Imperial College London

Portfolio Optimization with Profit Prediction

Author:
Rousland Dimitrov
Supervisor:
Daniel Kuhn

Department of Computing

Imperial College London, London, UK

A thesis submitted for the degree of

Master of Science

August 27, 2010

1. Reviewer: Daniel Kuhn

2. Reviewer: Berç Rustem

Day of the defense: August 24, 2010

Abstract

Portfolio optimization is a process that allocates resources to an investment based on the investor goals and the history of a market. The prices of assets are typically modeled as a stochastic process and their future values are thus unknown when the investment is made. This uncertainty introduces the notion of risk and is an important factor, besides the expected return of any portfolio.

We introduce an intuitive model of financial markets, called the Damping model, which can be used as a framework for price estimation. We then use the model to gain insights into the stochastic processes underlying asset prices and introduce an algorithm that predicts the relative profitability of each asset. We further derive some elementary strategies that accurately predict asset returns and brings high profit to the proposed portfolio with small associated risk. When transaction costs are introduced into the simulation, much of the gain is lost in taxation because all investments not expected to profit are fully sold out, unlike other selection algorithms that keep proportions of assets to hedge against risk.

We conclude with the analysis of until what point our method outperforms other exemplary portfolios on real data.

To human aspiration... Our future is in our own hands.

Acknowledgements

Although this thesis mostly consists of my own and sometimes naïve ideas, it would have not been possible without the help of the people, who raised my interest in Finance and portfolio optimization in particular. A warm thank you to

Dr. Daniel Kuhn for his inspirational course in Computational Finance and his openness to all my numerous questions, suggestions and 'what-if's.

Dr. László Györfi for coming to Imperial College and volunteering to share his ingenious research on portfolio optimization. Our discussion brought me up to speed on the subject and further aroused my interest.

Dr. Gernot Ziegler for accidentally sharing his thoughts on financial simulations and bringing the topic to me for the first time ever.

Rouslan Dimitrov

Contents

List of Figures

1

Introduction

We consider an investment as the allocation of resources in a market of assets. The proportion of wealth invested in each asset forms a portfolio and our aim is to pick a portfolio that achieves high profit with low risk of loss.

At day n, asset m has price $P_{n,m}$ and there are M assets in the market. The daily return is $X_{i+1,m} = P_{i+1,m}/P_{i,m}$ and the whole market can be characterized by the sequence of return vectors $\mathbf{X_i} = (X_{i,1}, X_{i,2}, \ldots, X_{i,M})$.

Our wealth S_n varies daily according to the assets' prices and our investment strategy. Important aspects of any portfolio strategy are the expected financial gain $E[S_n]$ and the associated risk, which is typically a proxy for the standard deviation of the return σ_{S_n}. Investors are risk averse as they prefer the minimal risk for a given gain. This is quantified with the investor's utility $U(x)$, which tells how much he or she values a given return.

Throughout our analysis, experiments are run on real market data[1] of 23 stocks[2] traded on the NYSE for the 4263 trading days from 02/01/1990 to 24/11/2006.

1.1 Asset Prices Growth

In this section we analyze just a single asset and thus drop the m subscript. By definition, $P_{i+1} = X_{i+1}P_i$. The return X_{i+1} is unknown in advance and is thus a random variable. In literature, the return vector in time $X_n = (X_1, X_2, \ldots, X_n)$ is often assumed to be a stationary random process and its components are independent random variables(Lue98). The price of the asset at day n is thus

$$P_n = X_n X_{n-1} \ldots X_1 P_1 \tag{1.1}$$

Taking the natural logarithm of both sides gives:

$$\ln P_n = \sum_{i=1}^{n} \ln X_i + \ln P_1 \tag{1.2}$$

Rearranging and dividing by n

$$\ln \left(\frac{P_n}{P_1} \right)^{\frac{1}{n}} = \frac{1}{n} \sum_{i=1}^{n} \ln X_i$$

[1]Kindly provided by Dr. László Györfi

[2]See Chapter 4.6 for details

Because X_i are independent, for $n \to \infty$, by the law of large numbers

$$\ln\left(\frac{P_n}{P_1}\right)^{\frac{1}{n}} = \mathrm{E}[\ln X_1]$$

Taking the anti-logarithm and raising to the power of n

$$\frac{P_n}{P_1} = e^{n\mathrm{E}[\ln X_1]}$$

$$P_n = P_1 e^{nW} \tag{1.3}$$

Equation 1.3 shows that $W = E[\ln X_1]$ can be considered as the growth rate of the asset price. Somewhat not intuitive, this means that for a single period investment we are interested in maximizing the expected gain and for a multi-period investment - the expected logarithm of the return.

Example 1 *Consider a throw of a fair coin. On heads our bet doubles and on tails it halves.*

$$X = \begin{cases} 2 & heads \\ \frac{1}{2} & tails \end{cases}$$

$$\mathrm{E}[X] = \frac{1}{2}(2 + \frac{1}{2}) = \frac{5}{4} > 1$$

$$W = \mathrm{E}[\ln X] = \frac{1}{2}(\ln 2 + \ln \frac{1}{2}) = 0$$

Therefore playing the experiment once is profitable, but repeating it many times will gain nothing.

1.2 Efficient Markets

An efficient market is one in which the price of an asset reflects the true value of a company (Fam91). It is an idealized model and in its *weak-form* states that future prices cannot be predicted by the analysis of historical data. This claim makes portfolio optimization challenging and brings the need to model asset prices as uncorrelated stochastic processes. We use this hypothesis to gain understanding of the market, but will modify it in a sense that it takes time until information propagates and prices settle.

Chapter 2 shows that some of the most successful portfolio optimization strategies are based on the analysis of past asset returns. The aim of this thesis is to unite these two conflicting statements in a new model.

1.3 Prices as Stochastic Processes

Empirical evidence (Figure 1.1) suggests that the X_i terms in equation 1.1 approximately follow a lognormal distribution. Substituting terms in equation 1.2 as

$$\underbrace{\ln P_{i+1}}_{\mathrm{L}_{i+1}} = \underbrace{\ln X_i}_{\epsilon_i} + \underbrace{\ln P_i}_{L_i}$$

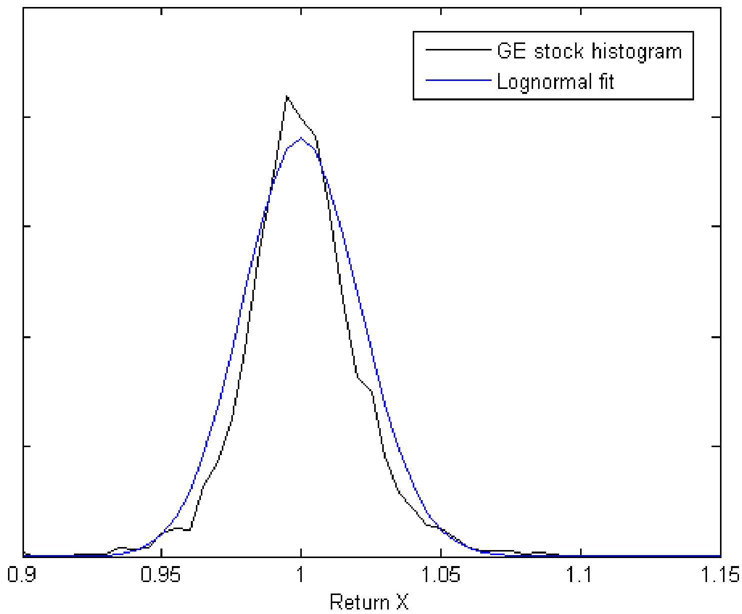

Figure 1.1: Logarithmic Fit - Histogram of the returns of General Electric (GE)

gives

$$L_n = L_1 + \sum_{i=1}^{n} \epsilon_i \tag{1.4}$$

Because $\epsilon_n \sim \mathcal{N}(\nu, \sigma^2)$ are independent according to the efficient market hypothesis

$$L_n \sim \mathcal{N}(L_1 + n\nu, n\sigma^2) \tag{1.5}$$

which tells us that the logarithm of the asset prices grows linearly in time.

Now that we have gained insights into the real asset prices, we need a mathematical model to simulate them. Consider the Wiener process

$$\frac{d\hat{P}}{\hat{P}} = \mu dt + \sigma\hat{\epsilon}\sqrt{dt} \tag{1.6}$$

with $\hat{\epsilon}_n \sim \mathcal{N}(0, 1)$. To bring it to a comparable form with equation 1.5, we take the natural logarithm of both sides. From Itô's lemma (Hul02),

$$d\ln \hat{P}_n = \left(\mu - \frac{\sigma^2}{2}\right) dt + \sigma\hat{\epsilon}\sqrt{dt}$$

Dividing the time-span of the above equation into small intervals of time Δt,

$$\ln \hat{P}_{n+1} - \ln \hat{P}_n = \left(\mu - \frac{\sigma^2}{2}\right) \Delta t + \sigma\hat{\epsilon}_n\sqrt{\Delta t} \tag{1.7}$$

By construction, the above process has the same mean and variance as 1.5. Letting $\hat{L}_n = \ln \hat{P}_n$ and $\nu = \mu - \sigma^2/2$, we get

$$\begin{aligned}
\mathrm{E}[\hat{L}_n] &= \hat{L}_1 + n\nu\Delta t \\
\mathrm{var}[\hat{L}_n] &= \mathrm{E}\left[\textstyle\sum_{i=1}^{n} \sigma\hat{\epsilon}_i\sqrt{\Delta t}\right]^2 \\
&= \sigma^2 \mathrm{E}\left[\textstyle\sum_{i=1}^{n} \hat{\epsilon}_i^2 \Delta t\right] \\
&= n\sigma^2\Delta t
\end{aligned}$$

Exponentiating 1.7,

$$\hat{P}_{n+1} = \hat{P}_n e^{\nu \Delta t + \sigma \hat{\epsilon}_n \sqrt{\Delta t}} \tag{1.8}$$

The last equation is called Geometric Brownian motion and is typically used to model stock prices. It always holds compared to a straight-forward discretization of 1.6, which is true only for $\Delta t \to 0$. We can rewrite it as

$$\hat{P}_{n+1} = \hat{P}_1 e^{\nu n \Delta t} \left(e^{\sigma \sqrt{n \Delta t}} \right)^{\bar{\epsilon}_n} \tag{1.9}$$

The stochastic factor in equation 1.9 consists of a slowly growing term raised to a random number. We will verify the predictive power of the proposed model in this thesis with this equation.

1.4 Portfolio Definitions

A portfolio at day i is characterized by the fraction of wealth invested in each asset. This is expressed as the portfolio vector $\mathbf{b_i} \in B$.

$$B = \left\{ \mathbf{b_i} \in R^m \; : \; b_{i,k} \geq 0, \sum_{k=1}^{m} b_{i,k} = 1 \right\} \tag{1.10}$$

The portfolio gain at i is then $\mathbf{b_i}^t \mathbf{X_i}$ and the total gain in the absence of transaction costs is

$$S_n = \prod_{i=1}^{n} \mathbf{b_i}^t \mathbf{X_i} \tag{1.11}$$

Our aim is to maximize 1.11. Each term in the product can be optimized independently and many authors formulate this greedy strategy in terms of the Bellman's equation (Sch02).

