Data Compression Techniques for Stock Market Prediction

Salman Azhar, Greg J. Badros, Arman Glodjo
Ming-Yang Kao and John H. Reif[‡]

Abstract

This paper presents advanced data compression techniques for predicting stock markets behavior under widely accepted market models in finance. Our techniques are applicable to technical analysis, portfolio theory, and nonlinear market models. We find that lossy and lossless compression techniques are well suited for predicting stock prices as well as market modes such as strong trends and major adjustments. We also present novel applications of multispectral compression techniques to portfolio theory, correlation of similar stocks, effects of interest rates, transaction costs and taxes.

1 Stock market models

In this paper we address the applicability of modern data compression techniques for predicting future behavior of stock markets. Motivated by stock market theory, we find that most compression techniques are well suited for successful prediction. We introduce different market models and present compression techniques that work well under these models. A stock market provides a framework for investors to allocate funds into stocks, and to try to make profits by buying "under valued" stocks and selling "over valued" stocks. Stock markets are one of the most complex and rewarding systems to model accurately. Since their incorporation in the latter part of the last century, there have been a vast number of different techniques to predict their future behavior. Based on the market models we present in this section, we introduce several data compression based techniques for their prediction. We first present definitions of classical views 4f the markets, namely, the technical and the quantitative in Sections 1.1 and 1.2, respectively. In Section 1.3 we introduce a modern market model that admits non-efficiency. We analyze the applicability of data compression techniques under these different models. Lastly, in Section 1.4, we introduce the portfolio theory and implications thereof to data compression algorithms.

1.1 Technical analysis of stock markets

Stock market prices, as most other economic time series, have been modeled by examining how new information impacts the prices, and how much these prices reflect the true value of the underlying security. The most common market prediction techniques can be divided into two main categories: technical and quantitative. The technical view of the markets is that the prices are driven by investor sentiment and that the underlying sequence of prices can be captured and predicted well using data charting techniques [31, 30]. This method studies the action of the market as a *dynamical entity*, rather than studying the actual goods in which the market

[‡]Azhar is with Computer Science Dept., North Carolina A& T State University, Greensboro. All others are with Department of Computer Science, Duke University, Durham, NC. Reif is supported by DARPA/ISTO Grant N00014-91-J-1985, Subcontract KI-92-01-0182 of DARPA/ISTO prime Contract N00014-92-C-0182, NSF Grant NSF-IRI-91-00681, NASA subcontract 550-63 of prime Contract NAS5-30428, and US-Israel Binational NSF Grant 88-00282/2., Kao is in part supported by NSF grant CCR-9101385.

operates. This is a science of recording the historical market data, such as prices of stocks and the volume traded, and attempting to predict the future from the past performance of the stock. This model implicitly admits a non-efficient market model, where the prices are not only a function of the underlying security valuation, but are also governed by investor sentiment, health of the economy and many other. This is a popular and highly contested market model. Terms like "primary", "secondary" and "minor" trends, "tides", "ripples", "head-and-shoulders top" patterns, and "triangles" were coined in the technical community by repeated observations of stock trends and their classification into more common patterns. Data compression techniques are therefore a perfect paradigm for algorithmic charting techniques. In Section 2 we present data compression techniques suitable for technical analysts.

1.2 The efficient markets hypothesis

No less controversial is the model of an efficient market. *Market efficiency* essentially states that the markets are priced in such a way that *all* public information available is already included in the price of a stock, and movements on the markets are only due to arrivals of new information [13, 14, 30]. Large numbers of investors insure that the prices on the market are fair, and always converge towards an equilibrium. The price history of the market does not affect the future, and consequently, the prices follow a random walk. Efficiency in capital markets implies that the time series of returns are identically and independently distributed (IID) random variables, and consequently, no investor can consistently beat that market. Hence, no predictive technique will be able to consistently outperform the buy-and-hold strategy. To quantify the above, we start by interpreting stock prices as a time series of data. Let P_t be the price of a particular stock at time t, where t can be measured in units of seconds, minutes, days, months or years. Under the efficient market hypothesis, the time series of prices P_t are a random walk, and the returns, $\frac{P_t}{P_{t-1}}$, of the stock are normally distributed. Under the efficiency hypothesis, no predictive technique is expected to consistently outperform the market. There is, however, considerable quantitative evidence that points to markets being non-efficient. Section 1.3 describes this evidence and defines the "nonlinear market model" used in this paper. We present data compression techniques suitable for this view of the markets in Section 2.4.

