4.0 Introduction

In the previous chapter, we analyzed stock price behaviour using various spectral methods such as power spectrum and cross-spectrum. Here we make an attempt to carry out a more exploratory analysis of the same using the sophisticated wavelet techniques. The major emphasis is on the investigating the time-scale decompositions, time frequency analysis and denoising. Besides, we also present a discussion on various dimensions of wavelet analysis that will lead to empirical analysis in subsequent sections.

The rest of the chapter is organized as follows: sections 4.1 to 4.7 are devoted to discussion on various technical aspects of wavelets. Section 4.8 documents the implementation procedure of the discrete wavelet system. We undertake the empirical analysis in section 4.6 followed by concluding remarks in section 4.10.

4.1 Wavelet Analysis

Wavelet analysis is characterized by a wavelet. A wavelet is a small wave, which has its energy concentrated in time to give a tool for the analysis of transient, non-stationary or time varying phenomenon. It still has the oscillating wave like characteristic (as Fourier analysis) but also has the ability to allow simultaneous time and frequency analysis with a flexible mathematical foundation.

There are two types of wavelets defined on different normalization and orthogonalization rules, namely, father wavelets $p$ (scaling function) and mother wavelets $y$ (wavelets). The father wavelet integrates to a constant and the mother wavelets integrates to zero: that is,
4.1.1 Wavelet System

There are many different wavelets systems (Harr, Daubechies, Symmlet etc) that can be used effectively, but all seem to have the following three general characteristics:

i. A wavelet system is a set of building blocks to construct or represent a signal or function. It is a two dimensional expansion set (usually a basis) for some class of one or higher dimensional signals. In other words, if the wavelet set is given by $\psi_{j,k}(t)$ for indices $j,k = 1, 2, ...$, a linear expansion would be $f(t) = \sum_j \sum_k a_{j,k} \psi_{j,k}(t)$ for some set of coefficients $a_{j,k}$.

ii. The wavelet expansion gives a time-frequency localization of the signal. This means most of the energy of the signal is well represented by a few expansion coefficients $a_{j,k}$.

iii. The calculation of the coefficients from the signal can be done efficiently. It turns out that many wavelet transforms (the set of expansion coefficients) can be calculated with order of $N$ [i.e.$O(N)$] operations. This means the number of floating point multiplications and additions increase linearly with the length of the signal. More general wavelet transforms require $O(N\log(N))$ operations, the same as for the fast Fourier transform (FFT).

Virtually all wavelet systems have these characteristics. Where the Fourier series maps a one dimensional function of a continuous variable into a one dimensional sequence of coefficients, the wavelet expansion maps it into a two dimensional array of coefficients. We will see that it is the two dimensional representation that allows the localizing the signal in both time and frequency.
Fourier series expansion localizes in frequency in that if a Fourier series expansion of a signal has only large coefficient, then the signal is essentially a single sinusoid at the frequency determined by the index of the coefficients. The simple time domain representation of the signal itself gives the localization in time. If the signal is a simple pulse, the location of that pulse is the localization in time. A wavelet representation will give the location in both time and frequency simultaneously.

4.1.2 Specific Characteristics of the Wavelet System

There are two additional characteristics that are more specific to wavelet expansion.

1. All the so called first generation wavelet systems are generated from a single scaling function or wavelet by simple scaling and translation. The two dimensional parameterization is achieved from the function (mother wavelet) $\psi(t)$ by

$$
\psi_{j,k}(t) = 2^{j/2} \psi(2^j t - k) \quad j, k \in \mathbb{Z}
$$

where $\mathbb{Z}$ is the set of all integers and the factor $2^{j/2}$ maintains a constant norm independent of scale $j$. This parameterization of the time or space location by $k$ and the frequency or scale by $j$ turns out to be extraordinarily effective.

2. All most all wavelet systems also satisfy the multiresolution conditions. This means that if a set of signals can be represented by a weighted sum of $\psi(t - k)$, then a larger set can be represented by weighted sum of $\psi(2t - k)$. In other words, if the basic expansion signals are made half as wide and translated in steps half as wide, they will represent a larger class of signals exactly or give a better approximation of any signal.

The operations of the translation of scaling seem to be basic to many practical signals and signal generating processes and their use is one of the reasons that wavelets are efficient expansion function. If the index $k$ changes, the location of the wavelet moves along the horizontal axis, which allows the expansion to explicitly represent the location of events in time or space. If the index $j$ changes, the shape of the wavelet changes in scale, which allows a representation of detail or resolution. For the Fourier series and transform, the expansion function (bases) are
chosen, then the properties of resulting transform are derived and analyzed. For the wavelet system, these properties or characteristics are mathematically required, and then the resulting basis functions are derived. Wavelet analysis is well suited to transient signals. Fourier analysis is appropriate for periodic signals or for signals whose statistical characteristics do not change over time (stationary). It is the localizing property of the wavelets that allow a wavelet expansion of a transient event to be modeled with a small number of coefficients. This turns out to be a very useful in applications.

4.2 The Discrete Wavelet Transform

Any function \( f(t) \in L^2(\mathbb{R}) \) could be written as

\[
  f(t) = \sum_{k=-\infty}^{\infty} c_k \varphi_k(t) + \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} d(j,k) \psi_{j,k}(t)  \quad ...(4.2)
\]

Using equation (4.1) a more general statement of expansion (4.2) can be given by

\[
  f(t) = \sum_{k} c_{j_0}(k) 2^{j_0/2} \varphi(2^{j_0}t - k) + \sum_{k} \sum d_{j_0}(k) 2^{j_0/2} \psi(2^{j_0}t - k)  \quad ...(4.3)
\]

or

\[
  f(t) = \sum_{k} c_{j_0}(k) \varphi_{j_0,k}(t) + \sum_{k} \sum_{j=j_0} d_{j}(k) \psi_{j,k}(t)  \quad ...(4.4)
\]

The coefficients in this wavelet expansion are called the discrete wavelet transform (DWT) of the signal \( f(t) \). These wavelet coefficients completely describe the original signal and can be used in a way similar to Fourier series coefficients for analysis, description, approximation and filtering. These coefficients can be calculated by inner products

\[
  c_j(k) = \langle f(t), \varphi_{j,k}(t) \rangle = \int f(t) \varphi_{j,k}(t) dt  \quad ...(4.5)
\]

and

\[
  d_j(k) = \langle f(t), \psi_{j,k}(t) \rangle = \int f(t) \psi_{j,k}(t) dt  \quad ...(4.6)
\]

The DWT is similar to a Fourier series but, in many ways, is much more flexible and informative. It can be made periodic like a Fourier series to represent periodic
signals efficiently. However, unlike Fourier series, it can be used directly on non-periodic transient signals with excellent results.