With transaction costs, we introduce the notion of net capital N_i which denotes the amount of wealth available for investment at the start of the i-th trading period. We invest $b_{i,j} N_i$ in each asset j. Prior to the rearrangement of wealth, the same asset holds $b_{i-1,j} S_{i-1}$.

With the above our end of day capital is

$$S_i = N_i \mathbf{b_i}^t \mathbf{X_i} \tag{1.12}$$

with

$$N_i = S_{i-1} - c_t \sum_{j=1}^{m} |b_{i,j} N_i - b_{i-1,j} S_{i-1}| \tag{1.13}$$

where the transaction cost c_t is the commission factor of the exchange authority. Unfortunately, equation 1.12 is hard to maximize because N_i appears in both sides of 1.13 and an optimization process needs to be used.

To gain insights into the source of transaction loses, consider a good strategy which looses little so that $N_i \approx S_{i-1}$. Substituting this into the first term in the modulus in equation 1.13,

$$N_i \approx S_{i-1} - c_t S_{i-1} \sum_{j=1}^{m} |b_{i,j} - b_{i-1,j}|| \tag{1.14}$$

and

$$S_i \approx (1 - c_t \|\mathbf{b_i} - \mathbf{b_{i-1}}\|_1) \, S_{i-1} \mathbf{b_i}^t \mathbf{X_i} \tag{1.15}$$

which shows that when transaction costs are considered, we are penalized for the movement of capital between assets. And although we would like to obtain the maximum for S_n, maximizing one term, might have negative effect on the ones that follow it. That is the reason why the constantly rebalanced portfolio where $\mathbf{b_i} = \mathbf{b_{i-1}}$ is relatively unaffected by transaction costs (See Chapter 4.6).

1.5 Portfolio Benchmarking

Portfolio return is important measure of performance, but investors also base their decisions on the return per unit risk. In this thesis, we adopt Sharpe's ratio(Sha94) for benchmarking various investment strategies. It is defined as:

$$S = \frac{\mathrm{E}[\mathbf{R} - \mathbf{R_f}]}{\mathrm{var}[\mathbf{R} - \mathbf{R_f}]} \tag{1.16}$$

where $\mathbf{R}$ and $\mathbf{R_f}$ are the portfolio and risk-free asset returns, respectively. It measures how well a portfolio compensates an investor for the risk taken and the aim of any strategy is to deliver high returns and high Sharpe ratios.

2

Related Work

In this chapter, we introduce several important investment strategies and give intuitive explanations when and why each one works. We begin with some classic portfolios and a more recent one, based on machine learning. We conclude with a discussion of several models for pricing an asset and estimating its usefulness in a portfolio.

For further reading, see the survey paper (GOU08), which provides more details and larger selection of investment strategies. For clarity, we omit all proofs here and point the reader to the papers that originally introduced the mentioned portfolios.

2.1 Constantly Rebalanced Portfolio

The Constantly Rebalanced Portfolio (CRP) is characterized by an investment vector **b** that is constant in time. The wealth achieved after time n is:

$$\begin{aligned} S_n &= S_{n-1}\mathbf{b}^t\mathbf{X_n} \\ &= S_1 \textstyle\prod_{i=1}^{n} \mathbf{b}^t\mathbf{X_i} \end{aligned}$$

It is important to note that assets perform differently and after each trading period i, the wealth $S_{i+1} = \mathbf{b}^t\mathbf{X_i}S_i$ is rebalanced to $\mathbf{b}S_{i+1}$ for each asset. Because of the constant investment vectors, relatively small amount of trading is performed. As predicted from equation 1.15, in figure 4.8 we will see that the CRP performs well in environments with high transaction costs.

We are interested in the best constantly rebalanced portfolio defined by

$$S_n = \max_{\mathbf{b_n}\in B} S_1 \prod_{i=1}^{n} \mathbf{b_n}^t\mathbf{X_i}$$

We used the subscript n in $\mathbf{b_n}$ to denote that in general, the optimal investment vector changes for different time intervals. Unfortunately, in real financial markets training $\mathbf{b_n}$ for past returns does not perform very well if used for the future. In the Results Chapter 4.6, we used the complete returns and all graphs indeed show the best CRP, which is non-causal and physically unrealizable.

Sometimes $\mathbf{b_n}$ can be computed analytically like in the following simple example.

Example 2 *Extending example 1, consider the option on betting on a coin*

$$X_1 = \begin{cases} 2 & heads \\ \frac{1}{2} & tails \end{cases}$$

or just keeping the money aside

$$X_2 = 1$$

We form a portfolio

$$\mathbf{b} = (b, 1-b)$$

and would like to pick such b as to achieve the highest growth when playing the game repeatedly. From section 1.1, we are interested in maximizing the growth rate W, which is the logarithm of the expected single-period return

$$\begin{aligned} W &= E[\ln \mathbf{b}^t \mathbf{X}] \\ &= \tfrac{1}{2}\ln\left(2b + (1-b)\right) + \tfrac{1}{2}\ln\left(\tfrac{b}{2} + (1-b)\right) \\ &= \tfrac{1}{2}\ln\left((1+b)(1-\tfrac{b}{2})\right) \end{aligned}$$

which is maximal at

$$\mathbf{b} = \left(\frac{1}{2}, \frac{1}{2}\right)$$

Imagine we play this game with initial capital $S_1 = \$100$. *We keep* \$50 *and bet with the rest. Provided we win, we end up with* \$150. *In the next round, we reinvest* \$75 *and keep aside the other* \$75*. Our expected profit after the n-th round will be*

$$\begin{aligned} S_n &= S_1 e^{nW} \\ &= 100 \cdot 1.061^n \end{aligned}$$

2.2 Universal Portfolio

An universal portfolio, as the name suggests, is a portfolio selection scheme that is data-driven and independent of the underlying probability distribution of asset returns. Most known universal portfolios are causal and achieve the same exponential profit S_n as the S_n^* of the best CRP. That is

$$\frac{1}{n}\ln\left(\frac{S_n^*}{S_n}\right) \to 0 \qquad \text{almost surely}$$

Cover introduced *the Universal Portfolio* (Cov91), with investment vectors $\mathbf{b}_i$ defined by

$$\begin{aligned} \mathbf{b}_1 &= \left(\tfrac{1}{m}, \tfrac{1}{m}, \ldots, \tfrac{1}{m}\right) \\ \mathbf{b}_{i+1} &= \tfrac{1}{r}\int_B \mathbf{b} S_i(\mathbf{b})\,\mathrm{d}\mathbf{b} \end{aligned}$$

where

$$r = \int_B S_i(\mathbf{b})\,\mathrm{d}\mathbf{b}$$

is a normalizing factor and

$$S_i(\mathbf{b}) = \prod_{k=1}^{i} \mathbf{b}^t \mathbf{X_k}$$

is the profit achieved of a CRP with investment vector $\mathbf{b}$.

The universal portfolio can be thought of as a performance weighted combination of non-anticipating constantly rebalanced portfolios. Experiments by Cover show that the performance of the universal portfolio typically falls between the best performing asset and the best CRP.

2.3 Expert-Based Portfolio Selection

Györfi et al. (GLU06) introduced a more exotic investment strategy that is causal, universal and typically outperforms the ones listed above. The main idea is that there is an array of experts, each one giving an investment vector $\mathbf{b}_{n,k,l}$ according to the parameters k and l.

$$\mathbf{b}_{n,k,l}(X_1 \dots X_{n-1}) = \max_{\mathbf{b} \in \mathrm{B}} \sum_{i \in J_{k,l,n}} \ln \mathbf{b}^t X_i$$

with

$$J_{n,k,l} = \{k < i < n : \|x_{i-k} \dots x_{i-1} - x_{n-k} \dots x_{n-1}\| \le c/l\}$$

where $\|\cdot\|$ denotes the Euclidean norm.

Intuitively, each expert compares the string of last k return vectors to all windows of length k in history. l determines the threshold used to prune windows that are dissimilar. Thus, the expert $\mathbf{b}_{k,l}$ bases its weight vector according to situations that occurred in the past.

The composite $\mathbf{b}_n$ for trading day n is formed by

$$\mathbf{b}_n = \sum_{k=1}^{\infty} \sum_{l=1}^{\infty} \omega_{n,k,l} \mathbf{b}_{n,k,l}(X_1 \dots X_{n-1})$$

where $\omega_{n,k,l}$ is a normalized weight formed according to past performance. Györfi suggests exponential weighting according to the previous day's gain.

2.4 Distribution Dependent Portfolio Optimization

In the classic paper (Mar52), Markowitz lays the foundations of modern portfolio theory by minimizing the portfolio's risk for a target return. He formulates the optimal diversification as mathematical optimization, based on the first 2 moments of the asset returns distribution. The portfolio has return

$$\bar{x} = \mathrm{E}(\mathbf{x}) = \mathrm{E}\left(\sum_{i=1}^{N} b_i x_i\right) = \sum_{i=1}^{N} b_i \mathrm{E}(x_i)$$

and variance

$$\begin{aligned} \sigma^2 &= \mathrm{E}[(\mathbf{x} - \bar{x})^2] \\ &= \mathrm{E}\left[\left(\textstyle\sum_{i=1}^{N} b_i x_i - \sum_{i=1}^{N} b_i \bar{x}_i\right)^2\right] \\ &= \mathrm{E}\left[\left(\textstyle\sum_{i=1}^{N} \sum_{j=1}^{N} b_i b_j (x_i - \bar{x}_i)(x_j - \bar{x}_j)\right)^2\right] \\ &= \textstyle\sum_{i=1}^{N} \sum_{j=1}^{N} b_i \sigma_{i,j} b_j \end{aligned}$$

where $\sigma_{i,j}$ is the covariance between assets i and j.

Then the optimal portfolio for a target return x_p has minimal variance σ_m^2 and can be found by

$$\sigma_m^2 = \min_{\substack{\mathbf{b} \in B \\ \bar{x} = x_p}} \sum_{i=1}^{N} \sum_{j=1}^{N} b_i \sigma_{i,j} b_j$$

where B is the set of weights as in 1.10. The above expression can be conveniently solved by the method of Lagrange multipliers and its solution is widely known. The set of all solutions forms the efficient frontier, containing all minimum risk portfolios.

Many financists question the applicabilty of the Markowitz portfolio because to make it work, we first observe the market to estimate properly the assets' mean and variance. This is not always accurate and the estimates might become obsolete by the time they are used. The model has been extended numerous times, typically by including skewness and higher order moments as in (CRHM10).

