, 2010. http://www.mathworks.com/matlabcentral/fileexchange/29686-multifractal-model-of-asset-returns-mmar

Appendix

"Goodness of fit test for the Multifractal Model of Asset Returns"[2]

By Peter Bell[1]

Note:

This appendix contains all the code I wrote to execute the estimation and simulation reported in the paper. The code is written in Matlab. There are two files: first, a long file that makes calls to various files to conduct the research; second, a function that is used to estimate the c(q) function from the paper.

To use the code provided below, you will require to download three files written by Christian Wengert (2010). A link is provided below[3]. These files allow you to simulate multifractal processes.

[1] - PhD student, pnbell@uvic.ca.
[2] -Term Paper, Econ 546 Econometrics II, April 2012, Dr David Giles.
[3] - http://www.mathworks.com/matlabcentral/fileexchange/29686-multifractal-model-of-asset-returns-mmar

```
%%  Code Supplement for paper "Goodness of fit test for MMAR"
%   By Peter Bell, April 2012.  For Econ 546 Econometrics 2, Dr David Giles
%
%   This file calls functions in order to estimate MMAR for GLD price data.
%   Also conducts simulation to establish power of test proposed in paper.
%
%
%
% See the following paper
% A Multifractal Model of Asset Returns by B Mandelbrot - 1997
%
%%  Empirical example to show estimation technique with price data:
%   Notation: qHq refers to c(q) from the paper.
%
clear all
load('GLD Data.csv')    %   Price data, 10 second frequency, Feb 13 2012
price = GLD_Data(:,10); %   set price equal to ask
empiric_qHq= qHqFunction(price)
%   Result: c(q) curve for empirical data
save('empiric_qHq.mat','empiric_qHq')   %   this is c(q) for GLD data

%   Next, calculate T(q,tau) curve for empirical data.
n = length(price);
offsetd = zeros(1,1);
counter = 1;    %   Used to calculate X(q,a)
step = 2;   %   Track current value of offset distance dt
top = 50;    %   Upper bound for dt (tau) in loop
%   Assemble data into tau-offset format
for j = 2:top
    step = j;
    counter = 1;
    while counter*step+1<n,
        pctchange = log(price(1+counter*step)/price(1+step*(counter-1)));
        offsetd(counter,j-1) = pctchange;
        counter = counter + 1;
    end
end
%   Create matrix to store values of dt, each row is a value of dt
for i = 1:top-1
    regstep(i) = i;
end
%   Calculate the Q-power of price changes, for each level of dt
qvec = [1,1.5,2,2.5,3,4,5];
for l = 1:length(qvec)  %loop over Q
    for m = 1:top-1     %loop over step size dt
       temps = sum(offsetd(:,m) ~=0,1); %sample mean for nonzero entries
       moment = qvec(l);
       regdata(m,l) = sum(abs(offsetd(:,m)).^(moment) / temps); %crucial
       Tcurve(m,1) = regstep(m);
       Tcurve(m,l+1) = regdata(m,l)
    end
end
%   Result: calculate T(q,tau) curve.
save('empiric_Tcurve.mat','Tcurve')   %   this is c(q) for GLD data

%   Next, calculate c''(q) curve for empirical data
```

```
Hprime_empiric = [0,0];
Hmat_empiric = empiric_qHq;
h = Hmat_empiric (2,1)-Hmat_empiric (1,1);
%   Calculate numerical derivative
for i = 2:length(Hmat_empiric (:,1))-1
    Hprime_empiric(i-1,1) = Hmat_empiric(i,1);
    Hprime_empiric(i-1,2) = (1/(2*h))*(Hmat_empiric(i+1,2)...
        - Hmat_empiric(i-1,2));
end

Hdprime_empiric = [0,0];
for i = 2:length(Hprime_empiric(:,1))-1
    Hdprime_empiric(i-1,1) = Hprime_empiric(i,1);
    Hdprime_empiric(i-1,2) = (1/(2*h))*(Hprime_empiric(i+1,2)...
        - Hprime_empiric(i-1,2));
end

save('Hdprime_empiric.mat','Hdprime_empiric')
%   this is c''(q) for GLD data

 %%  Simulation of monfractal to calculate variability in c(q) q-Hq curve
clear all
%   Key Parameters
H = .5;      % Hurst Exponent
ln_mu = 0.05;  % Drift from GBM equation
ln_sigma = 0.1;     %   Vol from GBM equation

%   Secondary Parameters
numsim = 1000; %    Number of paths to be simulated
toptau = 50;   %    Largest offset used when calculating percentage changes
b = 2;  %   Number of subdivisions
k = 10;     %   Length of interval

%   Calls to create data
temp0 = mmar(b, k, H, ln_mu, ln_sigma);
temp1 = diff(temp0);     %  Lose one observation here