1.3 Nonlinear market models

A comprehensive study of stock daily returns [12] found that returns were negatively skewed, there were more observations in the left side of the tail, the tails were fatter than a Gaussian and the peak around the mean was higher. The distribution of returns was consistently behaved in this manner, pointing to a non-I.I.D. nature of returns. This condition was coined *leptokurtosis.* A novel and widely respected statistical technique, the Brock-Dechert-Sheinkman (BDS) statistic is one of many that rejects the null hypothesis of the IID nature of returns [32, 22, 37]. In other words, the markets have been analytically found to be inefficient, something that has empirically been obvious for some time(in view of the fact that considerable profit has been made from trades, such as arbitrage, exploiting market inefficiency). This degree of inefficiency is crucial to investors who compare dynamic "buy-sell" strategies (which are overwhelmed by transaction costs) to "buy-and-hold" strategies (which depend on accurate forecasting of long term trends), and off-set their risks by diversifying their investments. More on this in Section 2.3. The financial academic community(as opposed to the market community, e.g. Wall Street) has not, as of yet, completely accepted the failure of the EMH, but has largely agreed that the present investment techniques are in real danger of becoming obsolete [30]. Vaga and Paters [30, 35] postulate that the markets are multimodal, and that they continuously drift between the following four modes: (1)*Efficient Mode*, where current prices reflect all "available information" with history of prices and returns; (2)*Transition Mode*, where the level of coherent investor behavior increases due to new information and/or other factors, hence resulting in the impact of this information to extend for prolonged periods of time; (3)*Chaotic Mode*, where the investor sentiment is very strong but variations in groupthink are increased as well, which for example may arize before the release of company quarterly reports or shortly before election results; (4)*Coherent Mode*, where investors act as more or less one unit, resulting in very strong positive or negative market trends. So far this has been found to be a highly realistic view of the actual operation of the markets [30]. We introduce this model of the stock market. In Section 3.2 we present data compression techniques that are suitable for prediction of nonlinear market models under fixed time series memory. Section 3.4 addresses data compression techniques for dynamic memory models as defined below. Mandelbrot [26, 27] postulated and Lo [25] and Peters [30] have empirically shown that the distribution of returns on the stock market follows a fractal distribution, known as stable *Levy distribution.* These distributions have fatter tails and are peaked higher at the mean than the Gaussian, and their underlying time series can experience trends, cycles as well as catastrophes. The characteristic function $f(t)$ of a random variable, t, for the Levy distribution is given by $\log(f(t)) = i\delta t - \gamma|t|^{\alpha}(1+\beta i(\frac{t}{|t|})\tan(\alpha\frac{\pi}{2}))$. This distribution has four characteristic parameters that are continuously changing: δ is the location parameter of the mean; γ a parameter to adjust for different granularity of data; β is a measure of skewness to the left or the right, valued from -1 to 1; α is a measure of the leptokurtosis of the distribution. For $\alpha = 2$, the distribution generalizes to the normal distribution. For example, if $\alpha = 2, \beta = 0, \gamma = 1, \delta = \mu$, we have $\log(f(t)) = i\mu t - (\frac{\sigma^2}{2})t^2$, the standard normal distribution $N(\mu, \sigma^2)$. The *nonlinear market hypothesis* [30] states that α can vary from 1 to 2. Noninteger values of α result in fractional Brownian motions of prices, which An important implication of this model is that a stock time series has a memory. Future prices are a function of new information as well as the past performance of the stock. For example, investors might be fairly reserved to invest in a company that is in financial crisis, even though the stock price is low. Estimation of the memory window can be tuned by adjusting the parameters of the underlying distribution. As the parameters of the Levy distribution are varying in time, in accordance with Vaga's multimodal model, the stocks have a continuously changing memory. The dynamic memory model of the stock market presents an intriguing problem for data compression. For example, the calculation of the memory for a particular stock is still an open question. But, following the works of Mandelbrot [27] described by Peters [30] and interpreting stock market time series as statistically self-similar fractals and using the tools of fractal geometry, one can get a very good approximation of the extent of current memory and hence, can form an appropriate window for our compression algorithm. We present this compression model in Section 3.4.