2.5 Market Models

There are many models in Finance, each one simplifying certain analysis. The Capital Asset Pricing Model (CAPM) (see (Fre03) for historical and theoretical remarks) has had a prominent role about investing as it, among other things, determines an appropriate return of an asset if it is to be added to a well diversified portfolio. CAPM has shaped how financists think about asset returns and risk as it assigns a value of an asset in a portfolio that is different from the asset's market value and is based on the asset's sensitivity to non-diversifiable risk (denoted by the asset's β). An interesting consequence is that sometimes even a loosing asset might be useful if it is sufficiently uncorrelated with the market.

The Cross-Section of Expected Stock Returns (FF92) is an example of another famous model used in portfolio optimization that classifies assets according to its parameters. Many of the present models deal with assets in a portfolio while in the following chapter we develop a new model that can be used to estimate the future price of an asset, without reference to the portfolio it is in.

3

Market Model

A discrepancy between market models and investment strategies exists in that the latter work in practice, but not in the theoretical truly random environment of equation 1.9. Consider the Universal Portfolio: In equation 1.6 the random term typically dominates the price change making yesterday's profit obsolete information. Because random walks are uncorrelated in time, Györfi's experts method will maximize over the empty set J and thus always fail.

This chapter introduces a new market model, called the Damping model, which can be used for price estimation and also gives intuitive explanation why the strategies mentioned above work theoretically.

For background reading on difference equations and discrete time signals, see (PM96).

3.1 The Damping Model

The main idea of the Damping model is that true asset prices $\mathbf{P}_m$ are decoupled from the observed market vectors $\mathbf{P}_o$. Through out this section we explain the model by making analogies to a mechanical damped harmonic oscillator that mimics the operation of the market (figure 3.1).

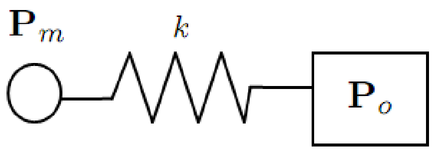

Figure 3.1: Damped harmonic oscillator - $\mathbf{P}_m$ reflects the true asset prices, while we can only see $\mathbf{P}_o$ as the current prices in the market

The *True asset prices* $\mathbf{P}_m$ are the true value of the asset and thus encompass all the company's dealings whose appreciation might not yet be known. Consider company XYZ releasing a new product on which it has heavily invested. It is not known whether it will live up to the stakeholders' expectations, which will become clear only in time. Nonetheless, an oracle can tell the product's influence on the company's stock and thus fix a proper price. Thus we model $\mathbf{P}_m$ as a random driving force.

Observed market prices $\mathbf{P}_o$ are the asset prices as seen in a stock exchange. They can be thought of as the observed position of a body subject to a drag force (eg. suspended in water) connected to $\mathbf{P}_m$ through a spring. The movement causes friction, which stands for persistence in investors' beliefs.

Consider again the example of XYZ. Even if the new product's sales are slow to take off, investors might be determined that it will be a success and thus decide to invest.

The *decoupling spring* serves to eventually propagate the true prices to the market prices.

The spring stiffness c_k and friction c_f coefficients are parameters of the model and determine its behavior. They also encompass the body's mass, which has a physical meaning, but is unnecessary in a market. For simplicity, we apply the proposed model to each asset individually and drop the vector notation from here on. Additionally, we use x for the observed market position in accordance with conventions from Physics.

From Newton's second law of motion (Mor08) we can express the acceleration $\ddot{x}$ of P_o as

$$\ddot{x} = -c_f\dot{x} - c_k(x - P_m)$$

Rearranging,

$$\ddot{x} + c_f\dot{x} + c_k x = c_k P_m \tag{3.1}$$

Because we are working with discrete data, we need to derive a discrete version of the above equation. Letting $x_n = x(n\Delta t)$, we substitute $\ddot{x}_i$ with the Laplacean operator $x_{i-1} - 2x_i + x_{i+1}$ and $\dot{x}$ with finite differences $1/2 \cdot (x_{i+1} - x_{i-1})$. Eq. 3.1 becomes

$$\left(1 + \frac{c_f}{2}\right) x_{n+1} + (c_k - 2)x_n + \left(1 - \frac{c_f}{2}\right) x_{n-1} = \frac{1}{c_k} P_{m,n} \tag{3.2}$$

3.2 The Market Transfer Function

To gain insights into how to predict future prices, we need to solve equation 3.2 for x_n. From superposition in linear difference equations, we can split the solution into two parts.

The *homogeneous solution* $x_{h,n}$ is obtained by setting the right-hand side to 0 and is of prime interest to us because it exposes how the system settles if left alone on its own. The *particular solution* $x_{p,n}$ can tell us little because $P_{m,n}$ is a random number and thus $x_{p,n}$ will be random as well.

Plugging $x_{h,n} = \gamma^n$ into 3.2, we get the characteristic polynomial

$$\gamma^{n-1}\left(\left(1 + \frac{c_f}{2}\right)\gamma^2 + (c_k - 2)\gamma + 1 - \frac{c_f}{2}\right) = 0$$

which we can solve for γ. The model parameters c_f and c_k govern the damping of the system and thus we can get $x_{h,n}$ in slightly different forms (see (Mor08) for details).

The solution of the above equation is

$$\gamma_{1,2} = -\frac{c_k - 2}{c_f + 2} \pm \frac{\sqrt{{c_f}^2 + (c_k - 2)^2 - 4}}{c_f + 2} \tag{3.3}$$

As both roots solve equation 3.2, the general homogeneous solution is[1]

$$x_{h,n} = A\gamma_1^n + B\gamma_2^n \tag{3.4}$$

The impulse response h_n of a discrete-time system is the output it produces from a unit sample excitation. In our case, this is simply equal to $x_{h,n}$ with the proper values for A and B. We can get

[1] We ignore the critically damped case of a single root in the discussion and leave it as an exercise.

the constants by explicitly writing out equation 3.2 in 2 time steps. Let us first substitute $P_{m,n}$ with the Dirac delta δ_n and all coefficients as follows

$$ax_{n+1} + bx_n + cx_{n-1} = \delta_n$$

We assume the system is at rest for $n < 0$, which gives

$$\begin{aligned} n = -1: \quad ax_0 + bx_{-1} + cx_{-2} &= 0 \;\Rightarrow\; & x_0 = A + B = 0 \\ n = 0: \quad ax_1 + bx_0 + cx_{-1} &= 1 \;\Rightarrow\; & x_1 = A\gamma_1 + B\gamma_2 = 1/a \end{aligned} \tag{3.5}$$

If the term under the square root in 3.3 is positive, $\gamma_{1,2}$ are real and the system is over-damped. From the first equation above, we get $A = -B$ and from the second

$$A(\gamma_1 - \gamma_2) = \frac{2}{2c_k + c_f c_k}$$

Therefore

$$A = \frac{1}{c_k(2 - c_k)}$$

If the $\gamma_{1,2}$ are complex, we get oscillatory behavior and the system is under-damped. We can express $x_{h,n}$ in the equivalent form of

$$x_{h,n} = \|\gamma_1\|^n C \sin(\theta n + D), \qquad \theta = \arccos(Re(\gamma_1)/\|\gamma_1\|) \tag{3.6}$$

Comparing with equation 3.5

$$\begin{aligned} x_0 &= C\sin(D) &= 0 \\ x_1 &= \|\gamma_1\| C \sin(\theta + D) &= 1/a \end{aligned}$$

From the first equation we get $D = 0$, which allows us to solve the second for C.

$$\begin{aligned} C &= 1/(a\,\|\gamma_1\|\,\sin(\theta)) \\ &= \frac{c_k}{1+\frac{c_f}{2}} \cdot \frac{1}{Im(\gamma_1)} \end{aligned}$$

Having the impulse response allows us to inverse equation 3.2 in the more convenient form of x being a function of P_m. From signal processing, we know

$$x = P_m * h \tag{3.7}$$

where the asterisk denotes convolution and $h \equiv x_h$ from before. This expands to

$$x_n = \sum_{k=1}^{n} h_k \, P_{m,n-k} \tag{3.8}$$

Note that for our purposes the impulse response $x_{h,n}$ is causal, which we achieve by truncating equation 3.6 for $n < 0$. Additionally, above we used $k = 1$ as the starting point of the summation because $x_{h,0} = 0$. This is somewhat consistent with our previous 1-based indexing[1].

[1] $P_{m,n}$ and x_n still start at $n = 0$.

3.3 Parameter Estimation

Estimating c_k and c_f from 3.2 is difficult because there are three unknowns ($P_{m,n}$ is the third) and one equation. Even if we repeat it in N time points, there are still $N+2$ unknowns. We can, however, pick an estimate for c_k and c_f, and then solve for P_m. This allows us to apply equation 3.8 to estimate x. We might be tempted to minimize the resulting error

$$\min_{c_f, c_k} \|x_{1\ldots N} - h(c_f, c_k) * P_{m,1\ldots N}\|_2 \tag{3.9}$$

The objective function above, however, is always 0 by construction because P_m is formed from x, c_f and c_k. For illustration, let's pick a slightly under-damped case with $c_f = 0.8$ and $c_k = 0.2$. The resulting transfer function h is shown in figure 3.2. Figure 3.3 shows the observed asset prices of IBM in 1H2010 versus estimated ones. The estimated graph is formed by computing P_m first and then using 3.8 to recover back P_o.

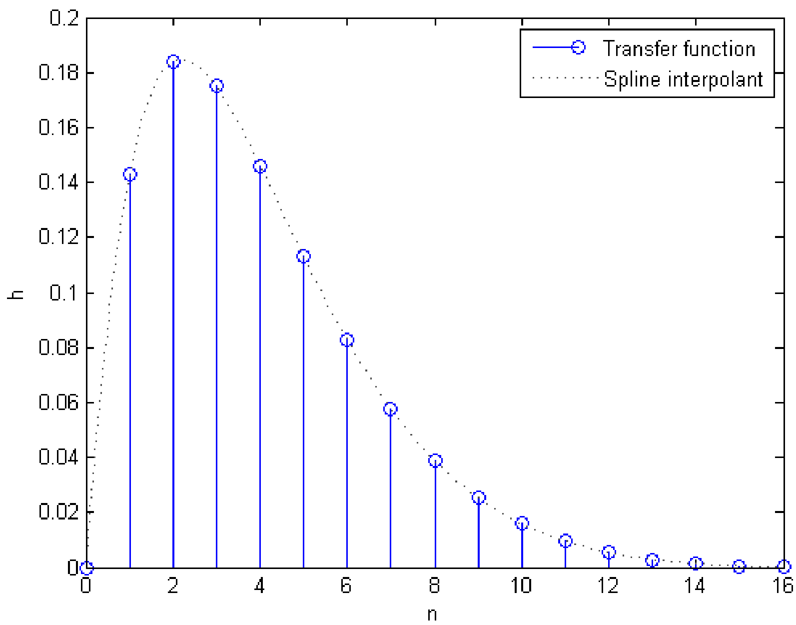

Figure 3.2: Transfer Function of the Damping Model for IBM - The dashed line is a spline interpolation of the transfer function to highlight its general outline

We might argue that there is an upper bound on the number of steps it takes for P_m to propagate to P_o, which restricts the domain of the parameters. Figure 3.4 graphically plots the error in parameter space for IBM's stock and there are still too many sets of parameters that yield acceptable recovery.