for i = 1: length(temp0)-1
    temp2 = cumsum(temp1(1:i));
    tempdata(i) = exp( temp2(i)/100);   %   Multiplicative process
end
tempans = qHqFunction(tempdata);
numpoints = length(tempans);
tempresample = tempans;     %   First c(q) curve under simulation
%   First c(q) curve assigns observation points {q} for results matrix

for i = 2:numsim
    %   Create geometric data from additive
    temp0 = mmar(b, k, H, ln_mu, ln_sigma);
    temp1 = diff(temp0);     %lose one observation here

    for j = 1: length(temp0)-1
        temp2 = cumsum(temp1(1:j));
        tempdata(j) = exp( temp2(j)/100);
    end
    tempans = qHqFunction(tempdata);
```

```
    trempresample(:,i+1) = tempans(:,2);
end
%   The simulated c(q) curves have been created and saved as tempresample()
H_sim1000=tempresample
save('Hcurve_sim1000.mat','H_sim1000')    %   this is c(q) for GLD data
%   Calculate average and variance in points along the c(q) curve as:
mean_qHq = tempresample(:,1);
stddev_qHq = tempresample(:,1);
for i = 1:numpoints
    mean_qHq(i,2) = mean(tempresample(i,2:numsim));
    stddev_qHq(i,2) = std(tempresample(i,2:numsim));
end
save('monosim_varCcurve.mat','stddev_qHq')    %   this is c(q) for GLD data
%   this replicates results from Fisher, Calvet

%%   Next: Calculate c''(q) for all simulated curves
Hprime = [0,0];
Hmat = tempresample;
h = Hmat(2,1)-Hmat(1,1);
for i = 2:length(Hmat(:,1))-1
    Hprime(i-1,1) = Hmat(i,1);
    Hprime(i-1,2) = (1/(2*h))*(Hmat(i+1,2) - Hmat(i-1,2));
end
for j = 3:length(Hmat(1,:))
    for i = 2:length(Hmat(:,1))-1
        Hprime(i-1,j) = (1/(2*h))*(Hmat(i+1,j) - Hmat(i-1,j));
    end
end
%
Hdprime = [0,0];
for i = 2:length(Hprime(:,1))-1
    Hdprime(i-1,1) = Hprime(i,1);
    Hdprime(i-1,2) = (1/(2*h))*(Hprime(i+1,2) - Hprime(i-1,2));
end
for j = 3:length(Hprime(1,:))
    for i = 2:length(Hprime(:,1))-1
        Hdprime(i-1,j) = (1/(2*h))*(Hprime(i+1,j) - Hprime(i-1,j));
    end
end
save('Hdprime_sim1000.mat','Hdprime')
%   This saves all of the simulations for numsim=1000.
save_temp = Hdprime(:,1:51)
save('Hdprime_sim50.mat','save_temp')
%   Result: compare c'' for empiric and simulated data

%%  Goodness of fit test
%   psuedo code: get second derivative of c(q) and at each point in domain
%   calculate the absolute value of c''.  This represents the distance from
%   zero.  The measure for goodness of fit is the total of the distances.
%   I interpret the top 5th percentile as the critical value.

clear all
load('Hdprime_empiric.mat')
load('Hdprime_sim1000.mat')

%   Calculate goodness of fit statistic for Hdprime
```

```
numsim = length(Hdprime(1,:))-1;
numq = length(Hdprime(:,1));
D = 0;
for i = 1:numsim
    gfit = 0;
    for j = 1:numq
        gfit = gfit + abs(Hdprime(j,i+1));
        %gfit = gfit + abs(Hdprime(j,i+1) - Hdprime_avg(j,2));
        %   Another possible test statistic
    end
    D(i,1)=gfit;
end
crit = sort(D);
%   Result: critical values for test statistic
D95=crit(1000-50)/length(Hdprime_empiric)
%   This gives the 95% rejection value: 0.0347
D99=crit(1000-10)/length(Hdprime_empiric)
%   This gives the 99% rejection value: 0.0406


%   Result: report distribution of test stat D.
[n,xout]=hist(D,25);
[xout' n']      %these are values of histogram

%   Result: apply test stat to GLD empirical data.
gfit = 0
for j =1:length(Hdprime_empiric)
    gfit = gfit + abs( Hdprime_empiric(j,2));
    %gfit = gfit + abs( Hdprime_empiric(j,2) - Hdprime_avg(j,2));
end
D_empiric = gfit    %   Observed stat is 1.7951, larger than critical value

%%  Simulation to calculate Power Curves
%   For several values of (Hurst exponent H) simulate many paths.  For each
%   path, calculate number of times that observed test statistic is larger
%   than critical value.  When larger, represents correct rejection of null
%
%
clear all
%   load('Hdprime_avg.mat');    %   Alternative test possible
keyvec = 0;

for h_index = 0.5:.01:0.95    %   Key Parameters
    H = h_index;     % Hurst Exponent
    ln_mu = 0.05;  % Drift from GBM equation
    ln_sigma = 0.1;    %   Vol from GBM equation

    %   Secondary Parameters
    numsim = 100;
    b = 2;
    k = 10;

    %   Calls to create data