1.4 Modern portfolio theory

The efficient markets hypothesis was a fundamental driving force for development of the *modern portfolio theory.* Modern portfolio theory provides the toold for a quantitative analysis of risky assets such as stocks, based on the expected return, which under the EMH is the mean expected value, and the risk, under EMH defined as the variance of a portfolio. A portfolio is a vector-valued time series of stocks $X = \{X_i | i = 1, ..., n\}$ where n is the number of stocks and X_i is the stock "price relative" defined as $X_i = \frac{P_{it}}{P_{it-1}}$. For example, $X_5 = 1.05$ would imply that the fifth stock in our portfolio went up by 5 percent in one day. In this section we present the investment theory from the efficient hypothesis point of view, which assumes that X_i's independent and identically distributed random variables and that the time series of X's is a stationary ergodic vector-valued stochastic process. We adopt the notion of the distribution of X as normal, for the time being. As a part of our future work, we will re-examine this hypothesis and present portfolio allocation techniques under the non-efficient model. Define a portfolio as a vector representing the allocation of wealth over a collection of stocks: $b = (b_1, b_2, b_3, ..., b_m)$. Therefore, the ratio of wealth from start to the end of a trading day can be defined as: $\mathcal{S} = b^t X$. $\mathcal{S}$, itself is a random variable. As with most things associated with the stock markets, the distribution of $\mathcal{S}$ is also subject of a lot of controversy. The Sharpe-Markowitz [7] of stock market tries to maximize the expected value of S while minimizing its variance. In this model, one examines the so-called efficient frontier, a region in the variance-mean space, while assuming that there is a one-to-one correspondence between a true value of a stock and its variance. This is the so-called Capital Assets Pricing Model (CAPM). In Section 2.3 we present data compression techniques applicable for optimal portfolio allocation problems under the CAPM model.

2 Market prediction via data compression

Data compression (DC) is the task of efficiently encoding a data source to derive a more compact data representation by removing redundancy, while preserving the essential information in a retrievable format. DC explicitly assumes that the input has an a priori probability distribution that is exploited. DC algorithms have two distinct functions: encoding the source data and decoding the encoded data. An encoding phase of DC algorithms performs two distinct tasks: (1) learning the distribution of the source; (2) assigning codes based on this learning. These two tasks are usually employed in tandem: the learning phase generates a dictionary that is used to encode the source. The decoding phase reverses the dictionary look-up to recover the original source data (or a close approximation thereof). DC algorithms can be used as predictive schemes using the characteristics of the dictionary to predict the frequency of the source. This observed relationship between DC and prediction has been extensively used in [15, 9, 10, 8, 5, 16, 17, 36, 11], particularly for stationary sources. In this section, we examine basic issues in prediction for stationary sources. Later, we do present several new results for dynamic schemes, specifically in relation with the stock market. Cover and Thomas [6] provide a theoretical background to information complexity issues establishing a direct relationship between high compressibility and high predictability. Vitter and Krishnan [36, 11] prove this result for some particular lossless DC based prediction algorithms. Since the readers are likely to be experts in DC, our discussion of DC is brief. We shall note prediction can be done by various known DC techniques, all in the presence of a stationary source. DC schemes can be lossy or lossless. Lossy DC allows the algorithm to permanently lose non-essential information to achieve higher compression ratios, whereas lossless DC schemes insist that the decoded data should be identical to the source data. Lossy DC techniques can be used to represent a source by mapping each source item to one of finite number of dictionary elements that minimizes distortion under a suitable metric. Quantization is a fundamental example of lossy compression. Scalar Quantization (SQ) observes a single source datum and selects the nearest approximation value from a predetermined finite set of allowed numerical values [1]. Vector Quantization (VQ) [18] is a generalization of SQ to quantization of a vector, a point in n-dimensional Euclidean or any other metric space, according to some vector space metric such as the metric induced by the l^2 norm. VQ can be used for predicting by filtering primary components to maintain correlations among different values. Predictive VQ (PVQ) schemes were introduced by Fischer an Tinner [15] as well as Cuperman and Gersho [9]. Extensive studies were performed by Cuperman and Gersho [10, 8], Chang and