Let's look in more detail why equation 3.9 is always true. It estimates x_i from $P_{m,1\ldots i-1}$, but $P_{m,i-1}$ is a based on x_i itself! If we can, however, guess a value for $P_{m,i-1}$, we could causally recover x_i.

Assuming x follows a Brownian motion, equation 1.8 allows us to guess x_i (denoted by $\tilde{x}_i$). Then we form $\tilde{P}_{m,i-1}$ from x_{i-2}, x_{i-1} and $\tilde{x}_i$ directly by equation 3.2. Convolving the new P_m with h results in an estimate of x_i, which we hope is better than $\tilde{x}_i$, given the proper set of parameters.

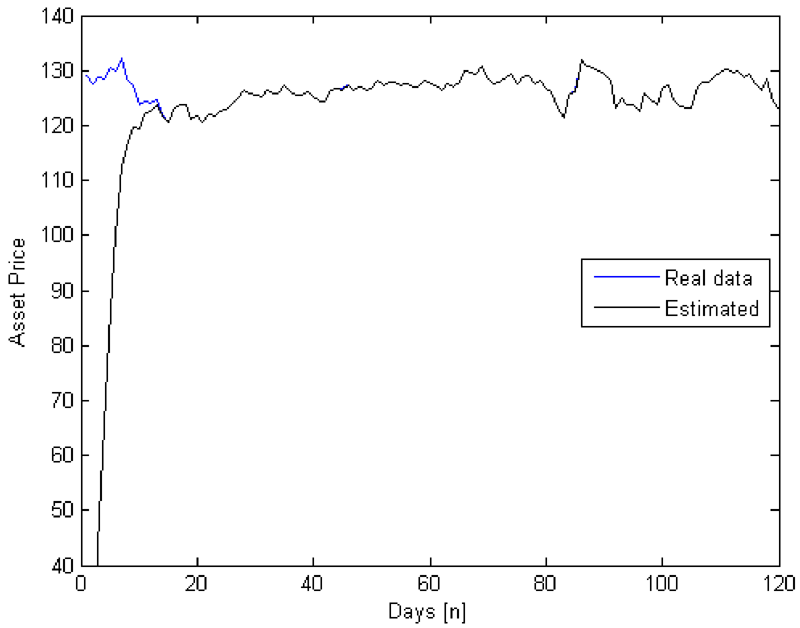

Figure 3.3: Estimating the price of IBM's stock - For illustration, we chose the initial estimate price to be 0, so it is visible how the spring pulls the weight and then it closely follows the real asset data.

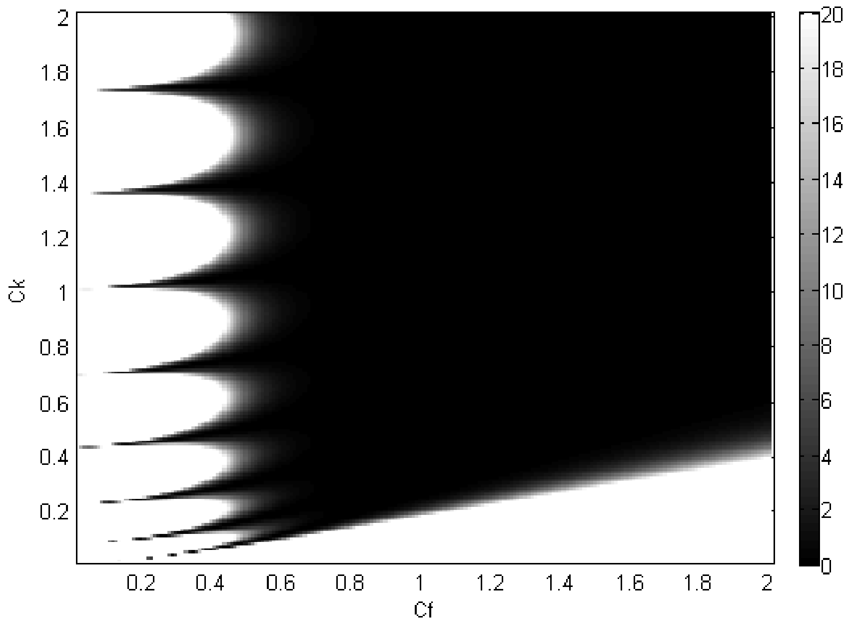

Figure 3.4: Objective function values for IBM's stock - The transfer function h was restricted to 16 samples. To get insights into the magnitude of error, the mean stock price is 126.52 and $N = 1000$

Therefore a more realistic and useful metric for estimating c_f and c_k is the total square error of prediction

$$\min_{c_f,c_k} \sum_{i=1}^{N} |x_i - \tilde{x}_i(x_{1\ldots i-1})|^2 \tag{3.10}$$

Equation 3.10 is difficult to optimize because it has no closed form solution and the objective function is chaotic with many local minima. Because the minima are often clustered together, a good strategy is to perform a 2-level uniform 2D search in parameter space. First we coarsely iterate over a larger parameter span and then focus around the minima found so far. Because of the randomness involved, we average several $\tilde{x}$ before using the result in 3.10. Figure 3.5 shows the procedure graphically.

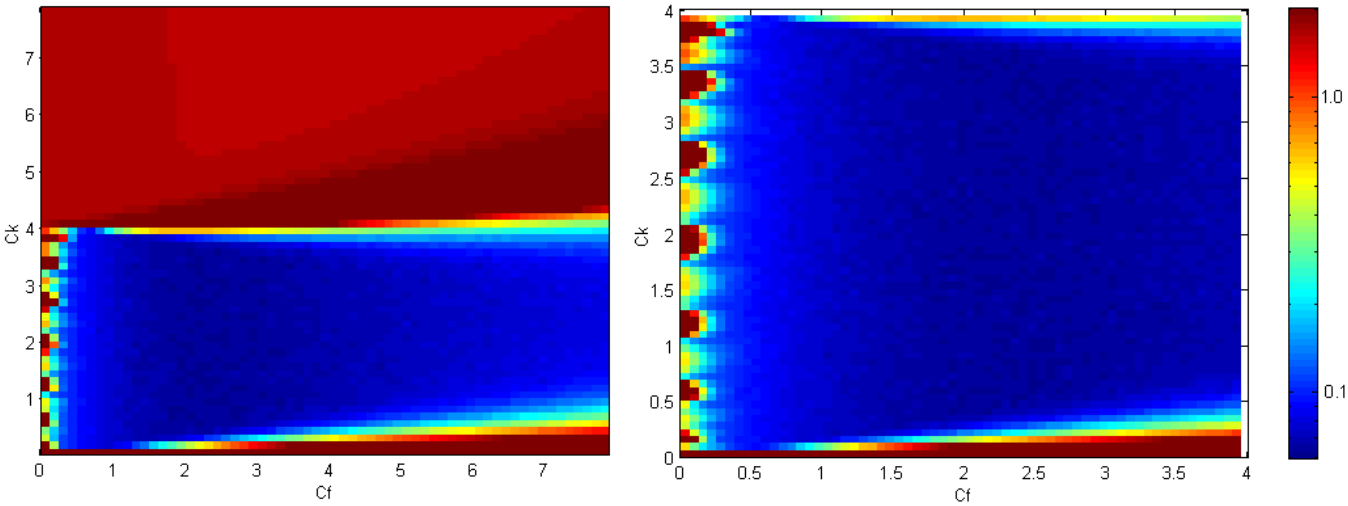

Figure 3.5: Estimation error - The results of a 2-level search in parameter space for a 256 days IBM stock sequence. The minimum square error of the asset returns is 0.059 at $c_f = 2.32$, $c_k = 2.51$.

The strategy above could be applied again for the new estimate and our guess would converge to the true value of x_i. Of course, the catch is that the second prediction typically is not better than the first, which already was completely random! To see why, let's write the *improved* guess as the convolution sum of $\tilde{P}_m$ and h split into a random and an exact component E.

$$x_i = h_1 \cdot \left(c_k + \frac{c_k c_f}{2}\right) \cdot \tilde{x}_i + E$$

where h_1 depends on the parameters and the damping case. For the over-damped, from equation 3.4 we get

$$h_1 = \frac{2\sqrt{c_f^2 + (c_k-2)^2 - 4}}{c_k(2-c_k)(2+c_f)}$$

If we were to use the Damping model to decrease the effect of the uncertainty of predicted values, the factor before $\tilde{x}_i$ needs to be less than 1. Instead of bogging down in details, we next present a much better-performing method for estimating c_f, c_k and x_i itself, by predicting a proper value for P_m.

Figure 3.6 shows that Pm ratios indeed show resemblance of a log-normal random process. From simulations we will later see that modeling it as a normal process performs better. We can guess a $P_{m,i}$ by recovering μ and σ of $P_{m,i}/P_{m,i-1}$ from the past few values of i and use them to generate a normal random number r. Then $P_{m,i} \approx rP_{m,i-1}$. Taking the mean of 16 such P_ms brings down the squared error to 0.014 for the dataset used in figure 3.5.

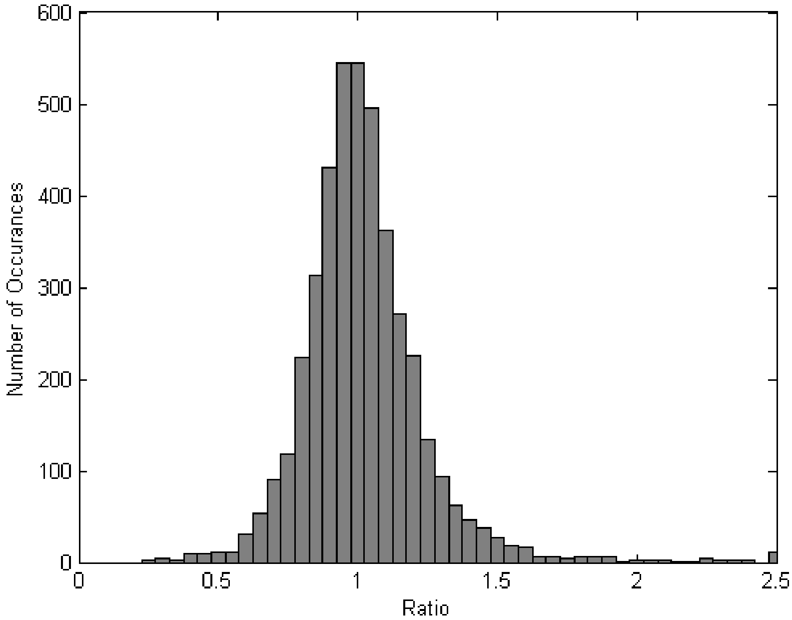

Figure 3.6: Histogram of P_m ratios - To decrease the effects of noise, the graph shows a histogram of IBM's $P_{m,i}/P_{m,i+1}$ for the 16 years period starting in 1990.

