ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Modeling Equity Markets with Agent Based Models: Trade Indicators and Calibration Methods

Master Thesis

Sindri Már Kolbeinsson

April 28, 2015

Advisors: Dr. Qunzhi Zhang
Lucas Fiévet, M.Sc. ETH
Tutor: Prof. Dr. Didier Sornette

Chair of Entrepreneurial Risks
Department MTEC
ETH Zürich

Abstract

Rich literature on the relevance and application of Agent Based Models in the context of financial markets has inspired recent efforts to explore their potential for predicting asset prices (see Zhang (2013)). This thesis contributes to these efforts on two fronts. Firstly, it dives deeper into the model's predictive capability by analyzing model parameters in search for indicators of increased probability of correct predictions. Secondly, it addresses the model's computationally intensive calibration process in an attempt to increase in-sample prediction rates by decomposing the model, analyzing agents' decisions and creating a few simplified Agent Based Models that are tested in terms of prediction rates and trading strategies. Results from broad scoped analysis of model parameters did yield some indicators of heightened prediction capability, although entirely dependent on which underlying asset is being modeled and when. Results from efforts to simplify calibration and increase predictive capabilities were promising in that deterministic one- and three agent models, who emulate short pieces of the time series perfectly, showed promising prediction rates and Sharpe ratios.

Contents

Chapter 1

Introduction

Agent Based Models (ABMs) are a well known tool for modeling the dynamics observed in complex adaptive systems. By way of replicating the macroscopic behavior of these systems, ABMs provide useful insight into the collective behavior of its agents, thus potentially exposing the system's internal properties. The central notion of agent based modeling is that on the individual level, agents have well defined properties and rule based behavior. However, on a larger scale, their interaction might lead to complex behavior and emergent phenomena. The benefit of ABMs can be derived directly from there, as they possess a key attribute of complex adaptive systems, namely the fact that they cannot be reduced to a sum of their parts. Or more precisely, attributes of the system's emergent phenomena cannot be mapped to its individual entities (Bonabeau, 2002). ABMs have been applied in numerous fields, including biology, economics, network theory and business. In this paper we will examine its use for financial markets and seek to uncover its potential for solving a well researched topic within finance, the prediction of asset prices.

Financial markets have proven difficult to predict, as reflected in the results of recent studies of performance of mutual funds (Barras et al., 2010; Fama and French, 2009). These findings serve to support the Efficient Market Hypothesis (EMH), which maintains that one cannot routinely gain returns in excess of the average risk-adjusted market return. Investors would have rational expectations and there would be no inefficiencies for them to exploit. Moreover, its weak form states that all information on historic prices is already reflected in the current price, thereby eliminating the possibility of predicting future prices using historic prices (Fama, 1970, 1991).

The weak form EMH therefore dictates that we should not be able to use ABMs for predicting the market, as asset prices follow a random walk (Malkiel, 2003) having no pattern whatsoever. Critics of the EMH have however pointed out possible inefficiencies in financial markets which give rise to

the notion that they are predictable to a certain extent. For example, Jegadeesh and Titman (2001, 1993) find that due to investors' overreaction, stock prices tend to trend positively for recent winners and negatively for recent losers. Similarly, behavioral finance can explain the tendency among investors to herd during certain time periods, causing bubbles and crashes in asset prices, completely disregarding the underlying value of assets (Brunnermeier, 2001). The above suggested inefficiencies in financial markets can for instance be reflected in recent events such as the financial crisis, the following sovereign debt crisis and soaring US equity markets, where the S&P 500 index reached an all-time high in December 2014. These findings and events that seemingly undermine the EMH have fuelled researchers in their search for a way to predict markets. Furthermore, research on agent based modeling has contributed to this search with promising results, such as in Jefferies et al. (2001) and Zhang et al. (2013).

Some important catalysts for agent based modeling in the context of simulating financial markets can be found in Rubinstein (1997). The concept of bounded rationality can easily be linked to participants in financial markets, e.g. in the way information available to agents does not lead them to the same decision. Real agents are simply not all capable of analyzing the information for acting in a completely rational way, resulting in varying economic success. ABMs, although not commonly used in traditional economic models, attempt to model the bounded rationality of agents and therefore capture the inefficiencies relating thereto. Or perhaps more precisely, attempt to capture the evidence of systematic deviation from the behavior of completely rational agents. By leaving out the concept of a known equilibrium in financial markets, ABMs can lead us closer to modeling market participants' choices and the assumptions they make about their environment (Rubinstein, 1997).

ABMs designed for modeling complex dynamics of real financial markets usually have a large number of parameters, which, coupled with their nonlinear influence on the outcome, makes it hard to calibrate the models with real financial time-series. Previous research with ABMs has however succeeded in reproducing many empirical findings in real markets, or the so called stylized facts of stock markets (Hommes, 2002, 2006). These include fat tails in return distribution of financial markets as well as clustered volatility and long memory. Moreover, Zhang et al. (2013) has found ABMs to successfully reproduce the above mentioned stylized facts in addition to absence of auto-correlation, gain/loss asymmetry, aggregational Gaussianity and more.

This report will partly build on the work of Zhang et al. (2013) who has constructed a mixed-game ABM to "reverse-engineer" financial markets by calibrating the model to real financial time series using a genetic algorithm.

Specifically, the model employs agents playing 4 different games, the Minority Game, Delayed Minority Game, Majority Game and the $-Game. The Minority Game, introduced by Challet and Zhang (1997), has already been subject to considerable research. Among all agents, agents playing the minority game consistently try to make a trade decision opposite to the majority, thus taking on the role of value investors who seek to buy when the asset is undervalued and sell when it's overvalued. Agents playing the delayed minority game have similar characteristics except that their decision preference is to be in the minority of the succeeding time step. The Majority Game, as described in Marsili (2001), includes agents who prefer making the same decision as majority of agents, making them trend followers in the market. Likewise, the $-Game, introduced by Andersen and Sornette (2003), includes agents that wish to make the most popular decision of the succeeding time step in the current time step, making them trend predictors. Combining different games such as the minority game and majority game in a model has already indicated some statistical properties of their interaction. Marsili (2001) and Lux and Marchesi (1999) have for instance found fundamental value investors, represented by the minority game, to be associated with switches between market phases. Additionally, trend followers, represented by the majority game, were found to cause speculative bubbles in the market.

When all is put together, a virtual market place is created with agents who are bounded rationally in their decision making in a model with no inherent equilibrium whatsoever, creating an artificial financial market which emulates the dynamic and inefficiency apparent in real markets. As mentioned above, this has been shown to reproduce many stylized facts of real markets. We will therefore not look for further validation of this ABM modeling method in terms of common properties with real financial markets, but will aim to shed further light on its predictive power and look closer at agents' individual strategies and decisions.

In particular, the thesis will be in two parts, both of which seek to illustrate the potential of using ABMs for trading. First, in section 3.2, we will rely on output from the ABM defined in section 2 for researching the possibility of enhancing an ABM based trading strategy with prediction indicators. Such a strategy could for example be based on statistical relationships between parameters in the model's output and properties of the actual time series being reverse engineered, in addition to the ABM's predicted returns. We will analyze the model's output from chosen experiments and report the relationship between parameters that we think might have potential for enhancing the model's predictions. We will therefore not construct or test any specific trading strategy, but only look for indicators that could be used in their construction. The search for such prediction indicators seemed to reveal heightened probability of correct model prediction for certain values

of specific model parameters. This was however purely dependent on what underlying time series was reverse engineered, and for which period in time. The practical application of such indicators is naturally limited.

Secondly, in section 4, we will take a few steps back from the sophisticated modeling and calibration method defined in section 2 and focus on agents' individual behavior. We will dig deeper into the potential of binary strategies as a trade decision tool for agents and contemplate their potential for simplifying the reverse-engineering process. In general, we believe there might be a simpler and more efficient approach for calibrating an ABM to real time series, which could be centered around the construction of each agent's strategy set. As an initial step for exploring this possibility, we'll demonstrate a few ABMs consisting of one or extremely few agents with potentially high prediction rates. We will then test basic trading strategies constructed from these simplified ABMs. Our initial results indicated that high prediction rates can indeed be attained with these models as well as strong performing trading strategies, when bench-marked to random strategies.

Chapter 2

The ABM

What follows is a basic description of the ABM. It was used by Zhang et al. (2013), whose results we use in subsequent analyses. Although the purpose of this paper is not to directly evolve or build upon the model itself, we feel it's necessary to explain it here in order for the reader, who is perhaps unfamiliar with ABMs, to familiarize him- or herself with the fundamental workings of the model. It will undoubtedly be easier for the reader to form an informed opinion of later analyses and results when having full knowledge of the underlying ABM itself. We refer to Zhang et al. (2013) for a more detailed description and the motivations behind various properties of the model which might be missing from the following explanation.

2.1 Agent behavior

As just mentioned, the ABM described here for predicting financial markets was implemented by Zhang et al. (2013) who was under the influence of recent research such as Challet and Zhang (1997), Jefferies et al. (2001), Andersen and Sornette (2003) and Wiesinger et al. (2012). This approach involves constructing a virtual stock market and populating it with a fixed number of agents, N. Agents' rule based behavior is then defined on a microscopic level, such that their interaction ultimately leads to a complex adaptive system with the properties of real financial markets. Their decisions to buy or sell shares of the virtual asset thus collectively make up the dynamics of our prediction tool. For simplification, the virtual agents are limited to trading a single asset. The imbalance of buy and sell orders at each time step then causes variation in asset prices, creating a time series of prices. The aim is for the synthetic time series to resemble the time series of real asset prices as much as possible. With adequate resemblance, we hope that some insight into real markets, through examination of the ABM, can be gained. This sort of reverse engineering was introduced in Johnson N. F. (2001) and Jefferies

et al. (2001) by first using a synthesized time series produced by an ABM, whose properties they pretended not to know, as the subject for reverse engineering. Much like here, they would then start with another ABM with randomized parameters and evolve it until the difference between the two time series was minimized.

Agents have three options at each time step; buy, sell or do nothing. As previously mentioned, when all agents have made their decision, the mismatch of buy and sell actions at each time step will produce a time series of returns. If t are discrete points in time and Z is the length of a trading period, returns for $t \in \{0, \ldots, Z\}$ are expressed as $\{r_1, \ldots, r_Z\}$ where

$$r_t = \frac{p_t}{p_{t-1}} - 1. \tag{2.1}$$

As agents are limited to using only past returns for making a decision, the decision process is somewhat simplified. However, if agents are able to trade varying amounts of shares, the number of trading strategies that project a finite history of returns onto a trade decision is endless. By reducing the return time series to binary form, such that negative r_t is represented by 0 and positive by 1, and by limiting the number of shares agents can trade to 1 at each time step, trading strategies for agents are computationally feasible. An agent's decision to trade is then represented by 1 if he is buying a share and -1 if he's selling. The entire set of strategy functions then becomes

$$\mathbf{F} := \{f | \forall f : \{0,1\}^Z \to \{+1, -1\}\}, \tag{2.2}$$

Where f is a strategy function that derives a trade decision from one possible return history. The total number of available strategy functions then sums up to 2^{2^Z}.

As previously mentioned, we also want to represent the bounded rationality of agents in the model. Just as in real markets, it is not assumed that agents have a capacity to account for the entire information history when making their decision. More realistically, they are only able to remember a certain number of steps back, denoted by their memory, m, where $m << Z$. Agents can therefore only use information from the m-sized binary vector of most recent market returns, μ_t, when deciding their action. This reduces the entire strategy set available down to 2^{2^m} and one strategy, as a function of μ_t, can be expressed as

$$f(\mu_t) : \{0,1\}^m \to \{+1, -1\}. \tag{2.3}$$

In line with the bounded rationality of agents, and to ensure sufficient heterogeneity among agents, each one can only employ a certain amount of

strategies, s. In order to adequately represent the intellectual limit of agents as well as make it possible for agents to have no overlapping strategies, an appropriate value for s can be $s \cdot N < 2^{2^m}$. However, overlapping strategies are possible for agents. The collection of strategies for agent i, $i \in \{1, \ldots, N\}$, is then represented by $F_i := \{f_i^1, \ldots, f_i^s\}$. All agents also have the same memory length, m, and number of strategies, s.