Gray [5], and Fischer and Tinnen [16, 17]. Lossless DC can be achieved by assigning shorter codes to frequently occurring source sequence, and vice versa. Lempel-Ziv algorithm [41, 34, 3] is an example of this technique. Analogously, prediction can be achieved by assigning higher probability to more frequently occurring sequences. Vitter and Krishnan [36, 11] use data compression based prediction to devise prefetching algorithms for theoretically optimal caching strategy. We present algorithms that use lossy and lossless schemes to predict movements on the stock market. Our algorithms are applicable for optimal portfolio allocation, technical analysis (charting) and other more advanced non-efficient market models. [33, 40, 4].

2.1 Vector quantization (VQ) based prediction

Vector quantization is a lossy DC technique is perfectly suited for market prediction under the technical and other nonlinear models. For example, lossy compression techniques will map a vector of a stock history to its representative under some desired metric, to form a collection of equivalence classes of market moves, and will reduce prediction complexity by orders of magnitude. All charting techniques can be viewed as implicit lossy compression schemes. VQ can be achieved by representing all source data using a dictionary consisting of representative vectors smaller in size and/or number than the original source. VQ maps an input vector to its closest neighbor in the dictionary based on a given distortion measure, and represents the input vector by the index to the dictionary [18]. We represent stock market time series data as a two dimensional space-time vector, where the rows of the vector contain, for example the securities' prices at the time indexed by the columns. Given a partial space-time vector, we can match it to the nearest space-time vector using an appropriate distortion measure. The matching vector is said to predict the unknown components as specified by it. *Tree Structured* VQ (TSVQ) is a variant of VQ that can reduce the time complexity of VQ from linear to logarithmic [19]. A TSVQ dictionary is implemented as a d-ary tree whose intermediate nodes represent the centroids of their children. TSVQ is most suited for our VQ-based prediction scheme for stock market indices for technical or other non-efficient market models. We therefore implement this variant of VQ for our empirical tests.

2.2 Lempel-Ziv based prediction

Lossless data compression can be regarded as a predictive algorithm applied to a set of quantized or raw stock market data. It is well-suited for optimal allocation of funds as given by CAPM in stock market portfolio theory. Lossless DC can be achieved by assigning shorter codes to frequently occurring source sequence, and vice versa. Lempel-Ziv (LZ) algorithm [41, 34, 3] is an example of this technique. LZ satisfies the unique parsing criteria: no two phrases in the parse of the source are identical. Cover and Thomas [6] show that LZ's unique parsing property makes it optimal for sources with stationary ergodic memoryless distributions. Vitter and Krishnan [36, 11] extend this result by proving that optimal compression algorithms can be converted in optimal prediction algorithms assuming stationary ergodic distribution.