### 4.2.1 Analysis –From Fine Scale to Coarse Scale

In order to work directly with the wavelet coefficients, we will derive the relationship between the expansion coefficients at a lower scale level in terms of those at a higher scale level.

The basic recursive equations are:

\[
\begin{align*}
\varphi(t) &= \sum_n h(n)\sqrt{2}\varphi(2t-n) \quad \ldots(4.7) \\
\psi(t) &= \sum_n \tilde{h}(n)\sqrt{2}\varphi(2t-n) \quad \ldots(4.8)
\end{align*}
\]

and assuming a unique solution exists, we scale and translate the time variable to give

\[
\varphi(2^j t - k) = \sum_n h(n)\sqrt{2}\varphi(2(2^j t - k) - n) = \sum_n h(n)\sqrt{2}\varphi(2^{j+1} t - 2k - n) \quad \ldots(4.9)
\]

which after changing variables \(m=2k+n\), becomes

\[
\varphi(2^j t - k) = \sum_m h(m-2k)\sqrt{2}\varphi(2^{j+1} t - m) \quad \ldots(4.10).
\]

If we denote \(\nu_j\) as

\[
\nu_j = \text{span} \{2^{j/2} \varphi(2^j t - k)\} \quad \ldots(4.11)
\]

then

\[
f(t) \in \nu_{j+1} \Rightarrow f(t) = \sum_k c_{j+1}(k)2^{(j+1)/2}\varphi(2^{j+1} t - k) \quad \ldots(4.12)
\]

is expressible at a scale of \(+7\) with scaling functions only and no wavelets. At one scale lower resolution, wavelets are necessary for the detail not available at scale \(j\). we have

\[
f(t) = \sum_k c_j(k)2^{j/2}\varphi(2^j t - k) + \sum_k d_j(k)2^{j/2}\psi(2^j t - k) \quad \ldots(4.13)
\]
where the $2^{j/2}$ terms maintain the unity norm of the basis functions at various scales. If $\varphi_{j,k}(t)$ and $\psi_{j,k}(t)$ are orthonormal, the $j$ level scaling coefficients are found by taking the inner product

$$c_j(k) = \langle f(t), \varphi_{j,k}(t) \rangle = \int f(t) 2^{j/2} \varphi(2^j t - k) dt$$  \hspace{1cm} (4.14)

which, by using (4.10) and interchanging the sum and integral, can be written as

$$c_j(k) = \sum_m h(m-2k) \int f(t) 2^{(j+1)/2} \varphi(2^{j+1} t - m) dt$$  \hspace{1cm} (4.15)

but the integral is the inner product with the scaling function at a scale of $j+1$ giving

$$c_j(k) = \sum_m h(m-2k) c_{j+1}(m)$$  \hspace{1cm} (4.16)

The corresponding relationship for wavelet coefficients is

$$d_j(k) = \sum_m h(m-2k) c_{j+1}(m)$$  \hspace{1cm} (4.17)

where $h(n)$ and $\tilde{h}(n)$ are filter coefficients for scaling and wavelets respectively.

### 4.2.2 Synthesis – From Coarse Scale to Fine Scale

As one would expect, the reconstruction of the original fine scale coefficients of the signal can be made from a combination of the scaling function and wavelet coefficients at a coarse scale resolution. This is derived by considering a signal in the $j+1$ scaling function space $f(t) \in V_{j+1}$. This function can be written in terms of the scaling function as

$$f(t) = \sum_k c_{j+1}(k) 2^{(j+1)/2} \varphi(2^{j+1} t - k)$$  \hspace{1cm} (4.18)

or in terms of the next scale (which also requires wavelets) as

$$f(t) = \sum_k c_j(k) 2^{j/2} \varphi(2^j t - k) + \sum_k d_j(k) 2^{j/2} \psi(2^j t - k)$$  \hspace{1cm} (4.19).

Substituting (4.7) and (4.8) into (4.19) gives
Because all of these functions are orthonormal, multiplying (4.18) and (4.20) by \( \varphi(2^{j+1}t - k) \) and integrating evaluates the coefficient as

\[
c_{j+1}(k) = \sum_m c_j(m)\hat{h}(k-2m) + \sum_m d_j(m)\hat{h}(k-2m)
\]  

Therefore, for the use of any wavelet, it is not always necessary to know the specific form of the function. If we know the filter coefficients for the scaling function \( h(n) \) of a particular wavelet, we can get the filter coefficients of the wavelet, \( h(n) \) by using \( h(n) = (-1)^n h(N-1-n) \). With the use of \( h(n) \) and \( h(n) \) we can solve for wavelet coefficients \( c_j \) and \( d_{j,k} \)’s. Except in some special cases, there is no analytical formula for computing a wavelet function. Instead, wavelets are derived using a special two-scale dilation equation. For father wavelet \( \varphi(t) \), the dilation equation is defined by

\[
\varphi(t) = \sum_n h(n)\sqrt{2}\varphi(2t - n)
\]  
The mother wavelet \( \psi(t) \) can similarly be obtained from the father wavelet by the relationship

\[
\psi(t) = \sum_n \hat{h}(n)\sqrt{2}\varphi(2t - n)
\]  
The coefficients \( h(n) \) and \( \hat{h}(n) \) are the low-pass and high-pass filter coefficients defined as:

\[
h(n) = \frac{1}{\sqrt{2}} \int \varphi(t)\varphi(2t - n)dt
\]  

\[
\hat{h}(n) = \frac{1}{\sqrt{2}} \int \psi(t)\varphi(2t - n)dt
\]
4.3 Wavelet Approximation

Any function $f(t)$ in $L^2(R)$ to be represented by a wavelet analysis can be built up as a sequence of projections onto father and mother wavelets generated from through scaling and translation as follows:

\[
\varphi_{j,k}(t) = 2^{-j/2}\varphi(2^{-j}t - n) = 2^{-j/2}\varphi\left(\frac{t - 2^j n}{2^j}\right) \quad \ldots(4.26)
\]

\[
\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - n) = 2^{-j/2}\psi\left(\frac{t - 2^j n}{2^j}\right) \quad \ldots(4.27)
\]

The wavelet representation of the signal or function $f(t)$ in $L^2(R)$ can be given as:

\[
f(t) = \sum_k c_{j,k}\varphi_{j,k}(t) + \sum_k d_{j,k}\psi_{j,k}(t) + \sum_k d_{j-1,k}\psi_{j-1,k}(t) + \ldots + \sum_k d_{1,k}\psi_{1,k}(t) \quad \ldots(4.28)
\]

where $J$ is the number of multiresolution components, and $k$ ranges from 1 to the number of coefficients in the specified component. The coefficients $c_{j,k}$, $d_{j,k}$, $\ldots$, $d_{1,k}$ are the wavelet transform coefficients given by the projections

\[
c_{j,k} \approx \int \varphi_{j,k}(t)f(t)dt \quad \ldots(4.29)
\]

\[
d_{j,k} \approx \int \psi_{j,k}(t)f(t)dt, \quad \text{for} \quad j = 1, 2, \ldots, J \quad \ldots(4.30).
\]

The magnitude of these coefficients reflects a measure of the contribution of the corresponding wavelet function to the total signal. The basic functions $\varphi_{j,k}(t)$ and $\psi_{j,k}(t)$ are the approximating wavelet functions generated as scaled and translated versions of $\varphi$ and $\psi$, with scale factor $2^j$ and translation parameter $2^j k$, respectively.