As the definition of agents' trading strategies is now clear, we need a way to evaluate their success. Agents make decisions by employing their most successful trading strategy, the success of which is ultimately bound to the game the agent is playing and derived from the last m time steps according to agents' memory. Before we introduce success of trading strategies, we will define how the strategies pay off at each time step.

The payoff function for agents, π, is dependent on which of the four games the agent is playing. When simplified, the payoff function can be expressed as a mapping of both the market majority- and individual agent decision onto either -1 or 1; $\{-1, 1\}^2 \to \{-1, 1\}$. A typical payoff for an agent playing the minority game would for instance be 1 when the market's majority action is opposite of the respective agent's action and -1 otherwise. Conversely, the payoff for an agent playing the majority game is 1 when his action is the same as the market majority's and -1 otherwise. The payoff functions are thus meant to emulate investor behavior observed in real markets. We will explain each game further in section 2.2.

Success of individual trading strategies is measured by the sum of payoffs in the last m time steps. We denote the success as a function of time and strategy, $\forall j = 1, \ldots, s$, and $\forall i = 1, \ldots, N$, as

$$U(f_i^j, t) = \sum_{\zeta=t-m}^{t-1} \pi(f_i^j(\mu_\zeta), \text{sign}(r_\zeta)). \tag{2.4}$$

Each agent therefore has the ability to calculate the success of each of his s strategies and compare them. The choice of which trading strategy to employ consequently comes down to the following maximization

$$f_i^* = \text{argmax}_{f_i^j \in F_i} U(f_i^j, t). \tag{2.5}$$

As already explained, agents have three options at every time step; buy, sell or do nothing. The third option, do nothing, is important for capturing effects of liquidity in the virtual market and can have a great influence on the number of agents trading. Intuitively, agents refrain from participating in the market if they're not happy enough with their best performing trading strategy. Introduced by Jefferies et al. (2001), the method aims to increase

the model's resemblance to real stock markets, who often see effects from varying liquidity in the form of large swings in supply and demand as well as other specific patterns familiar to market traders (Jefferies et al., 2001). This model will however, unlike in Jefferies et al. (2001), not introduce wealth constraints for agents. All agents will be considered to have enough wealth for trading throughout Z time steps regardless of past losses. The decision of whether to trade or not depends on whether the success rate of the most successful strategy exceeds a certain threshold, τ. The success rate of an agent's most successful strategy in the last T time steps is denoted as

$$\mathrm{sr}(f_i^*) = \frac{1}{T} \sum_{\zeta=t-T}^{t-1} \mathbf{1}_{R^+}(\pi(f_i^*(\mu_\zeta), \mathrm{sign}(r_\zeta))) \tag{2.6}$$

where $\mathbf{1}_{R^+}$ is a function taking the value 1 when payoff π is positive and 0 when it's negative.

We can now fully express agent i's action at all times t in the following way

$$a_i^t(\mu_t) = \left\{ \begin{array}{ll} f_i^*(\mu_t) & \text{if } \mathrm{sr}(f_i^*) \geq \tau \\ 0 & \text{if } \mathrm{sr}(f_i^*) < \tau \end{array} \right\} \tag{2.7}$$

In short, the agent will do nothing if his best strategy at time t has a performance success rate lower than τ, otherwise he will trade according to the best trading strategy. As agents only need to predict the market's directional change, represented in binary, there is 50% chance of predicting correctly with luck only. An appropriate minimum value for the threshold, τ, could therefore be at 0.5, meaning that agents do not feel confident trading with strategies that perform worse than luck.

As already mentioned, agents use the market's past directional change as input information for trading. For that purpose, the actual return time series produced by the virtual market is reduced to directional changes. We have however yet to define how the result of all agents' trade actions translates into the return time series. This is achieved here by using a method from Farmer (2002) and Kyle (1985) for relating order flow and price formation.

Equation 2.7 shows that agent i's action at time t is $a_i^t(\mu_t)$, and we know that it is within the space $\{-1, 0, 1\}$. The virtual market's collective decision at each time t can therefore be stated as

$$A_t = \sum_{i=1}^{N} a_i^t(\mu_t), \tag{2.8}$$

where the value of A_t obviously lies in the range $[-N, N]$. Furthermore, $A_t > 0$ means the market edged up due to higher demand than supply,

while $A_t < 0$ means it edged down for the opposite reason. One might therefore refer to the value A_t as a net order flow at time t. The collective decision A_t is now transformed into return at time t with the linear relationship

$$r_t = \frac{A_t}{\lambda}, \tag{2.9}$$

where λ is the scale factor representing the size of order flow necessary to move the price by a certain amount (Farmer, 2002).

So far, we have introduced 5 parameters that dictate agents' behavior, N, m, s, T and τ. The goal will be to find the right values for these parameters as well as endowing agents with a set of strategies F_i such that the virtual market produces a time series that most closely matches the real time series we are trying to predict.

2.2 Agents' games

For attempting to capture some of the heterogeneity among agents in real financial markets, this model instills separate convictions into agents regarding how to trade. Specifically, the virtual agents are split into four groups with flexible sizes, who have fundamentally different views on how to behave in the market. The behavioral difference of groups manifests itself in the way they evaluate how their strategies pay off. When deriving the payoff from each trading strategy, agents rely on a specific function, which in turn varies depending on what group the agent belongs to. Each group of agents are given behavioral preferences according to the one specific game they are playing, which differs, to a varying extent, from the games the other groups play. The four games employed here are the Minority Game, the Delayed Minority Game, the Majority Game and the $-Game. Since we are using a parameter to control for liquidity in the market, τ, the four aforementioned games receive the addition "grand canonical" to their names (Jefferies et al., 2001; Wiesinger et al., 2012; Zhang et al., 2013). However, we will use the original names here for simplicity reasons, although we are referring to the grand canonical versions. Now we shall briefly explain the properties of each game and how they might relate to agents in real financial markets.

2.2.1 The Minority Game (MG)

Agents playing the Minority Game believe, analogously to the name, that making the same decision as the minority of all agents will bring them the most economic profit. Their preference is to buy a share when the majority

is selling, and sell when most are buying. This particular behavior mirrors that of a fundamental value investor, who seeks to buy undervalued stock and sell when overvalued. The payoff function characterizing the group of agents playing the MG is as follows

$$\pi^{MG}(f_i^j(\mu_t), \text{sign}(r_t)) = -f_i^j(\mu_t)\,\text{sign}(A_t). \tag{2.10}$$

Consequently, payoff is positive when the signs of an agent's trade decision $f_i^j(\mu_t)$ and the market's collective move A_t differ.

2.2.2 The Delayed Minority Game (dMG)

The Delayed Minority Game works in an essentially similar way to the Minority Game. Agents endowed with the dMG's conviction have, however, one important distinction. Unlike the traditional MG, they prefer making a decision now that will be in the minority in the following time step. So in a certain sense, their preference in the market is to try and predict when the stock will become undervalued. The payoff function describing this preference is

$$\pi^{dMG}(f_i^j(\mu_t), \text{sign}(r_t)) = -f_i^j(\mu_t), \text{sign}(A_{t+1}). \tag{2.11}$$

or simply, the sign of an agent's action at time t is preferably opposite of the collective decision's sign at time $t+1$.

2.2.3 The Majority Game (MajG)

The group of agents employing the Majority Game acts according to the conviction that making the same decision as a majority of agents at each time will be most beneficial. This belief among agents is thought to represent so called trend followers, who are common in real financial markets, as evidenced by current and past bubbles and crashes in asset prices. Their goal to simply buy the stock when the majority is doing it, under the assumption that prices must go up as other agents employ the same line of thought, completely disregards the underlying value of the asset. This common property of real markets, intuitively called herding in behavioral finance, already has rich academic literature. The payoff function we use here to model this preference is

$$\pi^{MajG}(f_i^j(\mu_t), \text{sign}(r_t)) = f_i^j(\mu_t)\,\text{sign}(A_t), \tag{2.12}$$

where agent i derives positive payoff when the sign of his decision at time t is the same as the market's collective decision.

2.2.4 The $-Game (DG)

The $-Game is an enhancement of the traditional Majority Game. The group of agents playing the $-Game prefer making a decision at time t that majority of agents will make at time $t+1$. Consequently, similar to agents playing the MajG, they believe trading trending stock will yield the highest economic profit. The defining difference with agents playing the MajG is however that $-Game agents try to predict the trend before it happens, making them enhanced trend followers. A payoff function that instills this behavior into agents is

$$\pi^{DG}(f_i^j(\mu_t), \text{sign}(r_t)) = f_i^j(\mu_t)\,\text{sign}(A_{t+1}). \tag{2.13}$$

Here, the payoff becomes positive when the sign from agent i's decision at time t, $f_i^j(\mu_t)$, is the same as the market's collective decision at time $t+1$, A_{t+1}.

2.3 Calibration with real time series

At this point, we have explained the inner workings of the ABM we will use in section 3 for identifying prediction indicators. In essence, the model employs heterogeneous agents, with essentially different intentions, who trade in a virtual market. We have also explained how their well defined behavior on the micro-level results in a dynamically complex time series of returns that possess some of the key statistical properties of real financial time series of returns.

As mentioned before, we will use this ABM for reverse engineering a real time series. The objective is to identify the sets of parameters and trading strategies for agents such that the virtual market produces a return time series that most closely resembles the real time series. Our hope is that the complex collective behavior of agents in the optimized model will go on to characterize the real returns in unoptimized time steps beyond those of the calibration.

The calibration itself involves evolving the set of parameters within defined ranges until a termination condition has been satisfied. The optimization here, designed by Zhang et al. (2013), is an extension of the approach of Wiesinger et al. (2012), who held the model's parameters fixed while optimizing over agent's sets of strategies. The extension involves optimizing over all characterizing parameters, including number of agents allocated to each game, general properties such as N, m, s, T and τ as well as each agent's strategy set. The intervals over which the general properties are optimized are as follows:

- Total number of agents in game, $N \in \{3, \ldots, 103\}$
- Length of agents' memory, $m \in \{2, \ldots, 8\}$
- Number of agents' strategies, $s \in \{1, \ldots, 16\}$
- Duration of period over which success rate of strategy is calculated, $T \in \{1, \ldots, 25\}$
- Minimum success rate of best strategy in order for agent to trade, $\tau \in [0, 1]$

The optimization problem now takes on the form of a minimization with the following objective function

$$\text{Minimize: } \sum_{t=0}^{W_{is}} (r_t^{\text{r}} - r_t^{\text{abm}})^2 \, . \tag{2.14}$$

The variable r_t^r denotes the real return from the time series we have chosen to reverse engineer, while r_t^{abm} denotes the return from the ABM's virtual marketplace. Moreover, the variable W_{is} denotes the length of the period over which we are minimizing the difference between the real return time series and ABM produced time series. The subscript "is", is short for "in-sample", which we will explain later.

What makes this minimization problem particularly hard to solve is the extreme number of variables in the model. Therefore, for finding a good solution to the problem, a simple genetic algorithm is perhaps the most appropriate way to explore the solution space. Nevertheless, although various properties of agents' behavior have been considerably constrained, the optimization problem remains especially expensive in a computational way. Adding onto that, the highly non-linear affect of parameters on the collective behavior of the market makes the solution space all the more difficult to explore, despite "naive" preferences of agents and constrictions on their behavior. Because of this, calibrating the ABM to real financial time series of considerable length is best done on ETH Zurich's supercomputer, Brutus, on which experiments where performed.