3.4 Monte-Carlo Simulation

To verify the Damping model, we perform a Monte-Carlo simulation to investigate its real-world applicability. Algorithmically, we loop over the observed prices and on the i-th iteration we do as follows

1. Compute $P_{m,i-16\ldots i-1}$ from $x_{i-17\ldots i}$
2. Estimate μ and σ for the distribution Φ of $P_{m,i+1}/P_{m,i}$
3. Generate N random ratios $\mathbf{r}$ from Φ and estimate $\tilde{\mathbf{P}}_{m,i} = \mathbf{r}P_{m,i-1}$
4. Treat each outcome equally probable and pick the portfolio weights $\mathbf{b}$ to maximize the expected log-return

Mathematically, the last step finds

$$W = \frac{1}{N} \max_{\mathbf{b} \in B} \sum_{i=1}^{N} \log\ w^t \tilde{P}_{m,i}$$

where from equation 1.3 we know what W is the average growth-rate and is the reason why we perform the optimization. The maximization routines need to evaluate the above expression numerous times, and thus we need to use the more efficient equivalent

$$W = \frac{1}{N} \max_{\mathbf{b} \in B} \log \prod_{i=1}^{N} w^t \tilde{P}_{m,i}$$

Because the actual value of W is not needed, we can also omit the scale by the constant in front.

To validate the implementation, we first run the optimization on the synthetic assets as shown in figure 3.7. Both of the assets are non-profitable, however, we can easily see that for periods of time one is significantly more profitable than the other. Figure 3.8 indeed shows significant profits. Note that the algorithm (discussed shortly) is causal and does not look into the future. It nevertheless predicts accurately future profits because for such regular processes, the h_1 of the transfer function is small and thus the current value of P_m is mostly predicted from history.

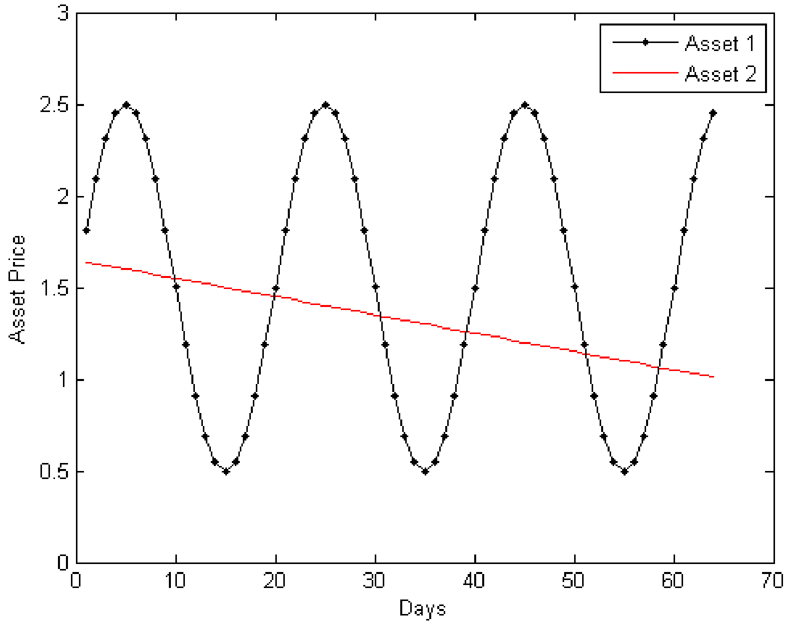

Figure 3.7: Synthetic Assets - To validate our algorithm, we run it on 2 uncorrelated non-profitable assets.

Perhaps the simplest, yet robust real-world test is a portfolio consisting of a risky asset and the risk-less investment, such as keeping the money in a bank. Program 1 gives the code listing of a MATLAB program to perform this task and figure 3.9 shows the results[1]

3.5 Comments and Conclusion

With the Damping model, we can intuitively justify the strategies mentioned in the preamble. The inertia of the market shows that Györfi's experts method exploits the fact that P_m cannot instantaneously modify the velocity of the observed market prices. Thus the set J can be thought of as moments back in time when the *weight* of figure 3.1 was moving with similar to the current velocity. It is like a bowling game, where different Js contain past trials with similar strike vectors. By comparing with them, we can quite accurately estimate how much we will score[2].

We have seen that the Damping model works well with predictable processes. (LM88) show that low value stocks show a much higher degree of correlation and we expect better performance if we

[1] $c_f = 1.2$ and $c_k = 3.2$ were used, which give weaker returns prediction, but perform better in portfolio optimization than the ones mentioned previously.

[2] To make the analogy complete, we need to assume very strong and chaotic wind over the track, though.

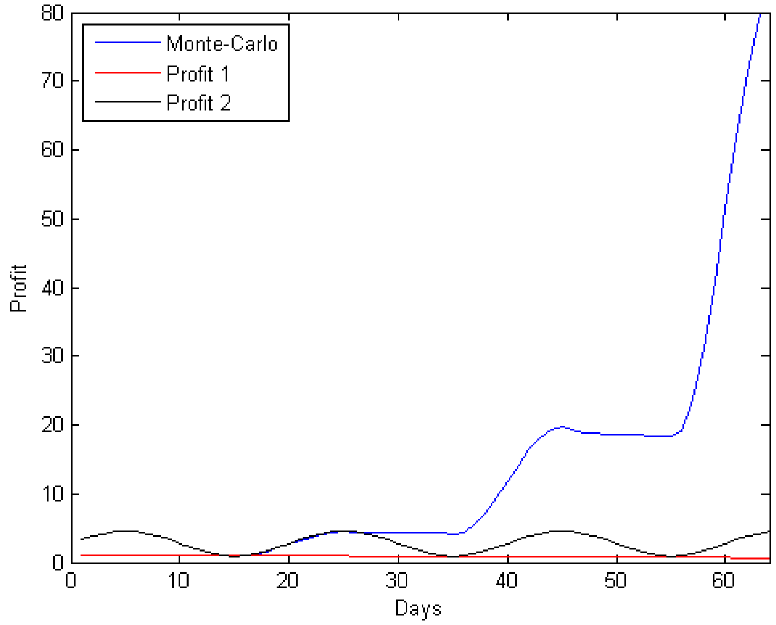

Figure 3.8: Profit from the Synthetic Assets - The Damping model predicts asset returns and the gain of the simulation stays relatively flat when the sinusoidal process is loosing.

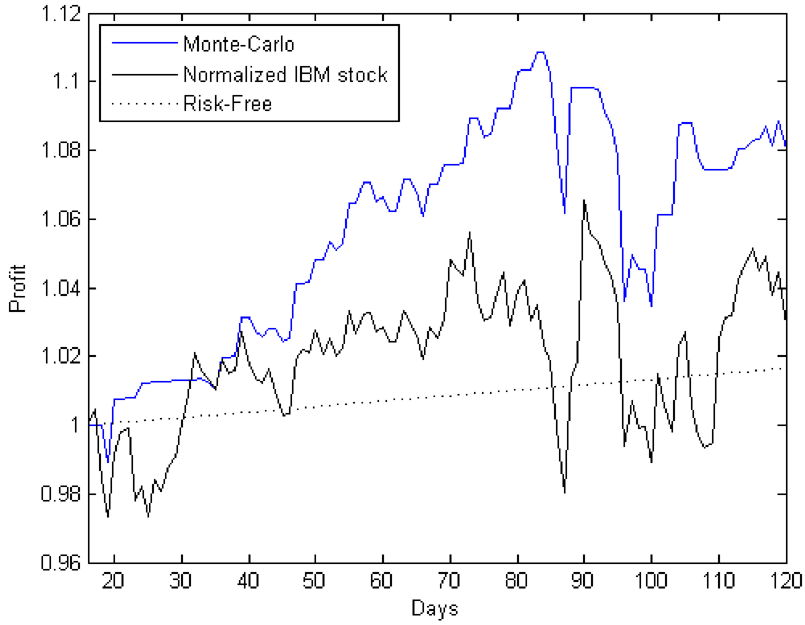

Figure 3.9: Simulation of a Single Asset Portfolio - Applying program 1 for IBM's stock in the first 100 trading days of 2010 results in a profit improvement of about 3%.

Program 1 Simulation program with 1 asset

```
% In:  P - observed prices, rf - risk free rate, cf, ck -
% model parameters, h - transfer function, n - number of runs
% Out: profit - daily profits
function profit = simrf(P, rf, cf, ck, h, n)

    len = size(P, 1);
    Pd = 0.5*[0; P(3:len) - P(1:len-2); 0];
    Pdd = [0; P(3:len) + P(1:len-2) - 2*P(2:len-1); 0];

    Pm = (Pdd + cf*Pd + ck*P)/ck;

    real_x = P(2:end)./P(1:end-1);

    % Reverse tr. function for convenience
    h = h(16:-1:1);

    p = ones(len,n);

    for r=1:n
        % start at 16, so that prediction is more accurate and to skip
        % bound checks / initial conditions for the convolution.
        for i=16:len-1

            Pm_history = Pm(max(1,i-32):i-1);
            Pm_ratio = Pm_history(2:end)./Pm_history(1:end-1);

            % calculate 32 sample Pms
            Pm_rand = Pm(i-1)*(mean(Pm_ratio) + std(Pm_ratio)*randn(32, 1));

            % explicit convolution
            pred_x = (h(1:end-1)*Pm(i-15:i-1) + h(end)*Pm_rand)/P(i);

            % log-optimal weight
            fobj = @(w) -prod(b*pred_x + (1-b)*rf(i));
            b = fmincon(fobj, 0.5, [1; -1], [1; 0]);

            % store profit
            p(i+1, r) = p(i, r)*(b*real_x(i) + (1-w)*rf(i));
        end
    end
    profit = sum(p, 2)/n;
end
```

apply the model to them. Although we have seen it works even in the rather random process of IBM, as a future work it would be interesting to investigate its performance on a much larger collection of assets.

(Fam91) states that transaction costs are sometimes a cause for market inefficiency. So far we have omitted them, but it would be useful to see their effect on the results or even better - incorporate them into the model itself.

Behavioral economics are also important to incorporate into any predicting model. For example, there are yearly market trends depending on the industry sector. One way to add them to the Damping model is to model the trends as forces on P_m.

4

Profit-Loss Prediction

Although the Efficient market hypothesis and the theory of Geometric Brownian motion mandate that asset returns are not correlated in time, we have seen this is not always true. This chapter shows my preliminary work on portfolio optimization, which led me to developing the Damping model. We will derive a few elementary strategies that show how past prices can be exploited for profit.