The scale factor $2^j$ is also called the dilation factor and the translation parameter $2^j k$ refers to the location. Here $2$ is a measure of the scale or width of the functions $\varphi_{j,k}(t)$ and $\psi_{j,k}(t)$. That is, the larger the index $j$, the larger the scale factor $2^j$, and hence the function get shorter and more spread out. The translation parameter $2^j k$ is matched to the scale parameter $2$ in that as the functions $\varphi_{j,k}(t)$ and $\psi_{j,k}(t)$ get wider, their translation steps are correspondingly larger.
4.4 Multiresolution Analysis

The discrete wavelet transform (DWT) calculates the coefficients of the wavelet representation (4.28) for a discrete signals \( f_1, f_2, \ldots, f_n \) of finite extent. The DWT maps the vector \( f = (f_1, f_2, \ldots, f_n) \) to a vector of \( n \) wavelet coefficients \( w = (w_1, w_2, \ldots, w_n) \).

The vector \( w \) contains the coefficients \( c_{j,k}, d_{j,k}, \ldots, d_{l,k} \) of the wavelet series representation (4.30). The coefficients \( c_{j,k} \) are called the smooth coefficients, representing the underlying smooth behaviour of the signal at the coarse scale \( 2^j \).

On the other hand, \( d_{j,k} \) are called the detailed coefficients, representing deviations from the smooth behaviour, where \( d_{j,k} \) describe the coarse scale deviations and \( d_{j,k}, d_{j-1,k}, \ldots, d_{1,k} \) provide progressively finer scale deviations.

In cases when \( n \) is divisible by \( 2^j \), there are \( n/2 \) coefficients \( d_{1,k} \) at the finest scale \( 2^1 = 2 \). At the next finest scale \( 2^2 = 4 \), there are \( n/4 \) coefficients \( d_{2,k} \). Likewise, at the coarsest scale, there are \( n/2^j \) coefficients each for \( d_{j,k} \) and \( c_{j,k} \). Summing up, we have a total of \( n \) coefficients:

\[
  n = n/2 + n/4 + \ldots + n/2^{l-1} + n/2^l + n/2^j.
\]

The number of coefficients at a scale is related to the width of the wavelet function. At scale 2, the translation steps are \( 2k \), and so \( n/2 \) terms are required in order for the functions \( \psi_{j,k}(t) \) to cover the interval \( 1 < t < n \). By similar reasoning, a summation involving \( \psi_{j,k}(t) \) requires just \( n/2! \) terms, and the summation involving \( \phi_{j,k}(t) \) requires only \( n/2! \) terms. The string of coefficients can be ordered from coarse scales as:

\[
  w = \begin{pmatrix}
    s_j \\
    d_j \\
    d_{j-1} \\
    \vdots \\
    d_1
  \end{pmatrix}
\]

...(4.31)

Each of the sets of coefficients in 'w' is called a ‘crystal’, and the wavelet associated with each coefficient is referred to as an ‘atom’.
The multiresolution decomposition of a signal can now be defined by using the product of the crystals and the corresponding wavelet atoms, namely:

\[
C_j(t) = \sum_k c_{j,k} \phi_{j,k}(t) \quad \ldots (4.32)
\]

\[
D_j(t) = \sum_k d_{j,k} \psi_{j,k}(t) \quad \text{for} \quad j = 1, 2, \ldots, J \quad \ldots (4.33)
\]

The functions (4.32) and (4.33) are called the smooth signal and the detail signals, respectively, which constitute a decomposition of a signal into orthogonal components at different scales. Similarly to the wavelet representation (4.28) of a signal in \( L^2(R) \), a signal \( f(t) \) can now be expressed in terms of these signals:

\[
f(t) = C_j(t) + D_j(t) + D_{j-1}(t) + \ldots + D_1(t) \quad (4.34)
\]

As each term in (4.34) represents components of the signal \( f(t) \) at different resolutions, it is called a multiresolution decomposition (MRD).

The coarsest scale signal \( C_0(t) \) represents a coarse scale smooth approximation to the signal. Adding the detail signal \( D_0(t) \) gives a scale \( 2^{J-1} \) approximation to the signal, \( C_{J-1}(t) \), which is a refinement of the coarsest approximation \( C_0(t) \). Further refinement can sequentially be obtained as:

\[
C_{j-1}(t) = C_j(t) + D_{j-1}(t) = C_j(t) + D_j(t) + D_{j-1} + \ldots + D_1(t) \quad (4.35)
\]

The collection \( \{C_0, C_{J-1}, C_{J-2}, \ldots, C_J\} \) provides a set of multiresolution approximations of the signal \( f(t) \).

### 4.5 Vanishing Moments of Wavelets

We now define the \( k^{th} \) moments of \( \phi(t) \) and \( \psi(t) \) as

\[
m(k) = \int t^k \phi(t) dt \quad \ldots (4.36)
\]

and

\[
m_1(k) = \int t^k \psi(t) dt \quad \ldots (4.37)
\]
and the discrete $k^{th}$ moments of $h(n)$ and $\hat{h}(n)$ as

$$\mu(k) = \sum_n n^k h(n) \quad \cdots (4.38)$$

and

$$\mu_1(k) = \sum_n n^k \hat{h}(n) \quad \cdots (4.39)$$

The partial moments of $h(n)$ are defined as

$$\nu(k,l) = \sum_n (2n + l)^k h(2n + l) \quad \cdots (4.40)$$

Note that $\mu(k) = \nu(k,0) + \nu(k,1)$.

From these equations, we obtain

$$m(k) = \frac{1}{(2^k - 1) \sqrt{2}} \sum_{l=1}^{k} \binom{k}{l} \mu(l) m(k-l) \quad \cdots (4.41)$$

which can be derived by substituting recursive equation into (4.36), changing variables, and using (4.38). Similarly, we obtain

$$m_1(k) = \frac{1}{2^k \sqrt{2}} \sum_{l=0}^{k} \binom{k}{l} \mu_1(l) m(k-l) \quad \cdots (4.42)$$

These equations exactly calculate the moments defined by the integrals in (4.36) and (4.37) with simple finite convolutions of the discrete moments with the lower order continuous moments.