The reverse engineering process is designed such that the time series of real returns we want to reverse engineer is split into multiple segments. The ABM is then calibrated on each segment separately and allowed to run for a small time period after each optimization with the same parameter values and strategy sets. The segments where our ABM is calibrated by solving the minimization problem are called "in-sample", and are of length W_{is}. The following few time steps where the ABM runs with the same parameters unoptimized are called "out-of-sample", and are of length W_{os}. Furthermore, the length of in-sample and out-of-sample periods are fixed for each experiment

whereas both can vary between experiments. In-sample periods usually do not come straight after one another but have instead a smaller gap that is usually the size of W_{os}. When the gap equals W_{os}, the entire reverse engineering process yields an unbroken, non-overlapping out-of-sample return time series, whereas the in-sample periods will clearly overlap each other as the in-sample period is longer than the out-of-sample period. The reason for not using the same length for both in-sample and out-of-sample periods is that shifts in the market dynamic are frequent and the assumptions behind parameters, optimized for one in-sample window, might not hold for but a few more time steps. Since the purpose of gathering out-of-sample returns is to explore the predictive power of the model by comparing the returns to the real returns, out-of-sample windows must be small on the assumption that the model parameters are still valid. To summarize, a reverse engineering process for a real time series could be as follows. At time t, the ABM is calibrated onto real returns in an in-sample window, then let run for a couple of time steps out-of-sample, after which it starts again now potentially at $t + W_{os}$ and repeats this process throughout the time series we want to reverse engineer. The following algorithm perhaps gives a better overview of how the reverse engineering process could look like.

Algorithm 1: Reverse engineering

Result: Reverse engineers real time series

$t \leftarrow t_0$; `/* Initialize time */`

$gap \leftarrow gap_0$; `/* Gap between in-samples */`

while *(t is not at end)* **do**

  $ABM_{is} \leftarrow ABM_0$; `/* Initialize in-sample ABM */`

  $ABM_{is} \leftarrow Calibrate(ABM_{is}, r^r_{t:t+W_{is}})$; `/* Optimize ABM on real returns */`

  $ABM_{os} \leftarrow ABM_{is}[t + W_{is} : t + W_{is} + W_{os}]$; `/* Run Optimized ABM out-of-sample */`

  $t \leftarrow t + gap$; `/* Increment t with gap */`

end

Each experiment produces a certain number of in-sample return time series and the corresponding out-of-sample returns. Furthermore, the reverse engineering process can return more than just the best solution. In fact, for some experiments it is set to return the ten best solutions, later referred to as ensembles, that the genetic algorithm could find. This makes it possible

to measure the effectiveness of the optimizer by comparing the solutions.

In section 3 we will analyze the results from chosen experiments in search for meaningful statistics we can use for predicting real returns. For that purpose we will especially rely on the optimized return time series in in-samples, as well as the unoptimized returns in out-of-samples.

Chapter 3

ABM Predictions

3.1 Validating the ABM

Applying the ABM in section 2 for prediction purposes has yielded positive results as demonstrated by Zhang (2013), whose results from testing the ABM will be briefly outlined here. The first test to conduct was measuring the rate at which the model can predict successfully in out-of-sample windows. The success rate was compared to those of random strategies who choose to buy at each day with probability f_+, where f_+ is the fraction of up-moves in the real time series. One can therefore say that the strategies used for comparison have an advantage over completely random strategies who have no idea of the value of f_+. The time series subject to reverse engineering were the following seven:

- S&P500 from 1992 to 2001
- S&P500 from 2002 to 2011
- NASDAQ from 1992 to 2001
- NASDAQ from 2002 to 2011
- Dow Jones from 1982 to 1991
- Dow Jones from 1992 to 2001
- Dow Jones from 2002 to 2011

All time series extend over 10 years and each was split into 100 windows, making 700 windows in total. The results from this were clear. Firstly, they showed that only 6.6% of out-of-sample windows had p-values larger than 0.1. Secondly, in over 70% of out-of-sample windows [1] there were at most

[1] All windows except for those in the time series of Down Jones between years 1982 and 2001

5% of them who report worse prediction rates than random strategies for a confidence level of 90%.

Further validation of the ABM came from measuring the success of a trading strategy whose buy and sell decisions are dictated by the model. By taking an artificial position in the reverse engineered index according to the model's prediction of the next day's return signal, the trading strategy produces a time series of returns whose significance can be measured. Specifically, the trading strategy included entering a position in the index at the beginning of each day and closing it at the end. No transaction costs were included in the calculations. Results were then compared to those of random strategies, where buy and sell decisions are made with probabilities b and $1 - b$ respectively, using 100 uniformly distributed b's between 0 and 1. Comparison was done in two ways, the total profit or loss of strategies and the Sharpe ratio. Results showed that in 15% of periods, the strategy based on the ABM could create profit that only 10% of random strategies could. Likewise, 15% of periods for the ABM based trading strategy produced Sharpe ratios that only 10% of random strategies were able to. These results were then verified statistically with relevant p-values close to 0.

The final step in validating the ABMs relevance regarding predicting real returns included confirming the creation of abnormal risk-adjusted returns, or α's. For this, the returns from the ABM based trading strategy were regressed onto the three-factor model by Fama and French (Fama and French, 1993). The model expands the traditional capital asset pricing model (CAPM) to include 2 more factors that are thought to take part in describing returns. The risk premium according to the model is

$$r_{abm} - r_f = \alpha + b \cdot (r_m - r_f) + c \cdot SMB + d \cdot HML, \tag{3.1}$$

where r_f and r_m are the risk free and market risk rates respectively, SMB represents 'small minus big' companies in terms of market capitalization and HML represents 'high minus low' companies in terms of book-to-market ratios.

When returns from the ABM strategy are applied to equation 3.1 in addition to returns from 20.000 random strategies for each period, results show that merely 0.1% of random strategies are able to match the α's of 21 out of 700 periods for the ABM trading strategy. These results indicate that 21 out of 700 have positive significant α's. We refer to Zhang (2013) for a more detailed overview of results.

3.2 Prediction indicators

As indicated in the previous section, the ABM could potentially be a beneficial prediction tool for real financial markets. With the interplay of virtual agents who rely solely on information embedded in past returns, the model creates the dynamic and macroscopic behavior common in real markets. Moreover, it demonstrates statistically significant predictions and economical gains, which certainly undermines the weak form EMH. When testing the model with a simple trading strategy, positive and significant α's were realized in 21 out of 700 periods. This might not be a high ratio on its own, but it is impressive considering that it was produced by a trading strategy that traded every day according to the ABM's output, without incorporating any additional information. Such a simple trading strategy is useful for measuring the model's prediction capability but we think that a more successful strategy could be constructed by adding additional input factors for determining when to trade. An enhanced strategy based on the ABM's output would perhaps be more applicable in a real setting, as trades would be made more selectively and with regard to transaction costs. This is opposed to the simple strategy explained above whose transaction costs would undoubtedly have a considerable impact on economic gains from trading.

We suspect that the model's potential for predicting real markets is not yet fully discovered. In fact, there might exist undiscovered factors that influence the ABM's prediction rate, as reflected in the slight variation of prediction rates over different periods (see in Zhang (2013)). Discovering these influencing factors could potentially allow us to recognize when the model is more likely to predict correctly, leading to strategies with potentially higher risk-adjusted abnormal returns. In the following sub-chapters we will analyze the factors we think might improve the prediction rate. Our analyses are not meant to be exhaustive nor provide final conclusion as to what will improve trading strategies based on the model. They are more thought of as initial steps that could provide indicators ready for incorporating into a test of paper-based trading strategies. Analyses in the following sub chapters therefore do not attempt to verify nor prove the model's predictive power, as this has already been done in Zhang et al. (2013). They merely focus on identifying factors that could indicate when (and if) the model is about to predict correctly.

We will often analyze results from an experiment done on the NASDAQ index between years 2002 and 2011. This particular experiment is convenient for analysis as its out-of-sample period length is short, making for a large number of windows and therefore better statistics across windows. The NASDAQ is also thought to be a suitable index for reverse engineering because of its focus on growth stocks, whose investor behavior is considered to be speculative (Wiesinger et al., 2012). Occasionally, we'll also compare

our findings to those of experiments done on other indexes and in other time periods, as well as aggregate results from numerous experiments. The comparison of results between the NASDAQ and other reverse engineering experiments is interesting because different assets might have different characteristics, making some prediction indicators from this experiment possibly non-usable for trading other assets. To specify, the behavior of investors trading in one asset might be fundamentally different from investor behavior for another unrelated asset. This might result in different dynamics on the macro level, which in turn might result in different prediction indicators. We would therefore not be surprised if prediction indicators are slightly different between different indexes, if they exist to begin with.

Some informative properties of the experiment we'll use in the beginning:

- Name: NASDAQ-100 Index
- Begin date: 02.01.2002
- End date: 30.12.2011
- In-sample 1 & 2 lengths: 20
- Out-of-sample length: 2
- Ensembles: 10

All experiments have 2 in-sample periods. The first period is when agents train their strategies in order for them to choose the ones who have performed best. The second period is where the minimization of the difference in virtual returns and real returns takes place. Both periods have fixed lengths within the same experiment but they can vary between experiments. Further, they do not have to be of same lengths in the same experiment.

3.2.1 In-sample success rate

The first possible factor we want to examine is the success rate of the in-sample windows, both the training period (in-sample 1) and optimized period (in-sample 2). Specifically, we are curious to see whether there exists any relationship between the in-sample success rate of predictions and the corresponding out-of-sample success rate. For instance, if there was a positive relationship between them, we were more likely to trust the ABM's predicted return if the in-sample calibration has a high success rate. It would then also come intuitively that higher in-sample success rates would lead to higher out-of-sample success rates. We will test this assumption here and estimate its potential for enhancing trade decisions.

The optimized in-sample period is naturally expected to have a higher success rate than the unoptimized one. We define success rate of an in-sample

period to be the proportion of days the ABM predicts a correct return signal. Furthermore, as the ABM sometimes predicts returns of zero, we keep track of that and limit its effect on our in-sample success rate results. We do that by counting how often a zero prediction happens in one period and excluding its success rate from the results if the count equals or exceeds half of the days in that particular period. This is beneficial as the calculation for the fraction of successful prediction days in one period only extends over the days where the ABM's return prediction does not equal zero. If this was not controlled for, periods where large proportions of days have predictions of zero possibly result in abnormally high or low success rates. The method we use here for dealing with the ABM's zero predictions is of course not the only possible one. Others might include zero predictions in the in-sample success rate somehow, for instance by treating them consistently as either correct or false predictions. Nevertheless, since excluding periods with too many zero prediction days does not result in too few periods, we deemed best to deal with them in that way. We do however use a slightly different approach for calculating the success rate of out-of-sample periods. In those cases, we do not condition on the proportion of zero prediction days within each window. Instead we simply just consider days that don't have a zero prediction in our success rate. Windows with a high proportion of zero predictions in out-of-sample periods tend to also have that in the in-sample periods. As such, those particular windows are not considered anyway as the proportion of zero prediction days is controlled for in in-sample periods, causing those windows along with their corresponding out-of-sample windows to be disregarded. For consistency, this will be the way we will calculate in-sample and out-of-sample success rates throughout the section.

We'll visualize the results by creating a histogram of windows on in-sample success rates. We can then plot the average out-of-sample success rate for each pillar on top of the histogram. Additionally we will include a 95% confidence interval for identifying where the prediction rate is statistically larger than 50%. The confidence intervals are constructed with the Agresti-Coull method for binomial distributions, which works better for small sample sizes than using other normal approximations (Agresti and Coull, 1998). If n_1 is the number of successful predictions out of n tries where n is the number of windows multiplied with number of prediction days in each window, the estimate of proportion of success becomes

$$\tilde{p} = \frac{n_1 + \frac{1}{2}z^2_{\frac{\alpha}{2}}}{n + z^2_{\frac{\alpha}{2}}}, \tag{3.2}$$

where $z_{\frac{\alpha}{2}}$ is the $100(1 - \frac{1}{2}\alpha)$-th percentile of a standard normal distribution. We will furthermore be using $\frac{n_1}{n} = \frac{1}{2}$, as we would like to identify where the

true prediction rate is more likely to be better than chance. The confidence interval is then denoted

$$\tilde{p} \pm z_{\frac{\alpha}{2}} \sqrt{\frac{\tilde{p}(1-\tilde{p})}{n + z^2_{\frac{\alpha}{2}}}}. \tag{3.3}$$

The results in figures 3.1 and 3.2 are from the NASDAQ experiment whose details we described before. Additionally, the results are derived from the experiment's first ensemble[2] only. As we can see on figure 3.1, somewhat surprisingly, there does not seem to be a meaningful relationship between the success rates of in-sample 1 periods and out-of-sample periods. In other words, out-of-sample success rates do not strictly rise with in-sample 1 success rates as previously thought, although there might exist a slight overall upwards trend around the middle. Moreover, the average out-of-sample prediction rate is similar to the one for in-sample 1, or 50.2% and 50.4% respectively. That might seem strange but is understandable as in-sample 2 is only optimized. Another thing to make note of in figure 3.1 is that the out-of-sample success rate exceeds the upper confidence interval two times. We will ignore the ends in this figure as the confidence intervals do not work as well there because of extremely small sample sizes.