4.1 Theoretical Analysis

Previously we defined profit as $X_{i+1} = \frac{P_{i+1}}{P_i}$. Substituting equation 1.9 into this:

$$\begin{aligned} X_{i+1} &= \frac{P_1 e^{\nu(i+1)\Delta t} e^{\sigma \bar{\epsilon}_{i+1}\sqrt{(i+1)\Delta t}}}{P_1 e^{\nu i \Delta t} e^{\sigma \bar{\epsilon}_i \sqrt{i\Delta t}}} \\ &= e^{\nu \Delta t} e^{\sigma(\bar{\epsilon}_{i+1}\sqrt{(i+1)\Delta t} - \bar{\epsilon}_i \sqrt{i\Delta t})} \end{aligned}$$

Using $\sqrt{i+1} \approx \sqrt{i}$ for large i and Taylor expansion around 0 for the first term ($\nu \to 0$; in the GE example from figure 1.1, $\nu = 0.000135$), gives:

$$\begin{aligned} X_{i+1} &\approx (1 + \nu\Delta t) e^{\sigma\sqrt{i\Delta t}(\bar{\epsilon}_{i+1} - \bar{\epsilon}_i)} \\ &= \mu C^{\bar{\epsilon}_{i+1} - \bar{\epsilon}_i} \end{aligned} \tag{4.1}$$

where $C > 1$ is a non-random factor and μ is the asset's risk-less growth rate which can be seen by setting $\bar{\epsilon}_i = \bar{\epsilon}_{i+1} = 0$. From equation 1.9, we know that in fact

$$\bar{\epsilon}_n = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \epsilon_i$$

which implies $\mathrm{cov}(\bar{\epsilon}_n, \bar{\epsilon}_{n+1}) = 1$, as $n \to \infty$, making equation 4.1 a worthless approximation. Nonetheless, we can make significant profits based on it, which is what initially led me to believe that looking long back in history is futile. Let's for now assume equation 1.9 holds only for small n and thus $\bar{\epsilon}_n$ are not completely correlated.

Investment in an asset is profitable if $X_{i+1} > 1$, which is true if $\bar{\epsilon}_{i+1} > \bar{\epsilon}_i$. The decision to invest is made at time t_i when $\bar{\epsilon}_i$ is already known. Thus we are mainly interested in assets for which

$$\mathbf{P}(\bar{\epsilon}_{i+1} > \bar{\epsilon}_i | \bar{\epsilon}_i) > \frac{1}{2} \tag{4.2}$$

We typically want to include also assets that might not be profitable to decrease the volatility of our portfolio. Thus we can use the above probability as an argument to a weight computing formula discussed below.

$$\begin{aligned}\mathbf{P}(\bar{\epsilon}_{i+1} > \bar{\epsilon}_i|\bar{\epsilon}_i) &= \textstyle\int_{-\infty}^{\infty} I(x > \bar{\epsilon}_i)\,\phi(x)\,\mathrm{d}x \\ &= 1 - \textstyle\int_{-\infty}^{\bar{\epsilon}_i} \phi(x)\,\mathrm{d}x \\ &= \tfrac{1}{2}\left(1 - \mathrm{erf}\left(\tfrac{\bar{\epsilon}_i}{\sqrt{2}}\right)\right)\end{aligned} \tag{4.3}$$

From the market history we can recover $\bar{\epsilon}_i$. Using equation 1.9 and making the substitutions made above, the current price of an asset is

$$P_i = P_0\mu^i\left(e^{\sigma\sqrt{i\Delta t}}\right)^{\bar{\epsilon}_i}$$

Inverting,

$$\begin{aligned}\bar{\epsilon}_i &= \log_{e^{\sigma\sqrt{i\Delta t}}} \tfrac{P_i}{\mu^i P_0} \\ &= \tfrac{1}{\sigma\sqrt{i\Delta t}} \ln \tfrac{P_i}{\mu^i P_0}\end{aligned} \tag{4.4}$$

Substituting equation 4.4 into 4.3

$$\mathbf{P}_{\text{profit}}(P_i) = \frac{1}{2}\left(1 - \mathrm{erf}\left(\frac{1}{\sigma\sqrt{2i\Delta t}} \ln \frac{P_i}{\mu^i P_0}\right)\right) \tag{4.5}$$

$\mu = \mathrm{E}[\mathbf{X}]$ and $\sigma = \mathrm{stdev}(\mathbf{X})$ can be extracted from past returns. In section 4.1.1, we will see that equation 4.5 is relatively insensitive to them.

4.1.1 Approximation

Equation 4.5 looks a bit formidable, and although it can be efficiently implemented, we can simplify it considerably with a few approximations.

Consider the logarithm argument in 4.4 and assume $P_0 = 1$. $\frac{P_i}{\mu^i}$ is always positive and from experimental observation less than 1. Using

$$\ln(z) = 2\sum_{n=0}^{\infty} \frac{1}{2n+1}\left(\frac{z-1}{z+1}\right)^{2n+1}$$

From the first term, $\bar{\epsilon}_i$ becomes

$$\bar{\epsilon}_i \approx \frac{2}{\sigma\sqrt{i\Delta t}} \frac{P_i - \mu^i}{P_i + \mu^i}$$

We can linearly approximate equation 4.3 because it is relatively straight around $\bar{\epsilon}_i = 0$ as seen in figure 4.1.

$$\mathbf{P}(\bar{\epsilon}_{i+1} > \bar{\epsilon}_i|\bar{\epsilon}_i) \approx -0.4\bar{\epsilon}_i + 0.5$$

With the above, equation 4.5 becomes

$$\mathbf{P}_{\text{profit}}(P_i) \approx -\frac{0.8}{\sigma\sqrt{i\Delta t}} \frac{P_i - \mu^i}{P_i + \mu^i} + 0.5 \tag{4.6}$$

without any transcendental functions.

The next section further exploits the past returns' profitability and fits an appropriate distribution empirically.

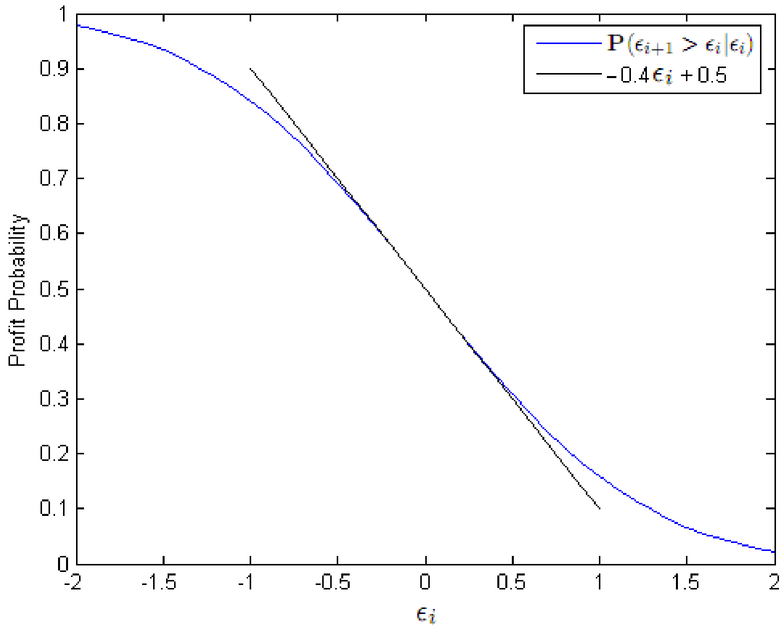

Figure 4.1: Probability of Profit - Approximation to $\mathbf{P}(\bar{\epsilon}_{i+1} > \bar{\epsilon}_i|\bar{\epsilon}_i)$

4.2 Profit/Loss Sequences

In equation 4.1, we can approximate $\bar{\epsilon}_{i+1} - \bar{\epsilon}_i$ with $\hat{\epsilon_i} \sim \mathcal{N}(0, 2)$[1] Then

$$X_{i+1} = \mu C^{\hat{\epsilon_i}} \tag{4.7}$$

which tells us that there is approximately equal chance if an asset performs well or not. Assuming $\hat{\epsilon_i}$ and $\hat{\epsilon_j}$ are independent for $i \neq j$, it might seem that history is irrelevant for predicting profits.

Figure 4.2 shows the return vectors X for Ford and IBM over a 40 days interval in 1991. Visually, returns seem not to be correlated in time (Fam91, Lue98), however, we see that they do seem to oscillate around unity. The average oscillation period seems slightly larger than 2 samples and sometimes there is more than one sample on the same side around $X = 1$. We hardly see, however, long strings of continuous loss.

A natural question to ask is: Given we already saw n consecutive loses, what is the probability that the asset will turn profitable in the next time step? To quantify our observations, we first apply a binary threshold whether an asset is profitable. Then we identify all windows in time that contain n consecutive loses and note the ratio of the number of windows that are followed by profit over their total number. This gives a profitability distribution and figure 4.3 shows the results on real-world data of 23 assets and 4263 trading days. Because the profitability distribution is somewhat varying in time, a surprisingly well performing approximation is

$$\mathbf{P}_{\text{consec}}(n_L) \approx 0.5 n_L \tag{4.8}$$

[1]By the formula $Z = X_1 + X_2 \sim N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$ for independent random variables.

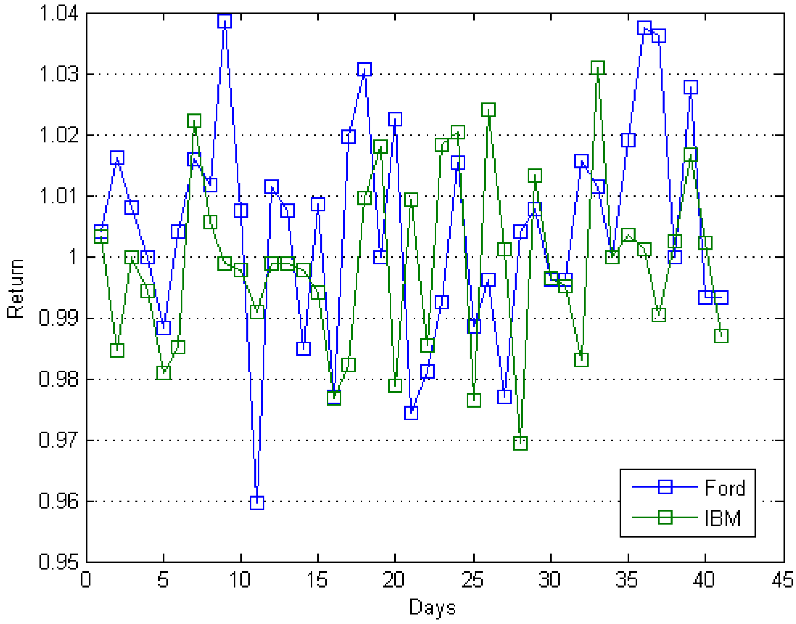

Figure 4.2: Returns History for Ford and IBM -

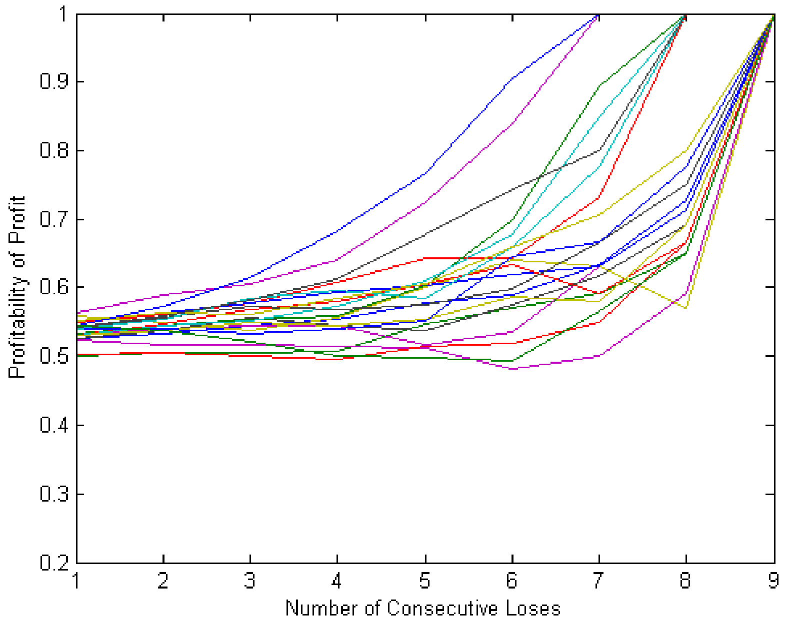

Figure 4.3: Profitability Distribution - For example, there is about 72% probability that the first red asset will turn profitable after seven consecutive loses.