If $\psi(t)$ is $K$-times differentiable and decays fast enough, then the first $K-1$ wavelet moments vanish i.e.

$$\left| \frac{d^k}{dt^k} \psi(t) \right| < \infty, \quad 0 \leq k \leq K \quad (4.43)$$

implies

$$m_1(k) = 0, \quad 0 \leq k \leq K \quad \cdots (4.44)$$
Unfortunately, the converse of this theorem is not true. However, we can relate the differentiability of $\psi(t)$ to vanishing moments. There exists a finite positive integer $L$ such that \[ \text{if } m_1(k) = 0 \text{ for } 0 < k < K - 1, \] then

\[ \left| \frac{d^P}{dt^P} \psi(t) \right| < \infty \quad \text{for } L \geq P > K. \]

For example, a three times differentiable $\psi(t)$ must have three vanishing moments, but three vanishing moments results in only one dimensional differentiability.

### 4.6 Types of Wavelets

**Haar Wavelets**

The first wavelet filter, Haar wavelet (Haar, 1910) remained in relative obscurity until the consequence of several disciplines to form what we now know in broad sense as wavelet methodology. It is the simplest wavelet with the scaling and wavelet functions which is as follows:

\[ \int \phi(t)dt = 1, \int \psi(t)dt = 0, \text{ and } \int \phi(t)\psi(t)dt = 0 \]

It is a filter of length 2 that can be succinctly defined by its scaling (low-pass) filter coefficients $h_0 = h_1 = 1/\sqrt{2}$ or equivalently by its wavelet (high-pass) filter coefficients $\hat{h}_0 = -\hat{h}_1 = 1/\sqrt{2}$.

The Haar wavelet has good properties such as simplicity, orthonormality and compact support. Although, the Haar wavelet is easy to visualize and implement, it is inadequate for most real world applications in that it is a poor approximation to an ideal band-pass filter. Also, it is discontinuous and so we have difficulty in approximately a smooth function by Haar wavelets with more regularity.

**Daubechies Wavelets**

Daubechies wavelets form an orthogonal basis with compact support. The Daubechies wavelet filters represent a collection of wavelets that improve upon the
frequency domain characteristics of the Haar wavelet (Daubechies, 1992). Daubechies derived these wavelets from the criterion of a compactly supported function with the maximum number of vanishing moments. In general, there are no explicit time-domain formulae for this class of wavelet filters; the simplest way to define the class of Daubechies wavelets is through the filtered coefficients. As the number of filter length increases, the Daubechies wavelets length increases. If the length of the filter is 2, then the wavelet is Daubechies - 2 (db2), which is identical to Haar wavelet. If the filter length is 4 and 6, then wavelets are Daubechies -4 (db4) and Daubechies - 6(db6) respectively. The (db) wavelets have following filtered coefficients:

\[
    h_0 = -\hat{h}_3 = \frac{1+\sqrt{3}}{4\sqrt{2}}, \quad h_1 = \hat{h}_2 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \quad h_2 = -\hat{h}_1 = \frac{3-\sqrt{3}}{4\sqrt{2}}, \quad \text{and} \quad h_3 = -\hat{h}_0 = \frac{1-\sqrt{3}}{4\sqrt{2}},
\]

and the (db6) wavelets filter coefficients are as follows:

\[
    h_0 = -\hat{h}_5 = 0.3327, \quad h_1 = -\hat{h}_4 = 0.8069, \quad h_2 = \hat{h}_3 = 0.4599, \quad h_3 = -\hat{h}_2 = 0.1350, \\
    -h_4 = -\hat{h}_1 = 0.0854, \quad h_5 = -\hat{h}_0 = 0.0352.
\]

The (db4) wavelet has one vanishing moment and the (db6) has two vanishing moments. One implication due to this property is that longer wavelet filters may produce stationary wavelet coefficient vectors from higher degree of non-stationary stochastic processes. We have mentioned in the previous section that the Haar wavelet is a poor approximation to an ideal band-pass filter but the level of approximation improves as filter length increases.

4.7 Wavelet Shrinkage

Separating noise from the signal is denoising. Wavelet shrinkage refers to thresholding, which is shrinkage of wavelet coefficients. The choice of a threshold in wavelet analysis is as important as the choice of a bandwidth in kernel smoothing. There are several thresholding rules. The key parameters in all thresholding rules is value of the threshold.
4.7.1 Value of the Threshold

This is an important step, which affects the quality of the compressed signal. The basic idea is to truncate the insignificant coefficients since less amount of information is contained. Then optimal thresholding occurs when the thresholding parameter is set to the noise level. Setting thresholding parameter less than noise level would allow unwanted noise to enter the estimate while setting thresholding parameter greater than noise level would destroy information that really belongs to the underlying signal. Donoho and Johnstone (1994) suggested a universal thresholding by setting

\[ t = \sigma \sqrt{2 \log N} \]

where \( \sigma \) is the standard deviation and \( N \) refers to total number of data points. The value of the threshold \( (t) \) should be calculated for each level of decomposition and only for the high-pass coefficients (low-pass coefficients are kept untouched so as to facilitate further decomposition).

**Hard Thresholding**

Literally interpreting the statement “keep or kill”, hard thresholding is a straightforward technique for implementing wavelet denoising. The hard thresholding function is easy to use and gives better reconstruction of discontinuities. The threshold value \( t \) is given by

\[ H(t,d) = \begin{cases} d & \text{if } |d| > t \\ 0, & \text{otherwise} \end{cases} \]

where \( d \) refers to the wavelet coefficients. This observation is not a continuous mapping and only affects input coefficients that are less than or equal to the threshold.

**Soft Thresholding**

The other standard technique for wavelet denoising is soft thresholding of the wavelet coefficient via
Instead of forcing wavelet coefficients to zero or leaving it untouched, soft thresholding pushes all coefficients towards zero. If the wavelet coefficient happens to be smaller in magnitude than the threshold, then it is set to zero as in hard thresholding. Thus, the operation of soft thresholding is a continuous mapping. The choice between these two thresholding rules depends upon what characteristics are desirable in the resulting estimate. For instance, if large spikes are present in the observed series, then hard thresholding will preserve the magnitude of these spikes while soft thresholding, because it affects all wavelet coefficients, will suppress them. On the other hand, soft thresholding will, in general produce a smoother estimate because all wavelet coefficients are being pushed towards zero. It is up to the practitioners to weigh these differences and apply the most appropriate thresholding rule.