The 95% confidence intervals will contain the true value of the out-of-sample success rate with $\frac{n_1}{n} = \tilde{p} = \frac{1}{2}$ in 95 out of a 100 samples they're constructed from. Further, we can also say that the difference between the observed out-of-sample success rate and the true rate is statistically significant to the 5% level if the observed value lies outside of the confidence bounds. So indeed, we can say that the out-of-sample success rates that exceed the upper boundary are higher than 50%, statistically significant to the 5% level. For completeness, let us also check the probability that some of the observed out-of-sample success rates will fall outside the confidence bounds, given $\frac{n_1}{n} = \tilde{p} = \frac{1}{2}$. That way we can be more confident that the prediction rates we are seeing are not just a part of those 5% of observations that fall outside the interval.

An observation can either be in the interval or not, with probability of being in it $p_{within} = 95\%$. Let's also assume that the 5% of observations that lie outside the interval are distributed equally, with 2.5% above the interval and the same below. Let's therefore define the probability of an observation being below the upper 95% confidence line as $p_{in} = 97.5\%$. The probability that k out of n_{bin} observations are below it is therefore:

[2]Ensembles represent different calibration solutions. The first ensemble is the best solution.

$$P(X = k) = \binom{n_{bin}}{k} \cdot p_{in}^{k} \cdot (1 - p_{in})^{(n_{bin}-k)}. \tag{3.4}$$

So if all our observed out-of-sample predictions are indeed samples from a binomial distribution with probability of success 0.5, the probability of at least one of them lying above the interval according to equation 3.4 is

$$1 - P(X = 13) = 1 - 0.975^{13} = 28\%. \tag{3.5}$$

We only use 13 out of the 18 intervals on figure 3.1 as the confidence intervals are not as reliable for the 5 intervals with the lowest window count.

With a probability of 28%, we are fairly likely to see at least one out-of-sample success rate above the interval when $\tilde{p} = 0.5$. However, in figure 3.1, we have two success rates above the interval. The probability of observing $n_{bin} - k$ or more out of n_{bin} success rates above the interval is

$$P(X \leq k) = \sum_{i=0}^{k} \binom{n_{bin}}{i} \cdot p_{in}^{i} \cdot (1 - p_{in})^{(n_{bin}-i)}. \tag{3.6}$$

When we use equation 3.6 with $n_{bin} - k = 2$ like in figure 3.1, we get $P(X \leq 11) = 4.06\%$. The event would therefore be very improbable, thus strengthening our belief that the actual out-of-sample success rates are indeed higher than 50%, at least for these two in-sample 1 success rate pillars.

Similarly, in figure 3.2, it is very hard to extract any kind of relationship between in-sample 2 and out-of-sample success rates. There doesn't seem to be a clear trend between the two, which indicates, in contrast to our assumption beforehand, that more successful calibration doesn't necessarily result in more out-of-sample prediction power. In fact, if anything, figure 3.2 shows a slightly decreasing relationship between in-sample 2 and out-of-sample success rates. We can also see that there is a distinct difference in average success rates of the in-samples, as in-sample 2 boasts an average of 78.6% compared to 50.4%. This was expected as the model was calibrated with the index on in-sample 2 periods. Additionally, what these pictures do not show is that 13 windows were eliminated from figure 3.1 and 67 windows from figure 3.2 due to a large proportion of prediction days in these windows having returns of 0. The reason for the big difference in number of windows eliminated is not clear, but might have to do with only one in-sample being optimized and not the other. What is also apparent on figure 3.2 is that out-of-sample prediction rates hardly ever cross the confidence interval. Except for the interval 50-55%, whose respective out-of-sample success rate is 100%. It does however only have three windows,

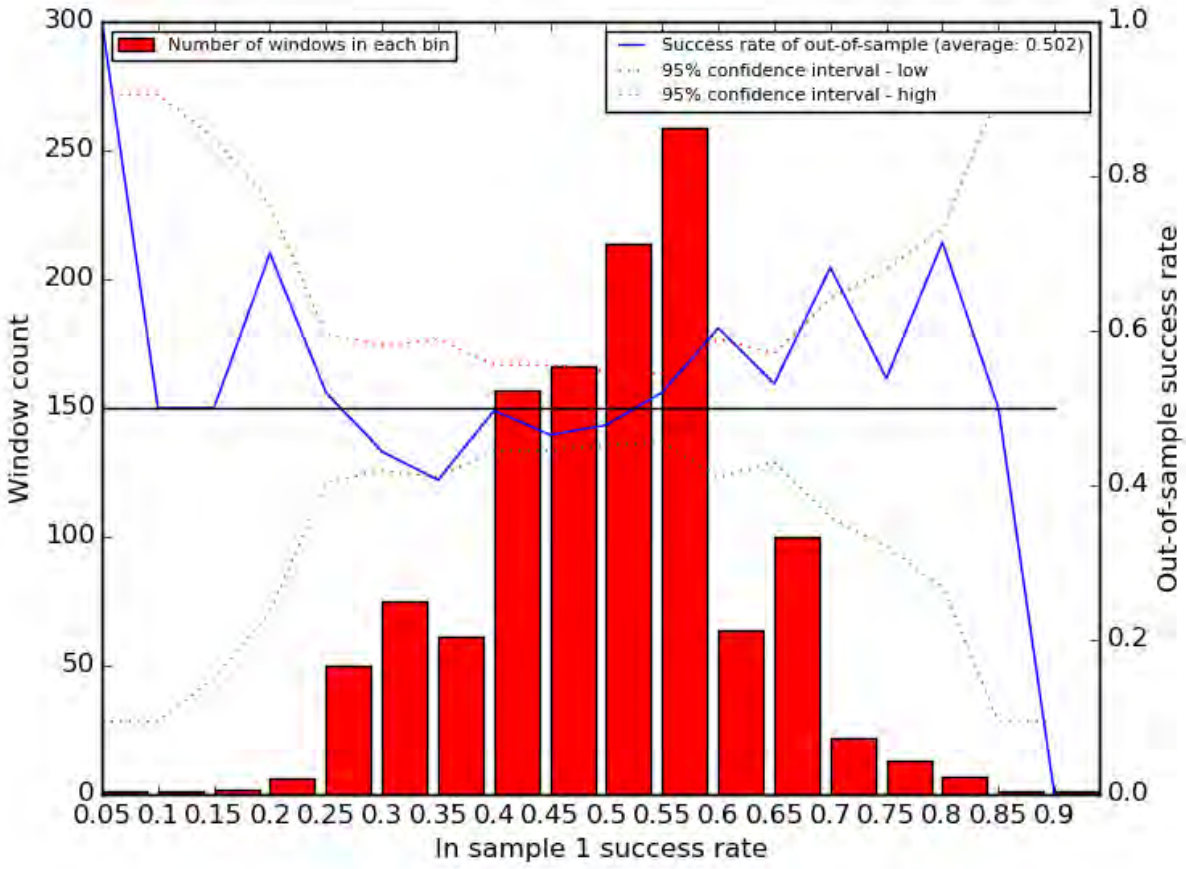

Figure 3.1. Analysis of results from one experiment on the NASDAQ in 2002-2011. Histogram of success rates from in-sample 1 periods with corresponding out-of-sample success rates plotted on top. A 95% confidence interval for binomial distributions with Agresti-Coull method is also displayed.

which certainly makes it less reliable. But if we estimate the probability of a 100% success rate in $n = 3 \cdot 2 = 6$ observations with $\tilde{p} = 0.5$ with equation 3.4, we get $2^{-6} \approx 1.56\%$. The small probability gives us some assurance that this interval's out-of-sample success rate is actually higher than 50%.

For comparison we will now compile results from 99 similar experiments with the NASDAQ for the ten years between 2002-2011. Each experiment has a different composition of in-sample 1 and in-sample 2 lengths although they all have the same out-of-sample length as before, namely 2. Only results from experiments' first ensemble (best solution) are compiled. By an-

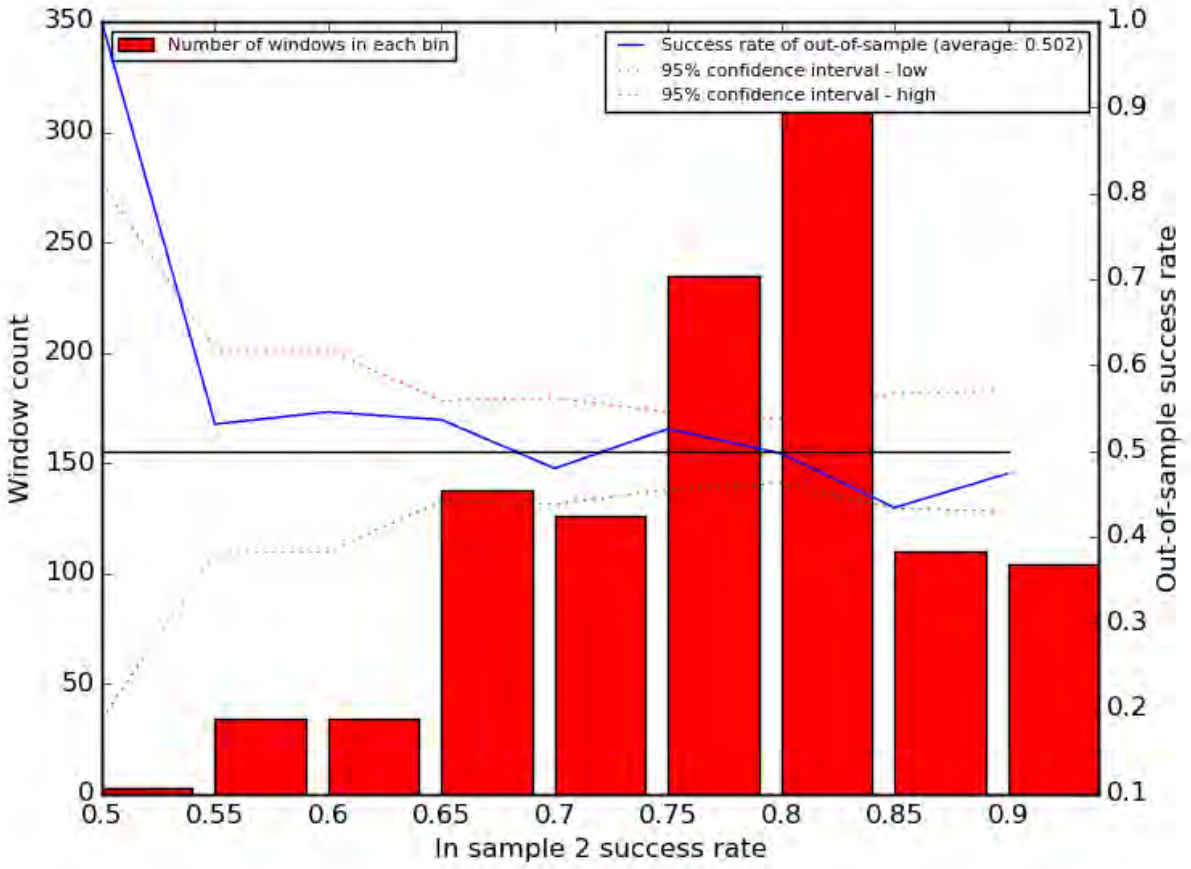

Figure 3.2. Analysis of results from one experiment on the NASDAQ in 2002-2011. Histogram of success rates from in-sample 2 periods with corresponding out-of-sample success rates plotted on top. A 95% confidence interval for binomial distributions with Agresti-Coull method is also displayed.

alyzing results from numerous NASDAQ experiments for the same time period and with the same out-of-sample length, we hope to get more reliable statistics which we can directly compare with our earlier analysis. With the power of more observations we might be able to confirm or refute our previous findings. Although there are various time periods for which we have results from experiments with the NASDAQ, the model needs to have ”operated” on the same market dynamics for our analysis here to be comparable. Calibrating it on different time periods and thus different kinds of market regimes would essentially yield results whose in-sample vs out-of-sample success rate analysis would maybe not be comparable to our earlier

analysis. Similarly, only aggregating results from experiments with out-of-sample length 2 is more dependable when we do not know if there are effects of varying out-of-sample lengths.