2.3 Multispectral prediction

Multispectral compression (MC) algorithms exploit the high degree of redundancy among spectral bands to compress a multispectral image [29]. This spectral redundancy is similar to the redundancies among stock market indices: only limited portion of spatial information changes among bands. For example, stock indices from geographically close areas, stocks of similar companies, interest rates and housing stocks are pairwise related to each other. Lossy MC is a well-researched area [29, 20, 2]. The encoding phase of MC consists of the following steps: (1) Histogram equalization, (2) Spectral transform (such as cosine or more advanced techniques, e.g. wavelet transform), (3) Scalar or Vector Quantizer and (4) Lossless compressor such as LZ [41]. The decoding process reverses each one of the above steps in complementary order. The ***histogram equalization*** is a non-linear process that equalizes the probability distribution function (PDF) $p(t)$ of a random variable t with PDF $p_d(s)$ of a desired random variable s. It is accomplished by identifying a monotone non-decreasing map $s = T[t]$ that equalizes the cumulative PDFs of the two images [23], and identifying the image that needs to be modified. $P(t) = \sum_i^t p(i)$ is equalized to $P_d(t) = \sum_i^s p_d(i)$. This can be viewed as SQ mapping one set of quantized spectra into another using a table to specify the mapping. Prediction is achieved by breaking the source into blocks of different spectra and encoding it as a juxtaposed vector. MC affords an impressive treatment of capital markets where several indices, returns, averages, interest rates, etc. show strong correlation among one another. In fact, Fama [13] shows that returns from different stocks exhibit up to 97% correlation with each other. Consequently, we expect MC to yield high compression of various data associated with the stock markets such as indices, returns, interest rates, etc. This enables us to exploit more information and combine it efficiently. We are investigating the application of wavelet techniques and shall report our results in near future.

2.4 Distortion controlled prediction

Our prediction algorithms can also be used to find a particular mode of the stock market. Vaga and Peters [35, 30] describe a market to be in one of the following mode: efficient, transition, chaotic and coherent, as defined in Section 1.3. By looking at the time-series of the changing parameters of Levy distributed returns, we can predict new characteristic parameters of the distribution. Consequently, by looking at the α, we can see how close our series is to efficiency, and we can obtain precise measurements of predictive powers of our compression techniques. We can therefore quantitatively measure the expected accuracy of compression algorithms for prediction. Accordingly, we need different levels of granularity to preserve distortion. Traditionally lossy DC algorithms have focused on compressing data at fixed bit rates. Markas and Reif [28] present Distortion Controlled VQ (DCVQ) algorithms that comply with specified distortion measures. When the market is in the efficient and chaotic mode, the information loss is more tolerable. Hence, we used DCVQ algorithms [28] to relax the distortion control to smooth the insignificant changes. On the other hand, when the market is in transition or coherent mode, we can increase distortion control to some extent.

3 Prediction model

This section describes the development of prediction tables for stock market trends, so that knowledge the past trends of a stock can allow us to estimate the prediction probabilities of the stock's future performance. This is a memory-based model, and this type of models offers significant advantages when compared with memoryless techniques. The process of generating a model for predicting trends in stocks can divided into three phases: the generation of prediction tables, the prediction of trends based on the known information, and finally adaptation of the model based on new data.

3.1 Generation of prediction tables

During the generation step, data consisting of either a collection of correlated stocks and/or events that affect those classes of stocks are collected in a form of a discrete vector of fixed length, which we call a *stock-event*. A large selection of such vectors will be used to generate the VQ codebook vocabulary: each entry represents a sequence of events that indicate the past performance of a stock. Training can be accomplished based on one of the known VQ training algorithms (such as LBG method [24]). Since the traditional VQ algorithm is a memoryless technique, some memory that keeps track of sequences of events, as well as their transitions probabilities among different events must be implemented within the model. Given a particular window of time, we calculate the current probability distribution of returns, and quantize it into a dictionary of past distributions. For our analysis, we consider returns on the stock market to be stable-Levy. This two-stage quantization is necessary for prediction because the first stage will capture the deterministic components by deterministically weighing the past, and the VQ stage will codify the present distribution to a nearest distribution under the same conditions in a statistical sense.