4.8 Implementation of the Discrete Wavelet Transform (DWT)

Let \( x \) be a dyadic length vector \( (N = 2^J) \) of observations. The length \( N \) vector of discrete wavelet coefficients \( w \) is obtained via

\[
S(t, d) = \begin{cases} 0 & \text{if } |d| < t \\ \text{Sign} \ (d)(|d|-t), & \text{otherwise} \end{cases}
\]

where \( \text{Sign}(d) = \begin{cases} +1 & \text{if } d > 0 \\ 0 & \text{if } d = 0 \\ -1 & \text{if } d < 0 \end{cases} \)

Let \( x \) be a dyadic length vector \( (N = 2^J) \) of observations. The length \( N \) vector of discrete wavelet coefficients \( w \) is obtained via

\[
\text{vec} \left( \begin{array}{c} \hat{x} \\ \sqrt{2} \tilde{x} \\ \vdots \\ \sqrt{2^J} \tilde{x} \end{array} \right) = \left( \begin{array}{c} D_1 \\ \sqrt{2} D_2 \\ \vdots \\ \sqrt{2^J} D_J \end{array} \right) \left( \begin{array}{c} x \\ \sqrt{2} x \\ \vdots \\ \sqrt{2^J} x \end{array} \right) = \left( \begin{array}{c} D_1 x \\ \sqrt{2} D_2 x \\ \vdots \\ \sqrt{2^J} D_J x \end{array} \right)
\]

where \( \text{to is Nxn orthonormal} \) matrix defining the DWT. The vector of wavelet coefficients may be organized into \( J + 1 \) vectors,

\[
w = [d_1, d_2, ..., d_j, c_j]^T \quad \text{... (1)}
\]
where \( w_j \) is a length \( N/2^j \) vector of wavelet coefficients with changes on a scale of length \( \lambda_j = 2^{-j} \) and \( c_j \) is a length \( N/2^j \) vector of scaling coefficients on a scale of length \( 2 = 2\lambda_j \).

### 4.8.1 Pyramid Algorithm

In practice, the DWT is implemented via pyramid algorithm (Mallat, 1989) that starting with the data \( x_n \), filters a series using \( h \) and \( h' \), sub samples both the filter outputs to half of their original lengths, keeps sub sampled output from the \( h' \) as wavelet coefficients, and then repeats the above filtering operations on the sub sampled output from the \( h \) filter. Figure 4.1 gives a flow diagram for the first stage of the pyramid algorithm. The symbol \( \downarrow 2 \) means that every other value of the input vector is removed (down sampling by 2).

For each iteration of the pyramid algorithm, we require three objects: the vector \( x \), the wavelet filter \( h \) and the scaling filter \( h' \). The first operation of the pyramid algorithm begins by filtering (convolving) the data with each filter to obtain the wavelet \( (d_j) \) and scaling coefficients \( (c_j) \). The \( N \) length vector of observations has been high and low-pass filtered to obtain \( N/2 \) coefficients. The second step of the pyramid algorithm starts by defining the data to be the scaling coefficients \( c_j \) from the first iteration and apply the filtering operations as above to obtain the second level of wavelet \( (d_j) \) and scaling \( (c_j) \) coefficients. Now the length of the filtered coefficients is \( N/4 \). Keeping all vectors of wavelet coefficients and the final level of scaling coefficients, we have the following length decomposition

\[
\mathbf{w} = [d_1, d_2, c_2']
\]

After the third iteration of the pyramid algorithm, where we apply filtering operations to \( c_2 \), the decomposition now looks like

---

1. Wavelet coefficients are obtained by projecting the wavelet filter onto a vector of observations. Since Daubechies wavelets may be considered as generalized differences, we prefer to characterize the wavelet coefficients this way. For example, a unit scale Daubechies wavelet filter is a generalized difference of length one—that is, the wavelet filter is essentially taking the difference between two adjacent observations. We call this a wavelet scale of length \( \lambda_1 = 2^0 = 1 \). A scale two Daubechies wavelet filter is a generalized difference of length two—that is, the wavelet filter first averages adjacent pairs of observations and then takes the differences of these averages. We call this a wavelet scale of length \( \lambda_2 = 2^1 = 2 \). The scale length increases by powers of two as a function of scale.
This procedure may be repeated up to \( J \) times where \( J = \log_2(N) \) and gives the vector of wavelet coefficients. Inverting, the DWT is achieved through up-sampling the final level of wavelet and scaling coefficients, convolving them with their respective filters (wavelet for wavelet and scaling for scaling) and adding up the two filtered vectors. Figure 4.2 gives a flow diagram for the reconstruction of \( x \) from the first levels wavelet and scaling coefficient vectors. The symbol \( t^2 \) means that a zero is inserted before each observation in \( d_j \) and \( c_j \) (up sampling by 2). Starting with the final level of the DWT, up sampling the vector \( d_J \) and \( c_J \) will result in two new vectors:

\[
d_j^0 = [0 \ d_j]^T \quad \text{and} \quad c_j^0 = [0 \ c_j]^T.
\]

The next step of reconstruction involves up sampling to produce \( d_{j-1}^0 \) and \( c_{j-1}^0 \). This procedure may be repeated until the first level of wavelet and scaling coefficients have been up sampled and combined to produce the original vector of observations.

### 4.8.2 Partial Discrete Wavelet Transform

If the data are of dyadic length, it is not necessary to implement the DWT down to level \( J = \log_2(N) \). A partial DWT may be performed instead, that terminates at a level \( J_p < J \). The resulting vector of wavelet coefficient will now contain \( N - N/2^J \) wavelet coefficients and \( N/2^J \) scaling coefficients. When we are provided with a non-dyadic length time series, e.g., 368, which is divisible by \( 2^4 = 16 \) and therefore we may perform an order \( J_p = 4 \) partial DWT on it.

### 4.9 Empirical Analysis

By design, the wavelet's usefulness is its ability to localize data in time-scale space. At high scales (shorter time intervals) the wavelet has a small time support and is thus better able to focus on short lived, strong transients like discontinuities, ruptures and singularities. At low scales (longer time intervals), the wavelet's time
support is large making it suited for identifying long periodic features. Wavelets have a intuitive way of characterizing the physical properties of the data. At low scales, the wavelet characterizes the data's coarse structure; its long-run trend and pattern. By gradually increasing the scale, the wavelet begins to reveal more and more of the data's details, zooming in on its behavior at a point in time.

Wavelet analysis is the analysis of change. A wavelet coefficient measures the amount of information that is gained by increasing the frequency at which the data is sampled, or what needs to be added to the data in order for it to look like it had been measured more frequently. For instance, if a stock price does not change during the course of a week, the wavelet coefficients from the daily scale are all zero during that week.

Wavelet coefficients that are non-zero at high scales typically characterize the noise inherent in the data. Only wavelets at very fine scales will try to follow the noise, whereas the same at coarser scales are unable to pick up the high frequency nature of the noise. If both the low and high scaled wavelet coefficients are non-zero then something structural is occurring in the data. A wavelet coefficient that does not go to zero as the scale increases indicates a jump (non-differentiable) has occurred in the data. If the wavelet coefficients do go to zero then the series is smooth (differentiable) at this point. Because of its localization in time and scale, wavelets are able to extract relevant information from a data set while disregarding the noise that may be present. Given the recording errors that occur during short, intense trading periods, and transient shocks that are caused by news reports, this de-noising technique is important to financial data.