As mentioned, the lengths of the second and first in-sample periods will vary between experiments, which is alright here as this analysis focuses only on revealing the relationship between success rates of in-sample and out-of-sample periods. We will check for effects of varying in-sample lengths in subsequent analyses. It is also worth mentioning that the same method as before for eliminating windows with a high proportion of 0 prediction days was used here. This might affect experiments whose in-sample lengths are very short. Our previous experiment on the NASDAQ in 2002-2011 had both in-sample lengths of 20, whereas all of the 99 experiments used here have a different combination of lower in-sample lengths than before. The effects of this might show in a higher number of pillars for in-sample windows with extremely high or low success rates. The lengths of in-sample periods in the 99 experiments we'll use, l_{is1} and l_{is2}, are in the space $l_{is1}, l_{is2} \in \{20, 18, 16, 14, 12, 10, 8, 6, 4, 2\}$. With all possible alternatives our total number of experiments sums up to $10 \cdot 10 - 1 = 99$, where we leave out the alternative $\{l_{is1}, l_{is2}\} = \{20, 20\}$ which is the subject of our earlier analysis.

Results for in-sample 1 vs out-of-sample success rates are displayed in figure 3.3. As we can see, the average success rate of out-of-sample periods is 50%. Additionally, our suspicion regarding increased number of windows in extreme in-sample success rates seems to be validated here, as the window count at the ends seems large in proportion to the rest. We can assume that a large number of these windows have a very small length, which increases the probability of high window success rate. Figure 3.3 does not seem to display a clear trend between in-sample 1 success rates and out-of-sample success rates. However, the light blue line of best fit included in this figure does have a very slight upwards slope. Specifically, the line of best fit would predict a 1.25% increase in out-of-sample prediction rates when in-sample success rate changes from 0 to 100%. It is therefore safe to say that if a distinct positive relationship exists between these parameters, it is to a small extent. In light of these results we can assume that higher in-sample 1 success rates are at best associated with higher out-of-sample success rates to a small degree. Since in-sample 1 periods are when agents train their strategies in order to determine which ones to use, this would indicate that it doesn't matter much how closely the virtual market mimics the real market during that process because it has little implication to prediction power. We can also see that the out-of-sample prediction rate is now always within the 95% confidence interval.

When comparing figures 3.3 and 3.1, we can see that both distributions of in-sample 1 success rates are similar in shape, with an expected value of

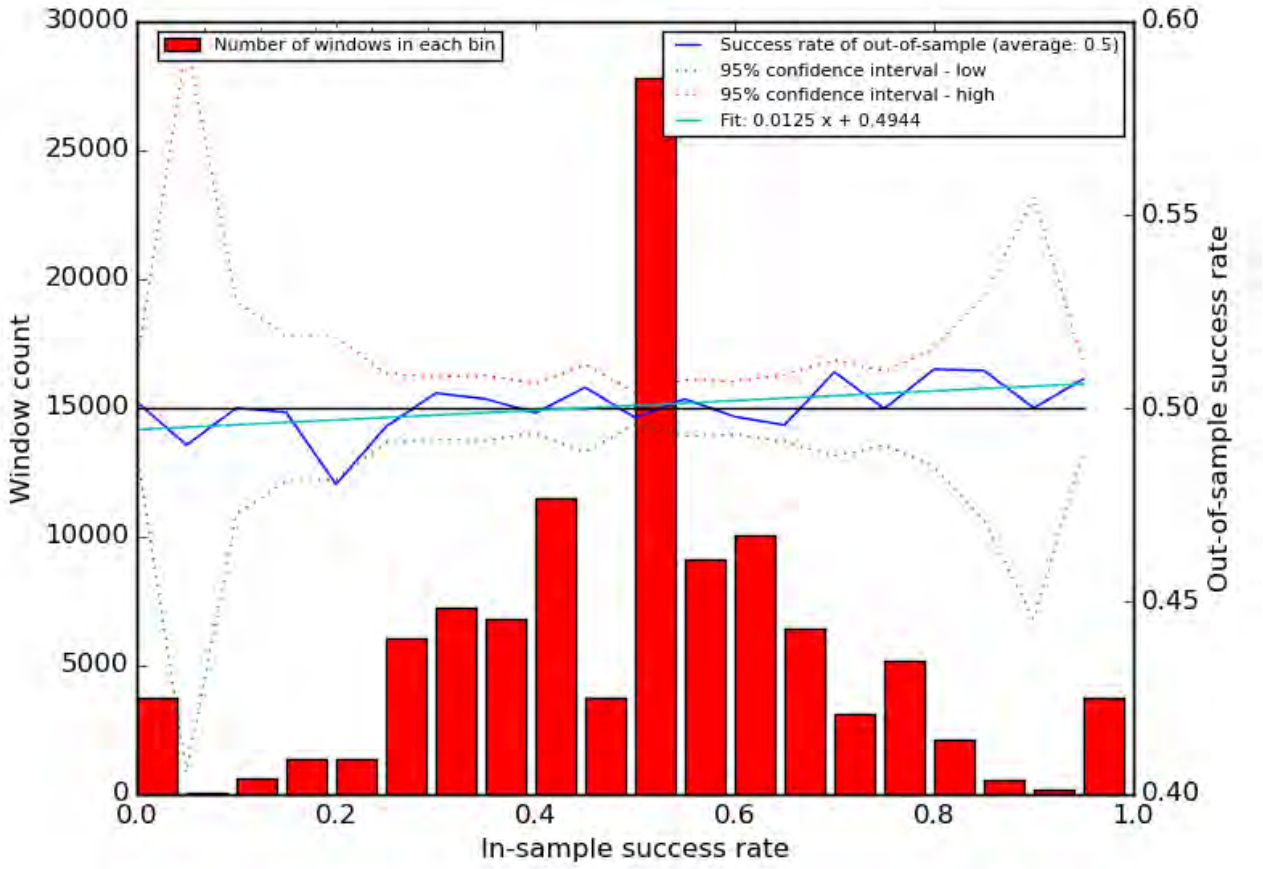

Figure 3.3. Histogram of success rates from in-sample 1 periods with corresponding out-of-sample success rates plotted on top. A 95% confidence interval for binomial distributions with Agresti-Coull method is also displayed. The results are from 99 experiments on the NASDAQ in the period 2002-2011. In-sample 1 and in-sample 2 period lengths vary but out-of-sample period is always of length 2.

around 0.5. Although there doesn't seem to be a strictly rising relationship between the success rates in either figure, they do indicate a slight upwards trend. We also note that average success rates for both in-sample 1 and out-of-sample are very similar for both analyses. Further, the spikes in the out-of-sample success rate in figure 3.1 at the intervals 70-75% and 80-85% seem to be also in figure 3.3. That does serve to confirm increased out-of-sample success rates for those intervals. They are nevertheless still within the 95% confidence interval, making them not statistically different from 50% to the 5% level.

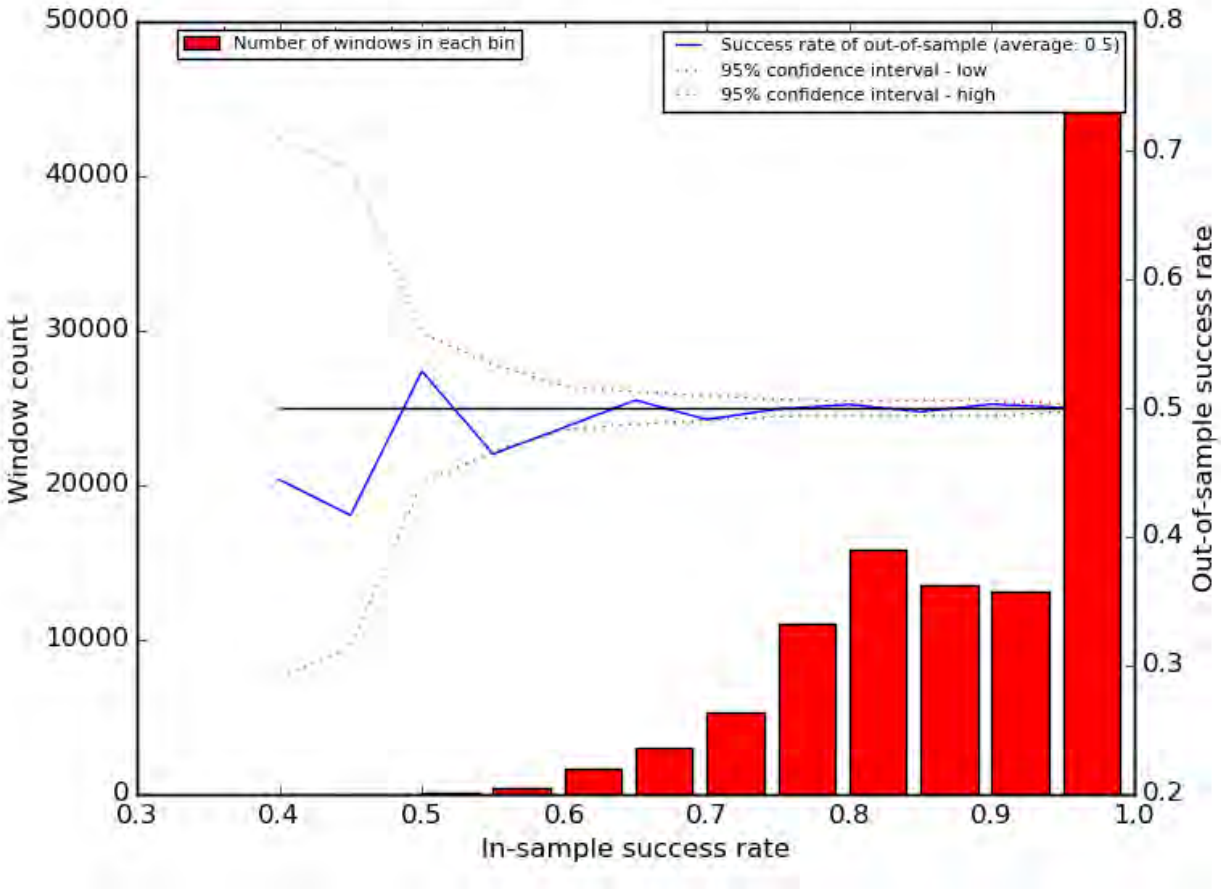

Figure 3.4. Histogram of success rates from in-sample 2 periods with corresponding out-of-sample success rates plotted on top. A 95% confidence interval for binomial distributions with Agresti-Coull method is also displayed. The results are from 99 experiments on the NASDAQ in the period 2002-2011. In-sample 1 and in-sample 2 period lengths vary but out-of-sample period is always of length 2.