3.2 Prediction

The prediction process can be seen as a predictive vector quantization(PVQ) algorithm [21]. PVQ consists of a series of discretization and matching operations that define the current state, and a prediction process that searches the prediction tree and identifies the probability tables for all possible transitions. The current state of a stock trend is defined as the sequence of the N past stock-events. The stock data sources that form an event are discretized at N-past time intervals, and a VQ process is initiated to identify their best-match vector from the existing codebook. The string that is generated during this process represents the current state of the stock. The complexity of the VQ algorithm is linear with respect to the size of the codebook. This step can become computationally prohibitive for large vocabularies, likely to be generated in stock market applications. The Tree-Structured Vector Quantization (TSVQ) [19] reduces the complexity of the VQ method from linear to logarithmic time. In TSVQ, the vocabulary is implemented in a d-ary tree form, whose intermediate nodes represent the centroids of their children nodes and the leaf nodes are the actual codewords. Once the current state has been computed, a sequence of events that exists in the prediction tree and matches with the current state needs to be identified. During this process, we search the past performance of a class of stocks to identify a sequence of trends that resemble the current performance of the stock. If such sequence is found, and if there exists enough statistical information concerning the possible trends of the stock, then we can predict how a stock is likely to behave in the next few periods. The identification of the current state is accomplished by identifying the maximum-length sequence of events that exists in the prediction tree. To achieve this, a string of past events is formed and search in the prediction table is initiated to identify a match for the current string. If a match is found, a new event is added to the sequence ($N = N + 1$) and a new matching process is initiated. If there is no match, the previous sequence is used as the current state. The matching process can be implemented efficiently using hashing tables similarly to the textual substitution [34, 38] algorithms used for data compression purposes of textual data. Once the maximum string has been identified, the prediction table for the future values of stock is predicted similar to the encoding process of arithmetic coders [39].

3.3 Adaptation

In this step, we use current information regarding the performance of a stock and prediction errors of our model dynamically adapt the prediction tree. This will improve the performance of our model. The adaptation part consists of modification of the transition tables so that the model learns new information: creation of new tree nodes in case that observed transitions do not exist in the current tables, and creation of new events once the distance between an observed event and its best-match vector is larger than a predefined threshold. This last step is similar to the splitting step that takes place during the creation of VQ codebooks for image or speech data. Once a vector (stock-event) whose distance from the best-match exceeds a certain threshold has been found, a new entry is created in the prediction tree. If a collection of vectors X_i, had B as the best-match vector, and a new vector Y has been encountered that has B as its best-match vector, but its distance from B exceeds a certain threshold, a new codevector B' is created. The X_i vectors will then be reassigned to either B or B' so that the quantization error for all X_i is minimized based on a given measure. It should be noted that a fair amount of computation is involved in this step since a large portion of the prediction tree will have to be readjusted.

4 Empirical results for Stock market prediction

4.1 Introduction

As noted earlier, data compression techniques are a perfect paradigm for stock market prediction. For our empirical tests we have adopted a notion of markets being multimodal [35], the prices obeying fractional Brownian motion and the returns distributed in accordance with stable-Levy distribution. This approach embodies the state of the art in nonlinear, chaotic, stock market models. It also presents some interesting compression issues. For example, we develop a dynamic memory VQ technique in order to quantize a vector that may have a variable window size. This window size is calculated by regarding the markets and statistically self-similar fractals and calculating its fractional dimension using the R/S statistic of Mandelbrot [30]. The R/S statistic adequately captures the long range memory of stock markets, and we can use this estimate to truncate the window appropriately. It is important to be able to precisely calculate this window, because if we quantized a price-time stock market vector that contained components that were not influencing the present, or for that matter, the future prices, our prediction and compression would be inaccurate. Once a memory window has been calculated, using lossy TVQ we map the price-time vector to its proper equivalence class, under a given market mode. From this data we can by the means of lossless LZ algorithm predict the next highest probable market mode, price of a stock or a group of stocks or any other parameter we are interested in. Preliminary experiments using only lossless techniques indicate a small, but significant degree of prediction.