Very little needs to be known about the relevant information or the information that one wants to extract. Because the wavelet transform captures the characteristics of the data in a few wavelet coefficients, if the wavelet coefficients whose magnitude is less than some prescribed value are set to zero and the few non-zero wavelet coefficients are used to recreate the data, the resulting data set will contain only the relevant information.

The data considered for wavelet analysis are daily closing values of Sensex, National Index, S&P CNX Nifty and S&P CNX 500 (for detail see appendix A).
Though our sample period is from 2nd January 1991 to 31st December 2001 constituting of 2552 data points, for wavelet decomposition analysis, we have used two different sample periods. First one is the dyadic length sample period from 2nd January 1991 to 21st December 1999 with 2048 \( (2^{11}) \) data points. Dyadic length data series has one potential advantage of performing up to the last level of wavelet decomposition (here 10). Second type of sample period is from 2nd January 1991 to 27th September 2000 with 2240 (divisible by \( 2^6 = 64 \)) data points. In this case, we apply partial discrete wavelet decomposition and perform an order \( J_p = 6 \) partial DWT on it. Since our original sample period consists of 2552 data points (divisible by \( 2^3 = 8 \)), we cannot perform an order of more than three partial DWT on it. Therefore, we conveniently take 2240 (divisible by \( 2^6 = 64 \)) data points of the return series for the purpose of DWT up to level 6.

The movements of the original data series are shown in figures 4.3 and 4.6 respectively. For the interest of the reader and also for better comparison with the wavelet decomposition, some key dates are highlighted in figure 4.1. It is easily observed from the data series that there was a significant variation in stock prices during April - May 1992, possibly due to Harshad Meheta led rally and scam unravels thereafter. It has crossed the 5000 mark in December 1999 and also reached its highest mark so far in February 2000. From the return series it is evident that there are increases in variance in the first and the latter part of the series.

All the return series are now subjected to discrete wavelet transform by using Haar (equivalent to \( \text{db}_2 \)), Daubechies - 4 (\( \text{db}_4 \)) and Daubechies - 6 \( \text{db}_6 \)) wavelets. Haar DWT results for all the four indices are reported in figures 4.7 to 4.11. The figure 4.7 reports the results of Haar DWT of a dyadic length return series (here \( 2048 = 2^{11} \)). As we mentioned earlier in the technical section of the wavelet analysis, if the data series is of dyadic length \( 2^n \), then we perform up to the last level of decomposition. In this case, we may perform up to 10\(^{th}\) level of wavelet decomposition where we will have only with two coefficients for both high and low-pass filter. Moreover in this case of dyadic length return series, we may analyze time-scale decompositions from finest scale (first level) to coarsest scale (here 10\(^{th}\) level) and it will give clear picture of both high and low-frequency fluctuations.
The DWT results of Sensex return series in figure 4.5 are arranged in to level - 1, level - 2, ... up to level - 10 coefficients, where the last level represents the low pass coefficients. The return series are plotted on the upper row of figure 4.5. The wavelet coefficient vectors $d_1$, ... $d_{10}$, using Haar wavelet are shown in the lower part of the figure 4.5. The first scale of the wavelet coefficient $d_1$ are filtering out the high frequency fluctuations by essentially looking at adjacent differences in the data. There is a large group of rapidly fluctuating returns between observations 220 and 270. A small increase in the magnitude $d_2$ is also observed between observations 250 and 300, but smaller than the unit scale coefficients. This vector of wavelet coefficients is associated with changes of scale $f_a$. The higher scale (low-frequency) vectors of wavelet coefficients $d_3$ to $d_6$ indicate variations from zero, which implies that the Sensex return series exhibits low frequency oscillations. The next level of wavelet coefficients $d_7$ and $d_8$, after differencing the 64 and 128 trading days averages of returns, show a quasi-periodic behaviour. The coarsest scale wavelet coefficient $d_9$ and $d_{10}$ show only the four and two data points respectively, whose interpretation is not of much use.

The same Haar wavelet decomposition is performed in non-dyadic length of Sensex return series and are provided in figure 4.8. The length of the return series is $N = 2240$, which is divisible by $2^6 = 64$ and therefore we perform an order $J_p = 6$ partial DWT on it. The DWT results are arranged in to level-1, level-2, up to level-6. Wavelet coefficients, where the last level represents the low-pass coefficients. The level-1 coefficients are the differences of the nearest neighborhood observations of the return series multiplied by Haar wavelet filter coefficient. The next scale coefficients are the differences of the nearest neighborhood averages. To be precise, finest scale (level-1) coefficients capture day-by-day fluctuations; the level-2 coefficients represent the differences of the two day averages and analogously level-6 wavelet coefficients captures the differences of 32-day averages of the return series. Hence, it is clear that the first scale wavelet coefficients filter out high frequency fluctuations by looking at the adjustment differences in data. Our results show that there is a large group of rapidly fluctuating returns between observations 220 and 270 and further there is also a group of fluctuations in the later part of the return series. Analogous results are
obtained in second and third levels of wavelet decompositions but detected fluctuations are smaller than the first level coefficients. A number of notable features appear at the $4^{th}$ and $5^{th}$ levels of decomposition, which shows that there are substantial differences between eight and sixteen days averages of the returns over the whole series. Higher scale wavelet coefficients indicate slow variations from zero, which implies low frequency fluctuations in return series. Interestingly, the $6^{th}$ level coefficients, after differencing the 32 trading days averaging in returns, show a quasi-periodic behaviour. As we go for higher level of wavelet decomposition, both high and low-pass coefficients become smooth which is quite obvious from the very nature of averaging and differencing the wavelet of each scale.

This analysis clearly indicates the usefulness of the time-scale decompositions and multi-scale nature of the wavelets. In the stock market, there are traders who take a very long term view and consequently concentrate on what are termed ‘market fundamentals’; these traders ignore short-term phenomena. For them, the high-level wavelet coefficients are very useful and they are more concerned about the same. In contrast, other traders are trading on a much shorter time-scale and as such are interested in temporary deviations of the market from its long-term path. Their decisions have a time horizon of a few months to a year; so they are interested in middle level wavelet decompositions of the return series. And yet other traders are in the market for whom a day is a long time and consequently concentrate on day-by-day fluctuations. Therefore, low level of wavelet coefficients of return series are more useful for them in the stock market. As we have discussed earlier, wavelet coefficients that are non-zero at high scales typically characterize the noise inherent in the data series. Only wavelets at very fine scales will try to follow the noise, whereas those wavelets at coarser scales are unable to pick up the high frequency nature of the noise. If both the low and high scaled wavelet coefficients are non zero then something structural is occurring in the data series. In this case, the large group of fluctuating returns between observations 220 and 270 indicated at all levels of wavelet decompositions provides some insight in to the Indian stock market. These large swings in stock returns were probably due to Harshad Meheta led rally and scam unraveled subsequently during March-May 1992. Again, the large group of rapid fluctuations in Sensex return series in the later part indicated at various scales of wavelet decomposition may be due to Kargil war
followed by lifting up of the US economic sanctions and upswing in the Indian stock market. This bullish trend in the Indian stock market during the December 1999 to February 2000 was due to favourable economic condition. The large peaks detected by some levels of wavelet decomposition analysis between observations 1950 to 2000 match with the Sensex reaching its all-time high of above 6000 mark and subsequent downslide due to union budget 2000-01.