Figure 3.4 shows our analysis of results from all 99 experiments for in-sample 2 against out-of-sample. Now, a considerable portion of all the windows falls into the highest interval, 95-100%. The large number of windows at that level is undoubtedly partly the result of small in-sample 2 lengths. Even so, it is clear that the genetic algorithm does a good job of calibrating the in-sample 2 period to the real return series. Similar to our analysis of in-sample 1 periods, there seems to be neither positive nor negative relationship between in-sample 1 and out-of-sample success rates. If we look closer at out-of-sample success rates for intervals of in-sample 2 success rates of

65% or higher, the line seems to be completely flat. It is interesting to note that it doesn't seem to matter much, or to a negligible extent, how good the genetic algorithm is at optimizing the model for in-sample 2 periods, at least after achieving 65% success rate. The out-of-sample prediction capability will not change drastically. At least we have seen, according to figures 3.3 and 3.4, that that is the case on average for the NASDAQ in this time period with out-of-sample period length of 2 and varying in-sample period lengths. Furthermore, these results obviously only hold for the data we have, that is, for in-sample success rates on these particular intervals.

Some additional things to note are regarding difference in figures 3.2 and 3.4. Again, with the power of more samples in figure 3.4, we cannot confirm that average out-of-sample success rates are statistically higher than 50%. The figures' average out-of-sample success rates are also very similar, unlike average in-sample success rates, who clearly differ by more. The reason for that is of course the one we described before. The genetic algorithm is more likely to find a good solution as the period length shrinks, which is variable in figure 3.4. There are simply fewer days to optimize for shorter in-sample 2 periods, making for a more straightforward optimization. When we check this factually by calculating the average over all in-sample 2 periods of length 2, we get an almost 100% success rate, while we only get a roughly 80% success rate for period length of 20.

We know now that in-sample period length has a direct influence on the success rate of predictions in in-samples. Yet to explore, however, is its effect on out-of-sample success rates. It would be interesting to see if altering the lengths of periods in which agents train their strategies and the model is optimized has an effect on out-of-sample success rate of predictions. We would suspect, intuitively, that predictions would be better if the model had longer periods for the aforementioned tasks. By "preparing" the model on shorter periods, it will not have dealt with as much of the recent market dynamic that might be apparent in the out-of-sample. Let us now examine the effect of varying in-sample period lengths using all 100 experiments on the NASDAQ between 2002 and 2011. We will display results in figure 3.5, where each point reports the average out-of-sample success rate for the indicated in-sample period length. Further, as we know, all 100 experiments have out-of-sample lengths 2 and each point is an average over all windows in experiments who vary in the other in-sample length between 2 and 20. This gives us an average for over 12,000 windows in each point.

In figure 3.5, red points represent out-of-sample success rates for fixed in-sample 1 lengths and blue represents them for fixed in-sample 2 lengths. We can immediately see, with help from lines of best fit through the points, that lengths for in-sample 2 periods do not seem to have any effect on out-of-sample success rates. The line of best fit through in-sample 1 points does

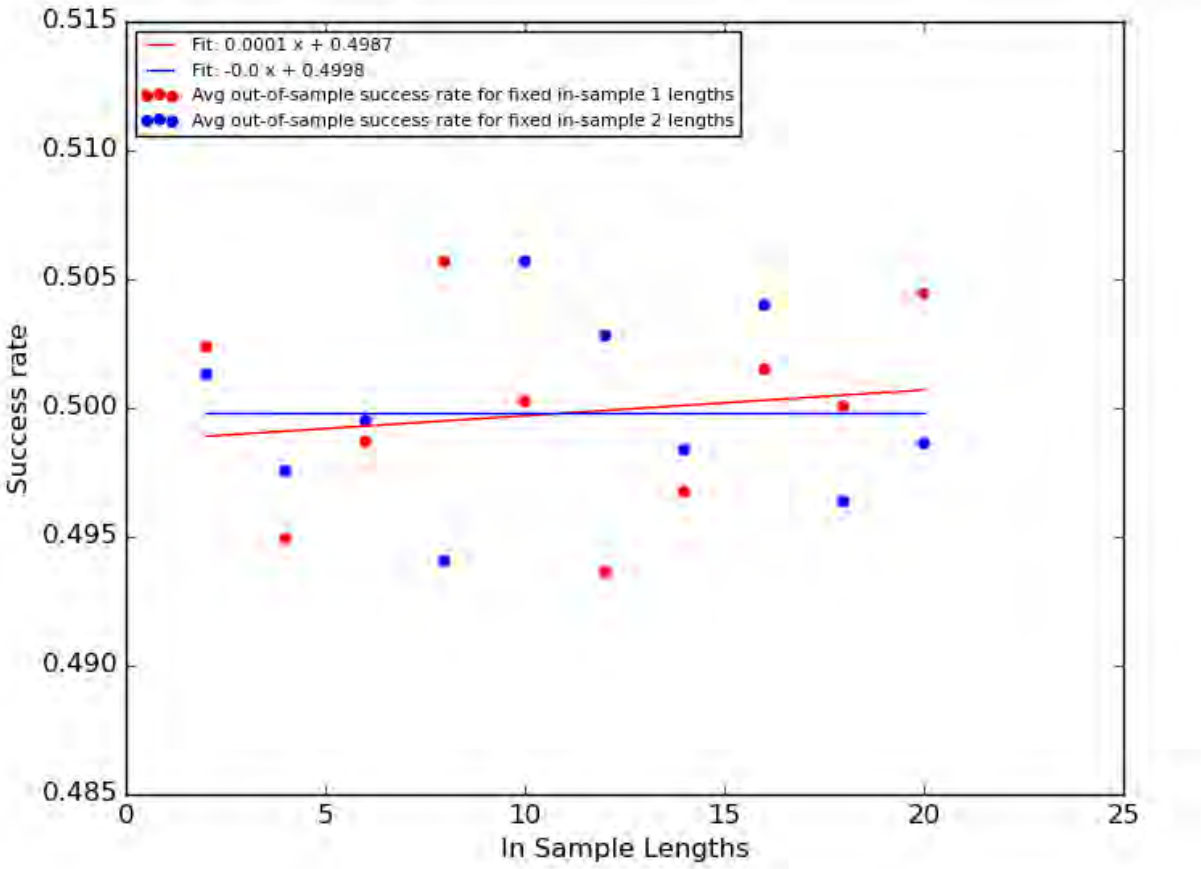

Figure 3.5. Scatter plot with average out-of-sample success rates for various fixed in-sample lengths for both in-sample 1 and 2. Data is from 100 experiments on the NASDAQ in 2002-2011. Out-of-sample length is always 2 and non-fixed in-sample length varies between 2 and 20. A line of best fit is added for better visualization.

have a very slight upwards slope, but judging from the distribution of points in the figure, this slope is far from statistically significant. This therefore tells us that varying in-sample lengths should not affect average out-of-sample success rate, or to a negligible degree.

At this point we have established that there is a very small positive relationship, if any, between in-sample success rates and out-of-sample success rates for the NASDAQ between 2002 and 2011. We can still only make this assumption for experiments in that index for that time period as we do not know if results would be different for other indexes or time periods. As

we explained before, the underlying investor behavior of other time periods or indexes might create market dynamics such that the ABM's results are exploitable to a larger or lesser degree. This is of course only a suspicion, which we are nevertheless inclined to resolve with the result data we have from numerous experiments on other indexes and time periods. For that purpose we will examine all experiments on indexes for which we have results for more than one time period. Specifically the 3 indexes we will compare over 2 time periods are the following:

- NASDAQ for the time period 1992-2001
- NASDAQ for the time period 2002-2011
- S&P 500 for the time period 1992-2001
- S&P 500 for the time period 2002-2011
- Dow Jones Industrial Average between 1992-2001
- Dow Jones Industrial Average between 2002-2011

We will display results in a similar manner as in figures 3.3 and 3.4, except now there is no histogram of in-sample success rates below the line relating in-sample success rates to average out-of-sample success rates. Instead there will only be out-of-sample success rate lines for each of the 6 categories, displayed in figures 3.6 and 3.7 for in-sample 1 and 2 success rates respectively. Moreover, each line will now contain averages from 100 experiments.

Figure 3.6 does show some variety in how in-sample 1 and out-of-sample success rates relate to each other. For instance, the purple and red lines, denoting respectively the Dow Jones and S&P500 in 1992-2001, seem to have an upwards slope, whereas inferring anything about other indexes and time periods is trickier. Therefore this does indicate that our suspicion regarding different results for different indexes and time periods is validated, although it certainly isn't very clear. Moreover, we can see that average success rates for these indexes and time periods differ, with the most successful index and period for predicting the market being the S&P500 in 1992-2001 with a 51.5% average success rate. Likewise the least successful one is the NASDAQ in 2002-2011 which, in a way, goes against our initial assumption that the NASDAQ would be most suitable for reverse-engineering. We must however also note that the results from experiments on the NASDAQ in 2002-2011 are expected to be somewhat different from others as they all have out-of-sample period length of 2. Experiments belonging to the other five categories all have an out-of-sample length of 16. The extent to which this contributes to a lower average out-of-sample success rate for the NASDAQ is however not known, if any.

Figure 3.7 relates in-sample 2 and out-of-sample success rates. Here, just as in figure 3.6, there does not seem to be consistency regarding out-of-sample

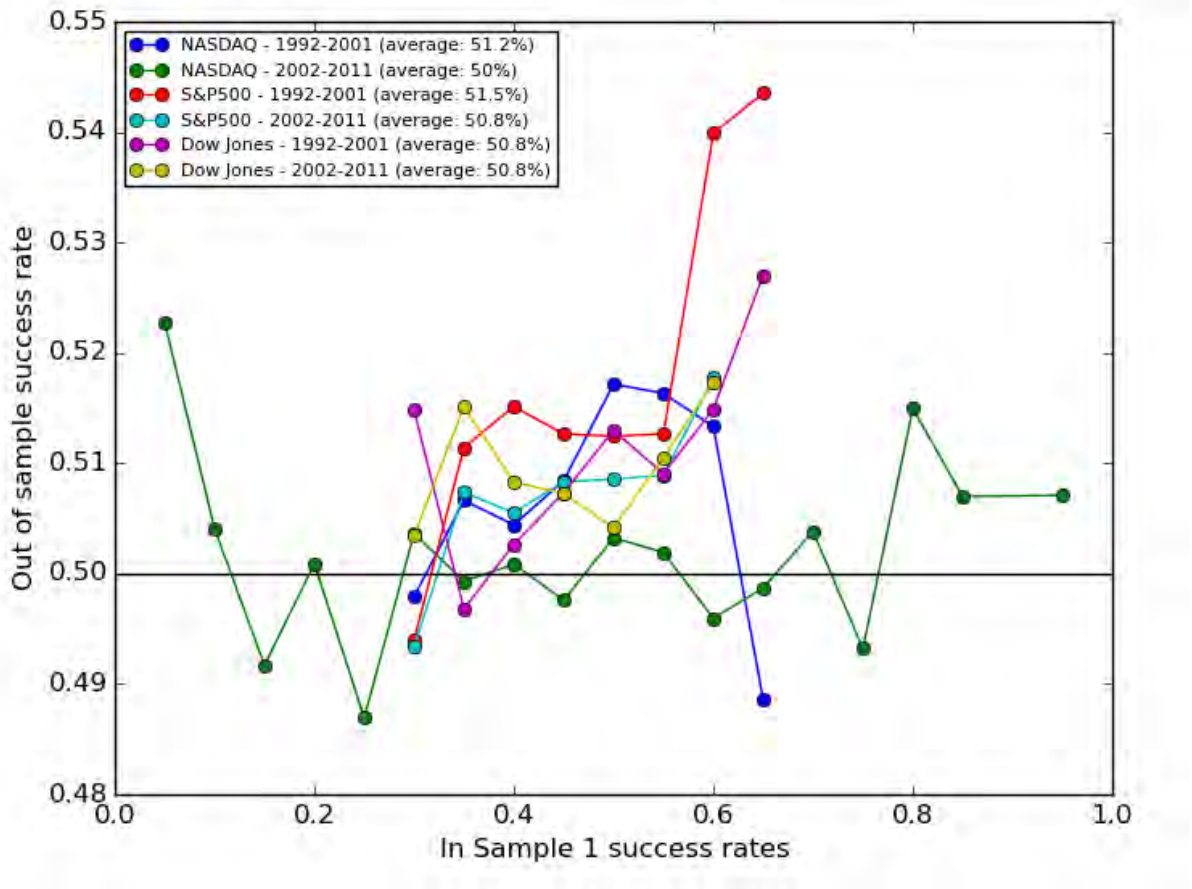

Figure 3.6. In-sample 1 vs Out-of-sample success rates for 3 indexes in 2 different time periods. Each line includes analysis of windows from 100 or more experiments. For limiting uncertainty, points on the graph are only displayed if they have averages from over 40 windows. This is the reason for varying lengths of lines.

success rates across indexes and time periods. One line could be on a decline where another is rising and vice versa. Another thing to note is that no line shows constant rising or falling out-of-sample success rates. It would therefore appear that in-sample 2 success rates also have an ambiguous effect on prediction capabilities of other indexes in other time periods.