4.3 Investment Strategy

In the previous sections, we derived two orthogonal probabilities based on the current price P_i and on the sequence of consecutive loses n_L. Although both of them provide reasonable investment strategies, combining them together gives a better performing portfolio characterized by its unnormalized portfolio vector $\mathbf{b_i}$

$$\mathbf{b}_i(n, P_i) = \omega_1(\mathbf{P}_{\text{consec}}(n_L)) \cdot \omega_2(\mathbf{P}_{\text{profit}}(P_i)) \tag{4.9}$$

where

$$\omega_1(x) = e^{c_1 x} \qquad \text{and} \qquad \omega_2(x) = x - c_2$$

Because $\mathbf{P}_{\text{profit}}$ is around 0.5, we would like a more pronounced weight for profitable assets and almost no weight for the others. Thus, $\omega_2(\ )$ can be thought of a contrast improving function. $\omega_1(\ )$ is an empirical function and has very strong impact on the gain of the strategy. Well performing values for the constants (chapter 4.6) are $c_1 > 30$ and $c_2 = 0.22$.

4.4 Transaction Costs

When transaction costs are introduced, our investment strategy needs to take into account the loses due to trading (see equation 1.12). The portfolio defined by equation 4.9 is based on predicting profitable assets and thus the investment vectors $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$ may end up very different. As a contrasting example, a buy-and-hold strategy has no loses in taxation as it performs no trading at all.

To decrease transaction costs, consider adding damping to the $\mathbf{b}_{i+1}$ by means of linear friction.

$$\hat{\mathbf{b}}_{i+1} = \mathbf{b}_{i+1} - c_3 \cdot (\mathbf{b}_{i+1} - \mathbf{b}_i)$$

where c_3 is the coefficient of friction. From equation 4.8, we can estimate how likely each asset is to continue to be profitable and modulate its friction accordingly.

$$\hat{\mathbf{b}}_{i+1} = \mathbf{b}_{i+1} - c_3 \mathbf{P}_{\text{consec}}(n_P) \cdot (\mathbf{b}_{i+1} - \mathbf{b}_i) \tag{4.10}$$

The parameter c_3 is somewhat dependent on the amount of transaction cost and as explored in the results chapter (4.6), values close to $c_3 = 0.87$ perform best for the given dataset[1].

[1]Fortunately, the three algorithm parameters vary little on real-world data and can be trained only once. Chapter 4.6 shows the optimal values

4.5 Algorithm

In this chapter we list the profit-loss portfolio strategy. The implementation is a simple for-loop in time over the return vectors. In the presence of transaction costs, movement of capital is restricted according to equation 4.10.

Although for clarity all algorithms shown here take for an input the complete returns vector $\mathbf{X}$, they are causal (i.e. operating without knowledge of the future). In a real-life scenario, on each trading day they will execute a single iteration of the outer for-loop to predict the portfolio weights $\mathbf{b}$ for the next day.

4.5.1 Profit-Loss Portfolio with no Transaction Costs

The profit-loss portfolio algorithm is a straight forward implementation of equation 4.9. Note that most lines operate on m-dimensional vectors and arithmetic operations are element-wise. For example, line 8 computes the standard deviation of each asset into $\sigma = (\sigma_1, \sigma_2, \cdots, \sigma_m)$

PROFIT-LOSS($X_{1 \dots N}, c_1, c_2, c_3, h$) :=

1 $p \leftarrow 1$
2 **for** $i \leftarrow 1 \dots N-1$
3 **do**
▷ Compute $\mathrm{P}_{\mathrm{consec}}$
4 $history \leftarrow X_{i,\dots,i-h} < 1$
5 $history \leftarrow$ CUMULATIVE-AND($history$)
6 $consec \leftarrow$ VERTICAL-SUM($history$)
7 $P_{consec} \leftarrow 0.5 \cdot consec$

▷ Compute $\mathrm{P}_{\mathrm{profit}}$
8 $\sigma \leftarrow$ STDEV($X_{1 \dots i}$)
9 $\mu \leftarrow \mathrm{E}[X_{1 \dots i}]$
10 $\epsilon \leftarrow \frac{1}{\sigma\sqrt{2i}} \log \frac{X_1 \dots X_i}{\mu^i}$
11 $P_{profit} \leftarrow 0.5(1 - \mathrm{ERF}(\epsilon))$

▷ Compute the weights
12 $w \leftarrow e^{c_1 P_{consec}}$
13 $w \leftarrow w \cdot \mathrm{MAX}(P_{profit} - c_2, 0)$
14 $w \leftarrow w\, / \mathrm{SUM}(w)$
15 $p \leftarrow p\, w^t X_{i+1}$
16
17 **return** p

Line 4 performs logical comparison over the $h \times m$ sub-matrix of X such that it sets the elements of *history* to 1 when the returns over the last h days are not profitable. The following example illustrates lines 4-6.

Example 3 *Consider*

$$X_{i,\dots,i-h} = \begin{pmatrix} 1.1 & 0.9 & 0.8 \\ 0.8 & 0.9 & 1.3 \\ 1.2 & 1.2 & 0.7 \end{pmatrix}$$

then

$$history = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

CUMULATIVE-AND *in line 5 performs the following operation*

$$\begin{pmatrix} X_i \\ X_i \& X_{i-1} \\ ... \\ X_i \& X_{i-1} \& ... X_{i-h} \end{pmatrix}$$

resulting in

$$history = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

and finally after VERTICAL-SUM *we end up with*

$$consec = \begin{pmatrix} 0 & 2 & 1 \end{pmatrix}$$

We can eliminate the parameter c_1 because in figure 4.4 we will see that performance is best as $c_1 \to \infty$. Lines 7 and 12 followed by the normalization in 14 in fact just select the assets that have been loosing for the longest, and thus the prior two can be replaced with

$w \leftarrow history == \text{MAX}(history)$

The other parameter in the listing - h governs how long back into history the algorithm considers. Experimentally, values after 3 bring diminishing returns, while $h = 3$ performs much better than 1 or 2.

4.5.2 Transaction Costs

Following equation 4.10, we need the probability P'_{consec} that a profitable asset will loose. This can be accomplished by the equivalent of lines 4-5 with the '<' sign replaced with '>' in line 4. The final weight needs to be further modulated by

$w \leftarrow w - c_3 P'_{consec} \cdot (w - lastw)$

followed by renormalization. *lastw* are the portfolio weights from the last iteration.

4.6 Results

The input data to all algorithms benchmarked in this chapter is the return vectors X_i of 23 assets[1], traded on the NYSE for the 4263 trading days from 02/01/1990 to 24/11/2006. Prior to computing the returns, all closing prices for day i are adjusted for dividends and stock splits, according to:

$$X_i = \frac{P_i}{P_{i-1} - \text{CASH\_DIV}_i} \frac{\text{SPLIT\_MULTIPLIER}_i}{\text{STOCK\_DIV}_i + 1}$$

4.6.1 Parameter Analysis with no Transaction Costs

The performance of our strategy depends on the parameters c_1 and c_2. Figure 4.4 shows that both the profit and the Sharpe's ratio asymptotically increase as $c_1 \to \infty$. The effect of high values for c_1 is that the strategy gives weight only to the assets that have been loosing for the longest amount of time, i.e. it only gives weight to assets having a loss of 3 consecutive trading days, if any. Else, to those of 2 consecutive loses and then to those of 1.

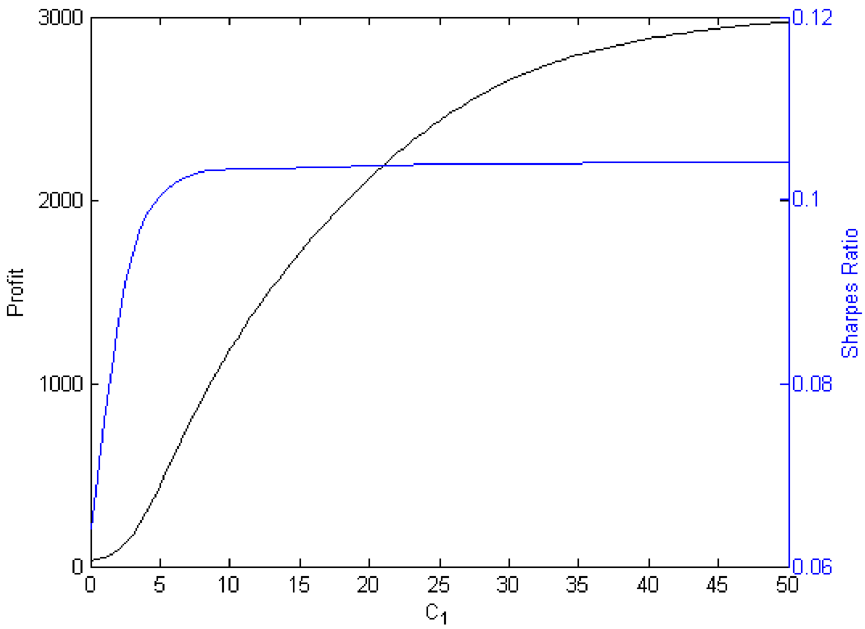

Figure 4.4: Effect of c_1 on profit and Sharpe's Ratio - The higher values of the parameter c_1 give more weight to assets that have been loosing several times in a row

Having fixed c_1, figure 4.5 shows how c_2 affects performance. At $c_2 = 0.22$, the profit-loss portfolio strategy achieves a maximal profit of 3069, with a Sharpe's ratio of 0.1043. In comparison, the non-causal best constantly-rebalanced portfolio achieves 27.081, with a Sharpe's ratio of 0.0512.

Figure 4.6 shows a time plot of the wealth achieved by both strategies.

4.6.2 Parameter Analysis with Linear Transaction Costs

In this section the capital gained on a trading day by an algorithm is discounted according to equation 1.12.