The same Haar wavelets decomposition has been performed for other three Indian stock market indices such as National Index, S&P CNX Nifty and S&P CNX 500 and is provided from figures 4.9 to 4.11. The interpretations for each of the vectors wavelet coefficients is the same as in case of Sensex return series. In all the above cases of wavelet decompositions, each level low-pass coefficients gives a smoothed replication of the original return series. As shown in above analyzed figures, we plot the low-pass (scaling) coefficients vector of only large level \( c_6 \) and it also reflects most of the fluctuating features of the original return series in a smooth manner. The most interesting findings from the various levels of wavelet decompositions is that with least number of data points, we are able to analyze the original return series. For example, in the first scale of wavelet decomposition the vector of wavelet coefficients consist of data points half of the return series, but still it gives a clear picture about the various useful features of the original series. In similar fashion, the sixth level wavelet coefficient vector with 35 points able to give a overall idea about the original return series.

In the next step we present the reconstruction results of the decomposed return series from figures 4.12 to 4.15. The idea behind the reconstruction of the wavelet return series is to show how wavelet coefficients are able to capture the various properties of the return series and then reconstruct them. Figure 4.12 reports the reconstructed wavelet series of the Sensex return series. For comparison, the return series is plotted in the upper row of the figure 4.12. The reconstructed series from wavelet coefficient vectors are \( R_{d1}, R_{d2}, \ldots, R_{d6} \), using the Haar wavelets are shown in the lower part of the figure 4.12. The first scale wavelet reconstructed series \( R_{d1} \) shows exactly similar behaviour as return series by capturing all the high frequency fluctuations. The most interesting finding is that if we use the wavelet reconstructed series instead of original return series, it gives better results
in economics and finance, which will be discussed in chapter V. The most promising advantages of the wavelet reconstructed series is that it removes the statistical anomalies by recognizing the potential for relationship between variables to be at scale level, not at the aggregate level. That is, one should recognize that the relationship between two variable series depending on the time scale that is involved. Another advantage is that if we are interested in fluctuations and try to find out the relationship between two variables by essentially looking at fluctuations, e.g., fluctuations due to news or other factors, wavelet reconstructed series is very much useful. The most important example in the stock market is to test for spillover effects across stock markets, where we are interested in relating fluctuations of two series, which will be discussed in chapter V. Similarly, second scale of wavelet series is reconstructed with wavelet coefficients which is $\frac{1}{4}$ of the original series and still it gives a clear idea about the original return series. In a similar fashion all the levels of wavelet coefficients vectors reconstruct their corresponding return series. Interestingly, the 6th scale wavelet coefficient vector even with 35 observations is able to pinpoint all the low-frequency fluctuations of the original Sensex return series. The same reconstruction is performed using stock indices such as National Index, S&P CNX Nifty and S&P CNX 500 and are provided in figures 4.13 to 4.15 respectively. The interpretations for each index reconstructed return series is the same as in case of Sensex return series.

Figures 4.16 to 4.19 display the wavelet decompositions using the $db_4$ wavelet filter for the original return series. As we have discussed earlier in technical session, $db_4$ wavelet filter has one vanishing moment, i.e., it does not see straight-line part of a series, whereas Haar does not see a constant part of a series. If some part of the series is smooth then Haar wavelet coefficients go to zero and a wavelet coefficient that does not go to zero indicates a jump has occurred in the data. In addition to that if some portion of the return series is straight then $db_4$ wavelet coefficients go to zero. One implication of this property is that if we are interested in high fluctuations only, then $db_4$ wavelet coefficients are very much useful. Figure 4.16 shows the $db_4$ wavelet decomposition coefficient vectors for Sensex return series. The analysis for each of the vectors of coefficients is the same as the case of the Haar wavelet filter. The wavelet coefficients will be different given the length of the filter is now four versus two, should isolate features in a specific frequency
interval better since the \( \text{db}_4 \) is a better approximation to an ideal band-pass filter over the Haar wavelet (Genacy et al. 2001). The wavelet coefficients in the \( \text{db}_4 \) basis capture the fluctuations very well after removing the straightline part of the return series.

To provide an analysis of the \( \text{db}_6 \) wavelet basis, we consider four return series used in this study and results are displayed in figures 4.20 to 4.23. The \( \text{db}_6 \) wavelets satisfy the criterion of a compactly supported basis set with two vanishing moments. This shows that \( \text{db}_6 \) wavelet removes the patchiness of the data and pinpoints zig-zag fluctuations in the data series. Similarly, as we go on applying longer wavelet filters, we get higher order vanishing moments wavelet coefficients. One implication of this property is that longer wavelet filters may produce stationary wavelet coefficient vectors from higher degree non-stationary stochastic processes. The longer wavelet filters induce significant amounts correlation between adjustment coefficients, thus produce even smoother vectors of wavelet and scaling coefficients.

The most useful property of higher order vanishing moments wavelets are that they remove patchiness from the data and pinpoints higher order fluctuations. The longer wavelet filters separate out order out of chaos and fractals. As it is clear if we cannot describe some part of the curve by polynomials the there may be evidences for chaotic behavior and fractal nature in the curve. Thus, the longer wavelet filters with the property of higher order vanishing moments separate out ordered part like regular and random changes from chaos and fractals. In this study we have not gone for finding out chaos and fractal nature of a curve, which may be taken as a scope for further research.