One thing is clear according to figures 3.6 and 3.7. Experiments on the S&P 500 index in 1992-2001 seem to show the most positive response to higher in-sample 1 success rates. Let's therefore look a bit closer at the S&P 500 for this time period and attempt to pinpoint correlations beyond

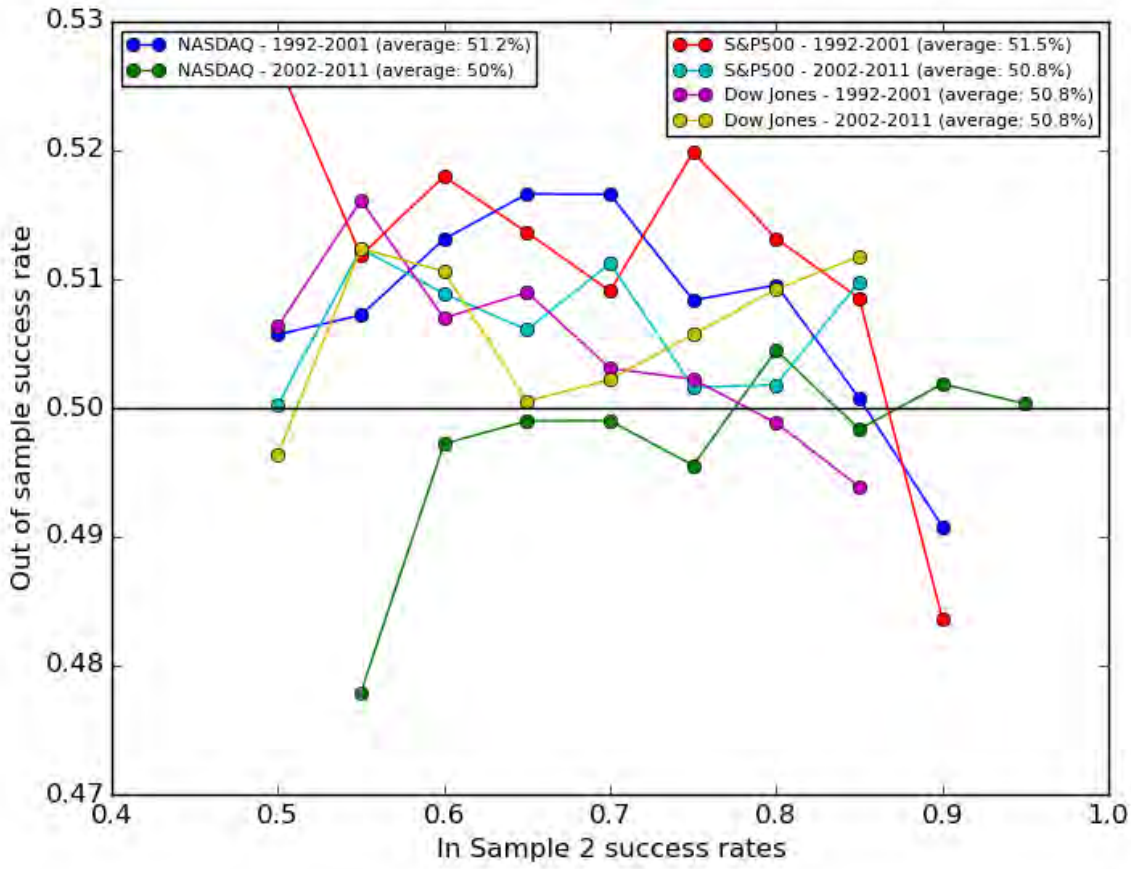

Figure 3.7. In-sample 2 vs Out-of-sample success rates for 3 indexes in 2 different time periods. Each line includes analysis of windows from 100 or more experiments. For limiting uncertainty, points on the graph are only displayed if they have averages from over 40 windows. This is the reason for varying lengths of lines.

what the NASDAQ has demonstrated. The most straightforward way to achieve that is to gather results from all 100 experiments on the index for all possible in-sample length combinations. We are essentially recreating the red lines in figures 3.6 and 3.7, however this time with confidence intervals and histograms. Results are in figures 3.8 and 3.9.

The more detailed plot in figure 3.8 reveals what seems to be a slightly upwards trending relationship between out-of-sample and in-sample 1 success rates throughout intervals 30-58%. This is followed by a large increase from 58-63%. Although the line is not monotonically increasing we can be fairly

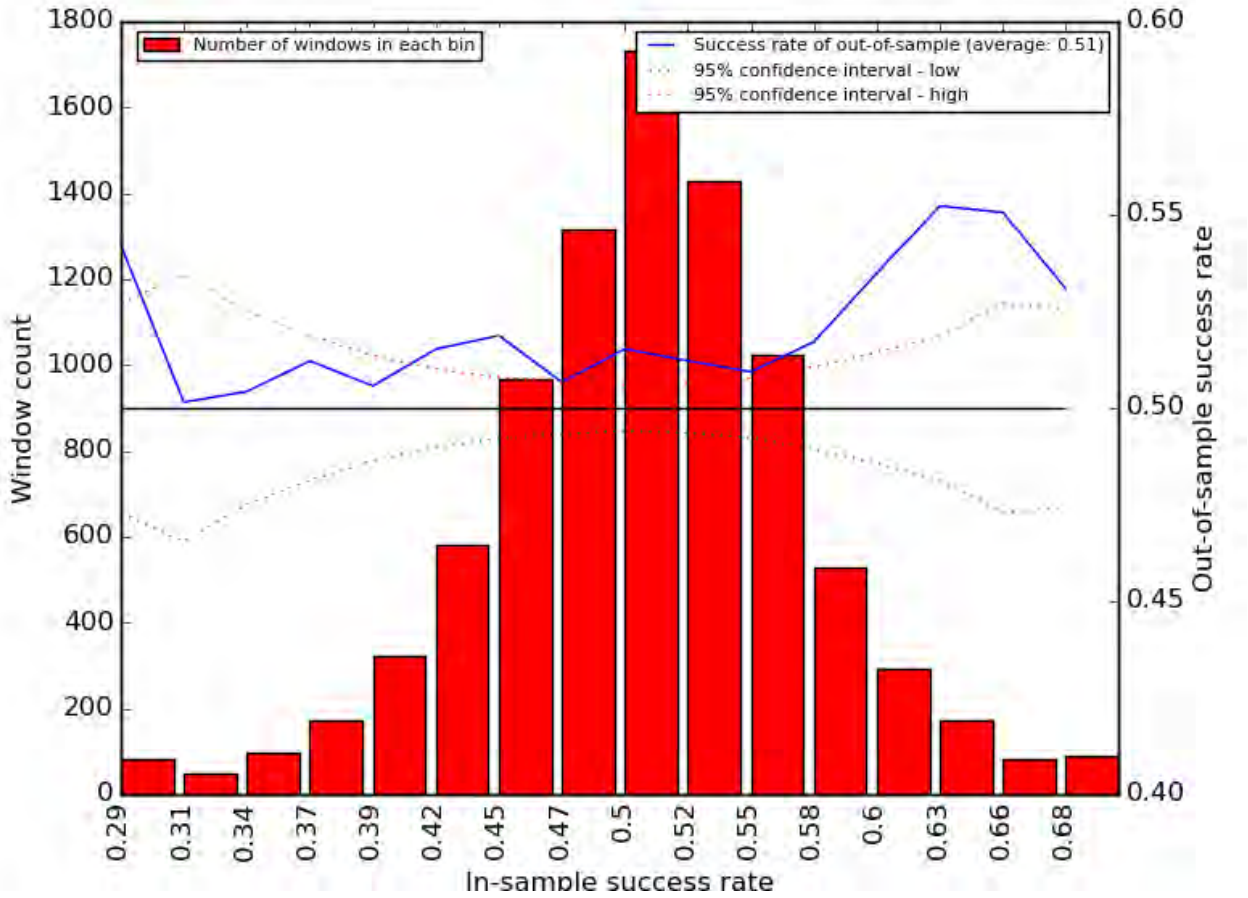

Figure 3.8. In-sample 1 vs Out-of-sample success rates for 100 experiments on the S&P 500 in 1992-2001. A histogram of sample sizes and a 95% confidence interval are included.

certain that better in-sample 1 success rates play a role for prediction power here. Furthermore, in contrast with the NASDAQ, most success rates fall above the 95% confidence interval. This is also the case in figure 3.9, which however does not seem to reveal a positive relationship between in-sample 2 and out-of-sample success rates. It is also worth noting here that the difference in average out-of-sample success rates between these figures and figures 3.6 and 3.7 is due to the way windows are filtered from the results. The filtering was a bit more stringent in the former two figures resulting in the difference we observe here.

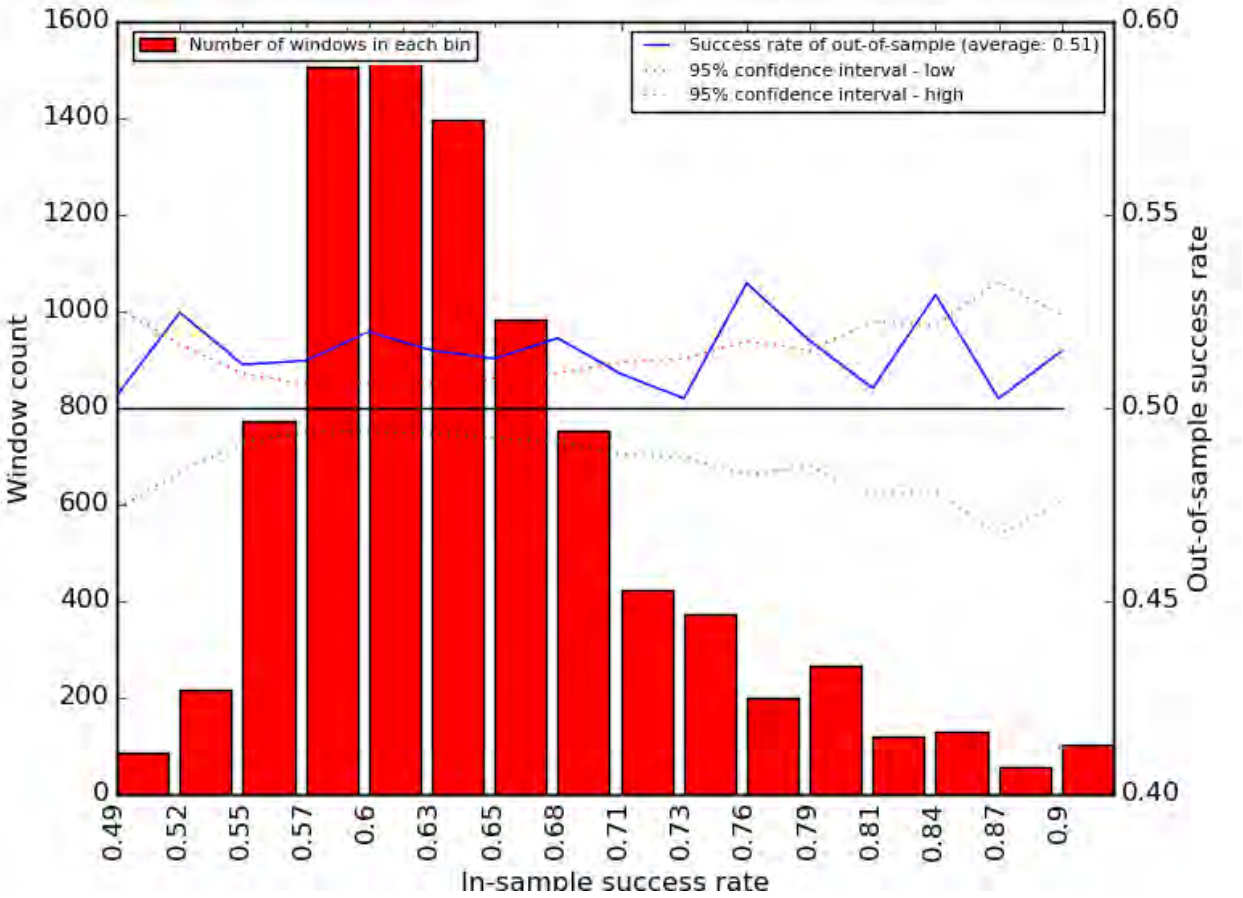

Figure 3.9. In-sample 2 vs Out-of-sample success rates for 100 experiments on the S&P 500 in 1992-2001. A histogram of sample sizes and a 95% confidence interval are included.