[1] Risk-free, ahp, alcoa, amerb, coke, comme, dupont, ford, ge, gm, hp, ibm, inger, jnj, kimbc, kinar, kodac, merck, mmm, moris, pendg, schlum, and sherw

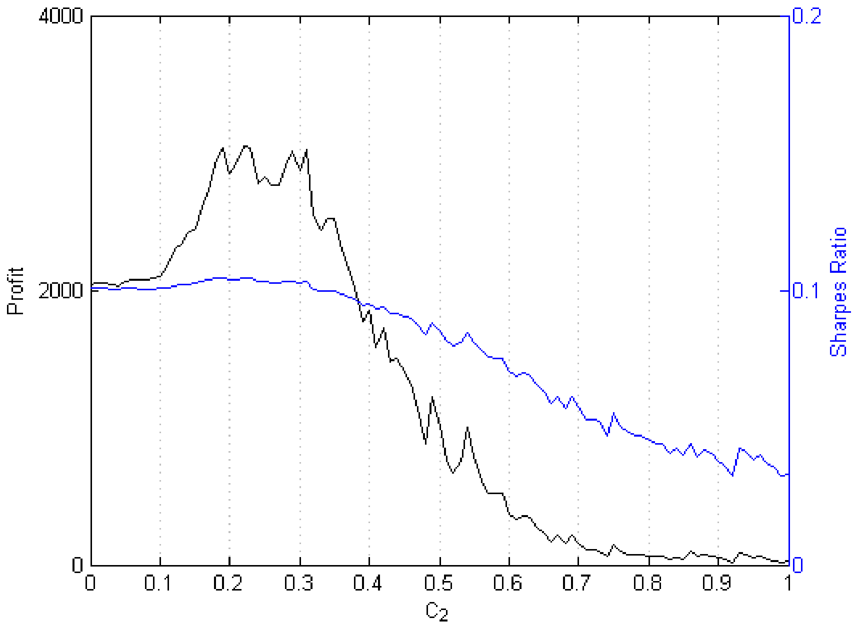

Figure 4.5: Effect of c_2 on profit and Sharpe's Ratio - The portfolio performs best at $c_2 = 0.22$

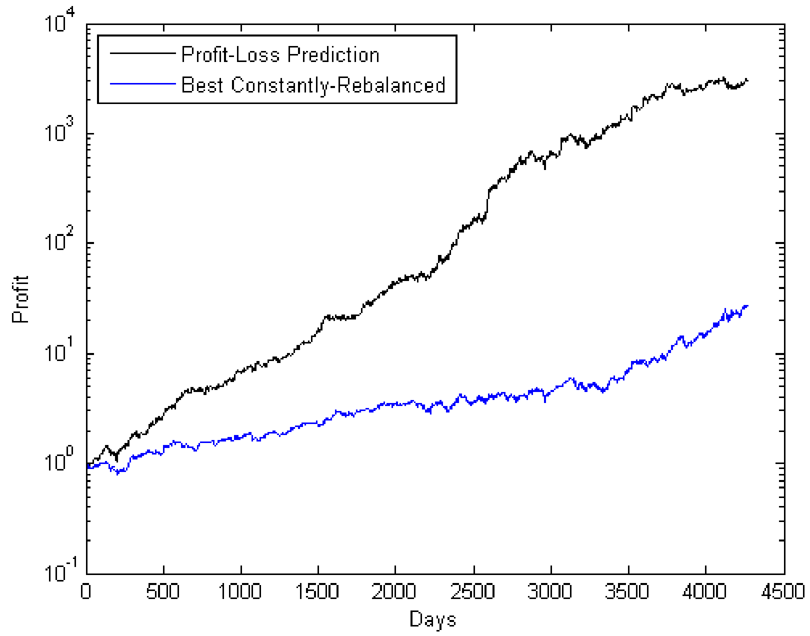

Figure 4.6: Profit-Loss Portfolio Performance - In the 16 year test period, the profit-loss portfolio outperforms the best constantly rebalanced portfolio by more than 2 orders of magnitude. For the same time-interval, the risk-free asset grows 2.1089 times.

	c=0	c=0.15%
Best Asset	12%	12%
Best CRP	22%	21%
Profit-Loss	61%	29%
Expert-based	137%	89%

Table 4.1: Comparison of Performance

Setting the friction parameter c_3 to values other than 0 improves performance when transaction costs c_t are considered. Figure 4.7 shows that for $c_t = 0.15\%$ performance is best at $c_3 = 0.87$. It turns out that c_3 is an asymptotically bounded, monotonically increasing function of c_t. For a wide range of transaction costs, however, $c_3 = 0.87$ performs well.

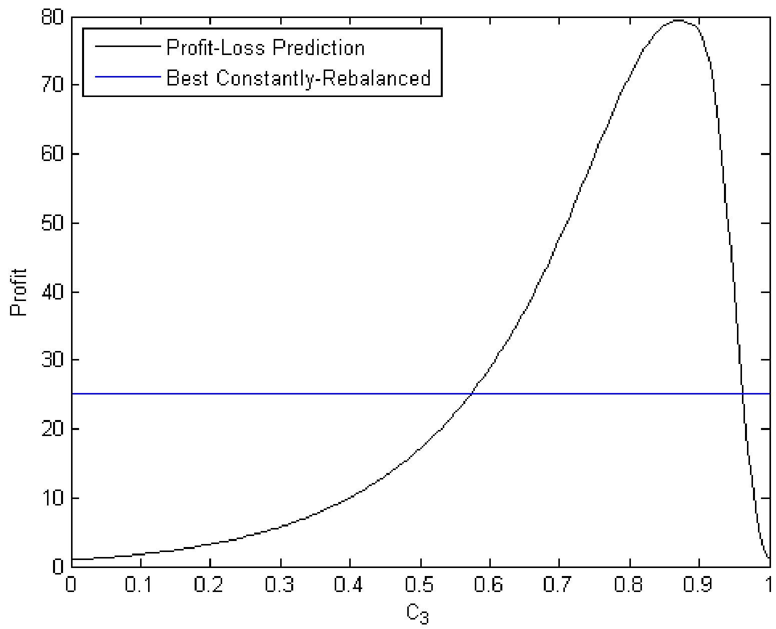

Figure 4.7: Effect of c_3 on profit - The effect of the parameter c_3 on the profit with transaction cost of 0.15%. The CRP's profit of 25.23 is also shown for comparison

Figure 4.8 shows the profit obtained for varying transaction costs. For $t_c \leq 0.23\%$, the profit-loss portfolio outperforms both the best constantly rebalanced portfolio (profit of 24.09) and the buy-and-hold of the best asset (profit of 18.75)

4.7 Comparisons and Conclusion

Table 4.1 compares the performance of the profit-loss portfolio to other exemplary portfolios[1]. The profit-loss portfolio falls second in terms of performance. In terms of run-speed, subjectively it runs instantaneously, while the best CRP takes seconds and the expert-based - a day.

[1]The expert-based gains might be over-estimated because the algorithm was run on an earlier period of the same assets.

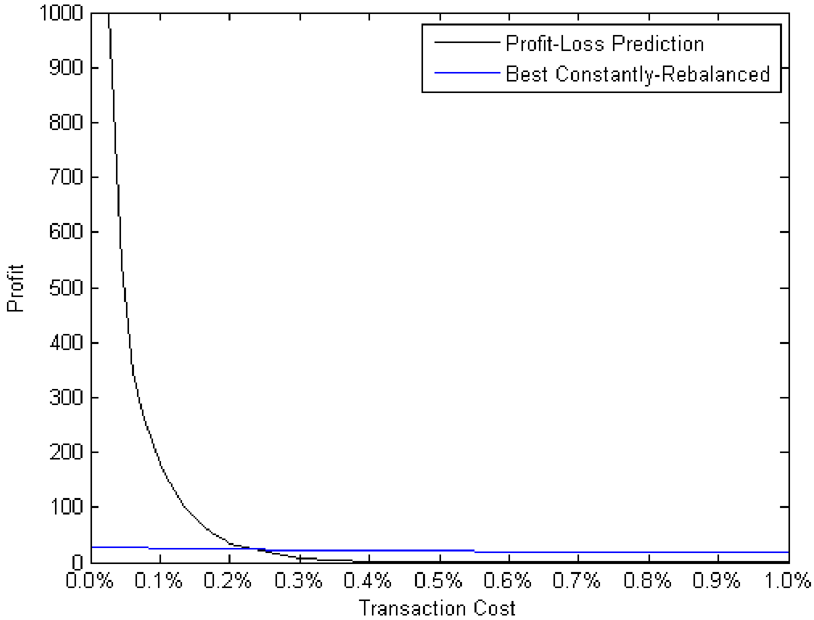

Figure 4.8: Profit with Transaction Costs - The profit-loss portfolio is relatively sensitive to transaction costs

The simplicity and high gains of the profit-loss portfolio are, in my opinion, worth further investigation. It led me to question the efficiency of markets and to develop the Damping model. Because the profit-loss portfolio as I presented it does not derive from the model, it might be instructive to further develop and unify them.

References

[Cov91] Thomas Cover. Universal portfolios. *Mathematical Finance*, 1:1–29, January 1991. 7

[CRHM10] Merrill W. Liechty Campbell R. Harvey, John C. Liechty and Petter Müller. Portfolio selection with higher moments. *Quantitative Finance*, pages 469–485, 2010. 9

[Fam91] Eugene Fama. Efficient markets: Ii. *Journal of Finance*, pages 1575–1617, 1991. 2, 20, 23

[FF92] Eugene Fama and Kenneth French. The cross-section of expected stock returns. *Journal of Finance*, pages 427–465, 1992. 9

[Fre03] Craig W. French. The treynor capital asset pricing model. *Journal of Investment Management*, pages 60–72, 2003. 9

[GLU06] L. Györfi, G. Lugosi, and F. Udina. Nonparametric kernel-based sequential investment strategies. *Mathematical Finance*, 16:337–357, 2006. 8

[GOU08] L. Györfi, Gy. Ottucsák, and A. Urbán. Empirical log-optimal portfolio selections: a survey. Machine Learning Summer School 2007, MLSS 2007 (invited lecture), 2008. 6

[Hul02] John Hull. *Options, Futures, and Other Derivatives*. Prentice Hall, 5th edition, 2002. 3

[LM88] Andrew W. Lo and A. Craig MacKinlay. Stock market prices do not follow random walks: Evidence from a simple specification test. *The Review of Financial Studies*, pages 41–46, 1988. 17

[Lue98] David Luenberger. *Investment Science*. Oxford University Press, 1st edition, 1998. 1, 23

[Mar52] Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7:77–91, March 1952. 8

[Mor08] David Morin. *Introduction to Classical Mechanics*. Cambridge University Press, 1st edition, 2008. 11

[PM96] John Proakis and Dimitris Manolakis. *Digital Signal Processing: Principles, Algorithms and Applications*. Prentice-Hall International, 3rd edition, 1996. 10

[Sch02] Dominik Schäfer. *Nonparametric Estimation for Financial Investment under Log-Utility*. PhD thesis, Stuttgart University, July 2002. 4

[Sha94] William Sharpe. The sharpe ratio. *Journal of Portfolio Management*, 21:49–58, 1994. 5