A critical innovation in estimation that is introduced by wavelets, although by no means necessarily restricted to wavelets is the idea of shrinkage. Traditionally in economic analysis the assumption has universally been made that the signal \( f(t) \), is smooth and the innovations, \( \epsilon(t) \) are irregular. Consequently, it is a natural first step to consider extracting the signal \( f(t) \) from the observed signal \( y(t) = f(t) + \epsilon(t) \). By locally smoothing \( y(t) \). However, when the signal is as, or even more irregular than the noise such a procedure no longer provides a useful
approximation to the signal. The process of smoothing to remove the contamination of noise distorts the appearance of the signal itself. When the noise is below a threshold and the signal variation is well above the threshold. One can isolate the signal from the noise component by selectively shrinking the wavelet coefficient estimates (Donoho and Johnstone, 1995; Donoho et al, 1995). If we want the probability of any noise appearing in our signal return series to be as small as possible then applying the wavelet transform and thresholding the wavelet coefficients is a good strategy. Utilizing the threshold we may then remove (hard thresholding) or shrink toward zero (soft thresholding) wavelet coefficients at each level of the decomposition in an attempt to eliminate the noise from the signal. Inverting the wavelet transform yields an estimate of the underlying denoised signal \( f(t) \). Thresholding wavelet coefficients are appealing since they capture information at different combination of time and frequency. Thus the wavelet based estimate is locally adaptive.

Figures 4.24 to 4.27 display the result of universal thresholding applied to wavelet coefficients of the four return series. In each figure the upper row is the plot of the respective return series and next two rows represent the estimates of wavelet coefficients after applying hard and soft thresholding respectively. Here we have used universal threshold of Donoho and Johnostone (1994) by setting

\[
t = \sigma \sqrt{2 \log N}.
\]

As we know hard thresholding affects wavelet input coefficients that are less than and equal to the threshold \( (t) \) i.e., make them as zero. Whereas the soft thresholding, instead of forcing wavelet coefficient to zero or leaving it untouched, pushes all coefficients toward zero. If the wavelet coefficient happens to be smaller in magnitude than the threshold, then it is set to zero.

Figure 4.24 shows the results of application of both hard and soft thresholding of the Sensex return series. The universal thresholding value in this case is found to be 1.9540. Therefore large spikes greater than 1.9540 are left untouched and values smaller than and equal to 1.9540 are set as zeros in case of first scale Haar wavelet coefficients by hard thresholding. Whereas soft thresholding suppresses the larger spikes towards zero, that is why it gives
relatively small spikes after applying the same. As a whole, after applying both the thresholding rules the Sensex return series retains all its spikes with a smooth plot of it. The universal threshold values for National Index, S&P CNX Nifty and S&P CNX 500 returns series are 1.74499, 1.8825, and 1.6879 respectively. The interpretations of the figures from 4.25 to 4.27 are same as Sensex return series.

Towards the end we have gone for estimating the summary statistics for original returns series and reconstructed returns series by wavelet coefficients for comparison, which are reported in table 4.1. The most interesting findings are that mean and skewness of reconstructed returns are zero. Since all the return series are used on the assumption of the zero mean, the wavelet reconstructed return series are particular answer to that. Another important thing is that standard deviations of the reconstructed series are less than original return series and skewness is zero for all the reconstructed series. Thus, they are relatively more approximated towards the normality assumption of the return series. These informations are very much useful while using return series for further statistical estimation or application.

4.10 Concluding Remarks

In this Chapter, the four returns series are subjected to discrete wavelet transform (DWT) by using Haar, Daubechies - 4 (D4) and Daubechies - 6 (D6). The results in this chapter have been an exploratory investigation into the applicability and usefulness of the wavelet analysis to detect element of fluctuations at various scales, and recover signals from noisy observations (also known as wavelet de-noising or wavelet shrinkage). It is apparent that wavelet analysis of return empirically explores the fluctuations, removes patchiness and looks for patterns possibly at certain levels. It also recovers signals from the noisy data by applying universal thresholding rule. In general, wavelet coefficients are very much useful for the statistical analysis of stock return series.
<table>
<thead>
<tr>
<th></th>
<th>Sensex</th>
<th>National</th>
<th>CNX Nifty</th>
<th>CNX 500</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RET</td>
<td>RET_d1</td>
<td>RET</td>
<td>RET_d1</td>
</tr>
<tr>
<td>Mean</td>
<td>0.027</td>
<td>0.000</td>
<td>0.027</td>
<td>0.000</td>
</tr>
<tr>
<td>SD</td>
<td>0.842</td>
<td>0.559</td>
<td>0.791</td>
<td>0.499</td>
</tr>
<tr>
<td>Sk</td>
<td>0.146</td>
<td>0.000</td>
<td>0.209</td>
<td>0.000</td>
</tr>
<tr>
<td>J-B</td>
<td>1229.3</td>
<td>1515.3</td>
<td>365807</td>
<td>1712.0</td>
</tr>
<tr>
<td>p-value</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Note: RET represents original return series and RET_d1 represents return series reconstructed from wavelet coefficients. Jarque-Bera (J-B) statistics is approximately distributed as central Chi-square (2) under the null hypothesis of normality.
Figure 4.1: Pyramid Algorithm (Down Sampling)

Figure 4.2: Pyramid Algorithm (Up Sampling)
Figures 4.3-4.6: Plots of Daily Closing Values of the Indices

Figure 4.3: Sensex

Figure 4.4: National Index

Figure 4.5: S&P CNX Nifty

Figure 4.6: S&P CNX 500
Figure 4.7: Wavelet Decompositions of a Dyadic Length Sensex Return Series (Haar)
Figure 4.8: Wavelet Decompositions of Sensex Return Series (Haar)

Figure 4.9: Wavelet Decompositions of National Index Return Series (Haar)
Figure 4.10: Wavelet Decompositions of S & P CNX Nifty Return Series (Haar)

Figure 4.11: Wavelet Decompositions of S & P CNX 500 Return Series (Haar)
Figure 4.12: Reconstruction of Sensex Return Series from Haar Wavelet Coefficients

Figure 4.13: Reconstruction of National Index Return Series from Haar Wavelet Coefficients
Figure 4.14: Reconstruction of S & P CNX Nifty Return Series from Haar Wavelet Coefficients

Figure 4.15: Reconstruction of S & P CNX 500 Return Series from Haar Wavelet Coefficients
Figure 4.16: Wavelet Decompositions of Sensex Return Series (db4)

Figure 4.17: Wavelet Decompositions of National Index Return Series (db4)
Figure 4.18: Wavelet Decompositions of S & P CNX Nifty Return Series (db4)

Figure 4.19: Wavelet Decompositions of S & P CNX 500 Return Series (db4)
Figure 4.20: Wavelet Decompositions of Sensex Return Series (db6)

Figure 4.21: Wavelet Decompositions of National Index Return Series (db6)
Figure 4.22: Wavelet Decompositions of S & P CNX Nifty Return Series (db6)

Figure 4.23: Wavelet Decompositions of S & P CNX 500 Return Series (db6)
Figure 4.24: Universal Thresholding to Wavelet Coefficients of Sensex Return Series

Figure 4.25: Universal Thresholding to Wavelet Coefficients of National Index Return Series
Figure 4.26: Universal Thresholding to Wavelet Coefficients of
S & P CNX Nifty Return Series

Figure 4.27: Universal Thresholding to Wavelet Coefficients of
S & P CNX 500 Return Series