4.2 Lossless compression - results

Below we present results of Lempel-Ziv for predicting rises and falls of all major world indices. We consider logarithmic returns, and apply uniform scalar quantization of the entries. We then run the LZ algorithm, and predict a rise or a fall from day to day returns. Each time we guess correctly we score a +1, and each time we guess incorrectly we score nothing. If we are at a leaf, we do not guess. The results, continent-wise, are tabulated below. Our lossless prediction scheme yields an average success of 55.06 percent, which we find very encouraging. It should be noted, however, that our results include periods where markets are stable(i.e. no price changes), and hence would in return yield higher compression ratios. These, although very few, instances do not significantly affect the results, and are discounted in our future work with lossy compression.

Simulation results - U.S. and Canada	
Market	Percent Correct
DowJones	51.10
Snp500	51.42
Toronto	54.83
Average	52.45

Europe	
Market	% Correct
Denmark	58.81
Germany	52.34
Finland	61.85
Netherla	51.48
Norway	56.07
Paris	57.11
Portugal	61.28
Spain	62.63
Sweden	55.80
Swtzrlnd	51.93
United Kingdom	52.85
Average	56.56

Asia and Africa	
Market	% Correct
Hongkong	50.95
Indonesia	60.03
Korea	53.98
Malasyia	52.33
Phillipines	53.97
Singapore	53.21
Taiwan	54.93
Thailand	52.58
Tokyo	51.93
South Africa	58.09
Average	54.20

References

[1] Gersho A and R. M. Gray. *Vector Quantization and Signal Compression.* Kluwer Academic Publishers, Norwell, Massachusetts, 1992.

[2] R. L. Baker and Y. T. Tse. Compression of high spectral resolution imagery. In *Proc. of SPIE: Applications of Digital Image Processing*, pages 255–264, August 1988.

[3] T. C. Bell, J. C. Cleary, and I. H. Witten. *Text Compression.* Prentice-Hall, 1990.

[4] Fischer Black, Michael Jensen, and Myron Scholes. The capital assets pricing model: some empirical tests. In Jenson, editor, *Jenson (ed.)*, pages 79–121, 1972.

[5] P. C. Chang and R. M. Gray. Gradient algorithms for designing predictive vector quantizers. *IEEE Trans. Accoust. Speech signal Processing*, ASSP-34:679–690, August 1986.

[6] T. M. Cover and J. A. Thomas. *Extensive Games and the Problem of Information.* John Wiley, 1991.

[7] T.M. Cover and J.A. Thomas. *Elements of Information Theory.* John Wiley and Sons, New York, NY, 1991.

[8] V. Cuperman. *Efficient waveform coding of speech using vector quantization.* PhD thesis, University of California, Santa Barbara, February 1983.

[9] V. Cuperman and A. Gersho. Adaptive differential vector coding of speech. In *Proceedings of GlobeCom*, pages 1092–1096, December 1982.

[10] V. Cuperman and A. Gersho. Vector predictive coding of speech at 16 kb/s. *IEEE trans. Comm.*, COM-33:685–696, 1985.

[11] K.M. Curewitz, P. Krishnan, and J.S. Vitter. Practical prefetching via data compression. *SIGMOD*, 1993.

[12] E.F. Fama. The behavior of stock market prices. *Journal of Business*, 38, 1965.

[13] E.F. Fama. *Foundations of Finance.* Basic Books, Inc., New York, New York, 1976.

[14] Eugene F. Fama. Efficient capital markets II. *Journal of Finance*, 46(5), June 1991.