    temp0 = mmar(b, k, H, ln_mu, ln_sigma);
    temp1 = diff(temp0);     %lose one observation here
```

```
for i = 1: length(temp0)-1
    temp2 = cumsum(temp1(1:i));
    tempdata(i) = exp( temp2(i)/100);
end

%   Calls to create c(q) curve for each simulated path
tempans = qHqFunction(tempdata);
numpoints = length(tempans);
tempresample = tempans;
%plot(tempans(:,1),tempans(:,2))

for i = 2:numsim
    %   Create geometric data from additive
    temp0 = mmar(b, k, H, ln_mu, ln_sigma);
    temp1 = diff(temp0);      %lose one observation here

    for j = 1: length(temp0)-1
        temp2 = cumsum(temp1(1:j));
        tempdata(j) = exp( temp2(j)/100);
    end
    tempans = qHqFunction(tempdata);
    tempresample(:,i+1) = tempans(:,2);
end

%
Hprime = [0,0];
Hmat = tempresample;
h = Hmat(2,1)-Hmat(1,1);
for i = 2:length(Hmat(:,1))-1
    Hprime(i-1,1) = Hmat(i,1);
    Hprime(i-1,2) = (1/(2*h))*(Hmat(i+1,2) - Hmat(i-1,2));
end

for j = 3:length(Hmat(1,:))
    for i = 2:length(Hmat(:,1))-1
        Hprime(i-1,j) = (1/(2*h))*(Hmat(i+1,j) - Hmat(i-1,j));
    end
end
%
Hdprime = [0,0];
for i = 2:length(Hprime(:,1))-1
    Hdprime(i-1,1) = Hprime(i,1);
    Hdprime(i-1,2) = (1/(2*h))*(Hprime(i+1,2) - Hprime(i-1,2));
end

for j = 3:length(Hprime(1,:))
    for i = 2:length(Hprime(:,1))-1
        Hdprime(i-1,j) = (1/(2*h))*(Hprime(i+1,j) - Hprime(i-1,j));
    end
end

%   Calculate goodness of fit statistic for Hdprime
numsim = length(Hdprime(1,:))-1;
numq = length(Hdprime(:,1));
D = 0;
```

```
    for i = 1:numsim
        gfit = 0;
        for j = 1:numq
            gfit = gfit + abs(Hdprime(j,i+1));
        end
        D(i,1)=gfit
    end
    key = sum(D>=1.3911);%  %   This is power
    int_index = floor(h_index*100-49);% h_index*20 - 9 % for 0.5:.01:0.95
    keyvec(int_index,1)=h_index;
    keyvec(int_index,2)=key;
end
%   The power of the test for each value of H is contained in keyvec.
powercurve = keyvec
save('PowerCurve_100sim.mat','powercurve')

%%  This concludes program
```

```
%Function calculates a diagnostic tool for the MMAR.
%   Equation of interest is log(X(q,dt)) = const + H(q) log(dt),
%   where dt is step size, and X(q,dt) is the average of:
%   abs(X(t)/X(t-dt)-1)^q.
%   Function returns estimate of H(q) for range of q-values.
%
%
% See the following paper
% A Multifractal Model of Asset Returns by B Mandelbrot - 1997
%
%
function [ qHmat] = qHqFucntion(price)
%   Basic Parameters
n = length(price);
offsetd = zeros(1,1);
counter = 1;     %   Used to calculate X(q,a)
step = 2;    %   Track current value of offset distance dt
top = 50;     %   Upper bound for dt (tau) in loop

%   Assemble data into tau-offset format
for j = 2:top
    step = j;
    counter = 1;
    while counter*step+1<n,
        %Note: I use log approx for percent change.
        %pctchange = price(1+counter*step)/price(1+step*(counter-1))-1;
        pctchange = log(price(1+counter*step)/price(1+step*(counter-1)));
        offsetd(counter,j-1) = pctchange;
        counter = counter + 1;
    end
end


%   Create matrix to store values of dt, each row is a value of dt
for i = 1:top-1
    regstep(i) = i;
end

%   Calculate the Q-power of price changes, for each level of dt
q = 0.375;
qstop = 5.5;
qstep = 0.125;
numstep = qstop/qstep;
regdata=zeros(numstep,top-1);

for l = 1:numstep       %loop over Q
    for m = 1:top-1     %loop over step size dt
       temps = sum(offsetd(:,m) ~=0,1); %sample mean for nonzero entries
       moment = q;
       regdata(m,l) = sum(abs(offsetd(:,m)).^(moment) / temps); %crucial
    end
    %   do regression: log(regdata(step,Q)) against log(step) for fixed Q.
    tempreg = polyfit(log(regstep), log(regdata(:,l)'),1);
    finalanswer(l,1) = q;
    finalanswer(l,2) = tempreg(1);
```

```
    q = q+qstep;
end

qHmat = finalanswer;
%%  Graphing functionality:

%hold on
for i = 1:numstep
    %   Use one of the following at a time:
    %plot( log(1:top-1), log(regdata(:,i)))
        % Stacked curves S(t,q) vs t.  As in Matteo (2007).
    %semilogx(log(regdata(:,i)) - log(regdata(1,i)))
        % Curves with same intercept as in Calvet and Fisher (2002).
end

end
```