[15] T. R. Fischer and D. J. Tinnen. Quantized control with differential pulse code modulation. In *Conference Proceedings: 21st Conference on Decision and Control*, pages 1222–1227, December 1982.

[16] T. R. Fischer and D. J. Tinnen. Quantized control using differential encoding. *Optimal Control Applications and Methods*, pages 69–83, 1984.

[17] T. R. Fischer and D. J. Tinnen. Quantized control with data compression constraints. *Optimal Control Applications and Methods*, 5:39–55, January 1984.

[18] R. M. Gray. Vector Quantization. *IEEE ASSP Magazine*, 1:4–29, April 1984.

[19] R. M. Gray and H. Abut. Full search and tree searched vector quantization of speech waveforms. *Proc. ICASSP*, pages 593–596, May 1982.

[20] S. Gupta and A. Gersho. Feature predictive vector quantization of multipsectral images. *Trans. on Geoscience & remote sensing*, 30(3):491–501, May 1992.

[21] H. Hang and J. W. Woods. Predictive vector quantization of images. *IEEE Trans. Acous. Speech and Sig. Proc.*, 33, Nov. 1985.

[22] D.A. Hsieh. Chaos and nonlinear dynamics: Application of financial markets. *J. Finance*, 46(5), 1991.

[23] J. S. Lim. *Two Dimensional Image and Signal Processing.* Prentice-Hall, 1990.

[24] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantizer design. *IEEE Trans. on Commun.*, 28:84–95, Jan. 1980.

[25] A.W. Lo and C. MacKinlay. Stock prices do not follow random walks: Evidence form a simple specification test. *Rev. Finan. Stud.*, 1(1), 1988.

[26] B. Mandelbrot. The pareto-levy law and the distribution of income. *Inter. Nat. Econ. Rev.*, 1, 1960.

[27] B. Mandelbrot. Statistical methodology for non-periodic cycles. *Ann. Econ. Soc. Measr.*, 1, 1972.

[28] T. Markas and J. Reif. Image compression methods with distortion control capabilities. In *Proceedings of 1st Annual Data Compression Conference*, pages 121–130, April 1991.

[29] T. Markas and J. Reif. Mutispectral image compression algorithms. In *Proceedings of 3rd Annual Data Compression Conference*, pages 391–400, April 1993.

[30] E.E. Peters. *Chaos and Order in the Capital Markets.* John Wiley & Sons, 1991.

[31] J. Magee R.D. Edwards. *Technical Analysis of Stock Trends.* John Magee Inc., Boston, USA, 1992.

[32] Jose A. Scheinakman. Nonlinearities in economic dynamics. Technical report, Working paper, SFI Economics Research Program, 1989.

[33] William F. Sharpe. Capital asset prices: A theory of market equilibrium under conditions of risk. *Journal of Finance*, pages 425–442, September 1964.

[34] J. A. Storer. *Data Compression: Methods and Theory.* Computer Science Press, Rockville, MD, 1988.

[35] T. Vaga. The coherent market hypothesis. *Financial Analysts Jounral*, December/January 1991.

[36] J. S. Vitter and P. Krishnan. Optimal prefetching via data compression. In *Proceedings of 32nd Annual IEEE Symposium on FOCS*, pages 121–130, October 1991.

[37] B. LeBaron W.A. Brock, D.A. Hsieh. *Nonlinear Dynamics, Chaos, and Instability.* MIT Press, Boston, USA, 1991.

[38] T. A. Welch. A technique for high-performance data compression. *Commun. of ACM*, pages 8–19, June 1984.

[39] I. H. Witten, R. M. Neal, and J. G. Cleary. Arithmetic coding for data compression. *Commun. of ACM*, pages 520–540, June 1987.

[40] Holbrook Working. Capital asset prices: A theory of market equilibrium under conditions of risk. *American Economics Review*, 48:188–199, 1958.

[41] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions of Information Theory*, 24:530–536, September 1978.