3.2.2 Predicted and real returns

Other factors to be examined are both predicted and real returns. Specifically, we would like to see if the model is more likely to predict correctly for some values of predicted or real returns and less for others. For example, can we expect the model to be more often right when it predicts returns higher than zero? Or alternatively, do we get better predictions when the absolute of predicted returns is low? If we were to reveal a relationship between the model's predicted or real returns and success rate of predictions, we could be more sure of its output when the corresponding returns match our criteria. Information of this kind could therefore potentially be used for enhancing a trading strategy based on ABM predictions. For consistency, we

will be analyzing the experiment based on the NASDAQ in 2002-2011 and comparing to chosen experiments on either different indexes or different time periods.

We will start by looking for a relation between out-of-sample success rate of predictions and predicted returns in out-of-sample days. Results are displayed in figure 3.10 along with a 95% Agresti-Coull confidence interval.

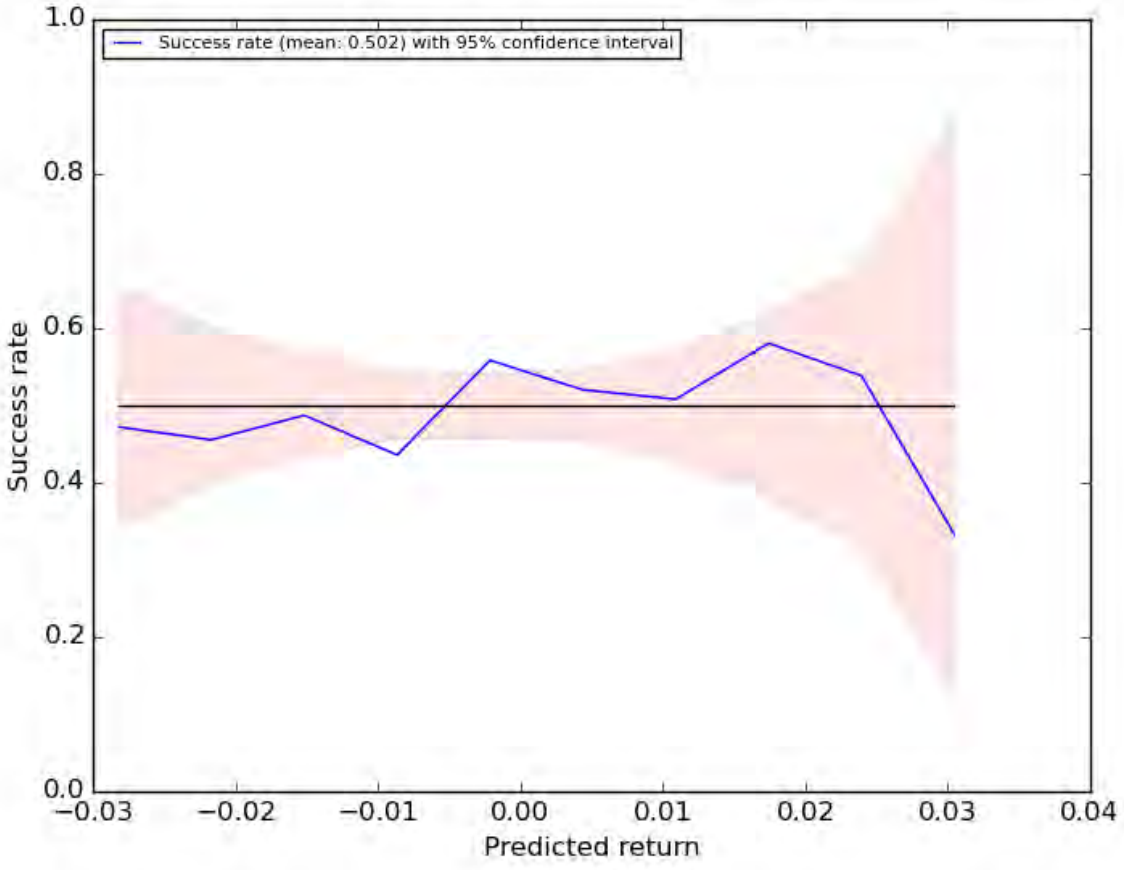

Figure 3.10. Out-of-sample success rate of predictions for each predicted return. The data is from an experiment on the NASDAQ in 2002-2011. A 95% confidence interval is plotted on top.

Figure 3.10 does not show an explicitly rising or falling relationship between the success rate and predicted return value. The success rate does however seem to increase considerable right before zero predicted returns. It therefore looks like the model is more likely to give a correct prediction when

predicted returns are on the interval -0.5-2.5%. This is at least true for this index in this time period. Another thing to note is that success rates only manage to exceed the upper 95% confidence level once, indicating that success rates for other values of predicted returns are not statistically higher than 50% to the 5% level.

Here, we are inclined to continue our earlier assumption that different indexes might give different results. Therefore, let us compare this to other indexes and time periods for which we have experiment data. Specifically we'll chose 5 experiments, all with the same lengths for in-sample 1, 2 and out-of-sample, or 160, 160 and 16 respectively. This way they'll be more comparable and have the same proportion of in-sample lengths to out-of-sample length as our previous results who had in-sample 1, 2 and out-of-sample lengths of 20, 20 and 2 respectively. The indexes and their corresponding time periods are as follows:

- NASDAQ between 1992-2001
- S&P 500 between 1992-2001
- S&P 500 between 2002-2011
- Dow Jones between 1992-2001
- Dow Jones between 2002-2011

Results for the same analysis of these indexes are compressed and displayed in figure 3.11.

Interesting to note is that the relationship between success rates and predicted returns for the NASDAQ in 1992-2001 takes on a very similar shape as it does for the same index in 2002-2011, with a rise around -0.5%. In fact, we can see that there are similarities between same indexes for different time periods. Moreover, what they all have in common is that the success rate is somewhat more likely to be above 50% if predicted returns are above zero. This is especially true for the NASDAQ and S&P 500 where difference in out-of-sample success rate is clear depending on what return the model is predicting. This would certainly indicate that a trading strategy relying on the ABM's predictions could be enhanced by weighing trading decisions according to the value of the model's predicted return, at least for trading in the NASDAQ and S&P 500.

Real returns are another potentially revealing indicator. However, looking at real returns during the out-of-sample period won't be very helpful, so we will focus on identifying a relationship between real returns during the in-sample 2 period on the one hand and out-of-sample success rates on the other. We'll start by examining volatility during the in-sample 2 period and how that connects with out-of-sample success rates. This is done by

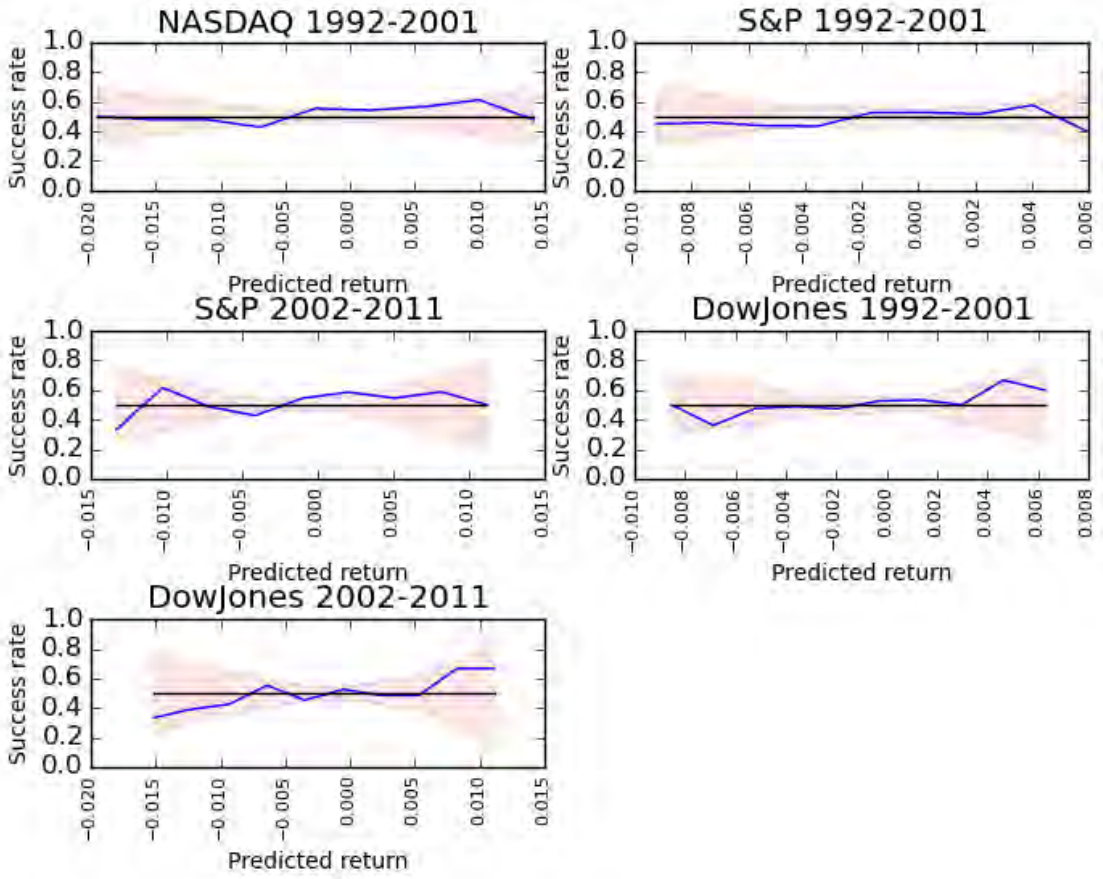

Figure 3.11. Out-of-sample success rate of predictions for each predicted return. Each picture analyzes a different experiment. The 5 experiments have a different combination of the subject index and time period. A 95% confidence interval is plotted on top.

measuring the volatility of real returns in each in-sample 2 window and the success rate of predictions of the corresponding out-of-sample window. Figure 3.12 shows the same three indexes as before for the two time periods of 1992-2001 and 2002-2011. We define volatility as the corrected sample standard deviation of each in-sample 2 window of length n as

$$s^2 = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} x_i - \bar{x}_i}. \tag{3.7}$$

Our results in figure 3.12 indicate no meaningful relationships between in-

sample 2 volatility and out-of-sample success rate of predictions, regardless of index or time period. Interestingly, there is however a significant drop in success rates around volatility of 0.025 for the NASDAQ in both time periods. Nevertheless, this drop does not seem to be a part in an overall decline in success rates with volatility, so it could very well be coincidental. Very large spikes in success rates can most probably be ignored here as they are likely the result of a very small sample size. Overall, this would suggest that the model performs just as good in observed turbulent in-sample 2 periods as it does in calmer ones.

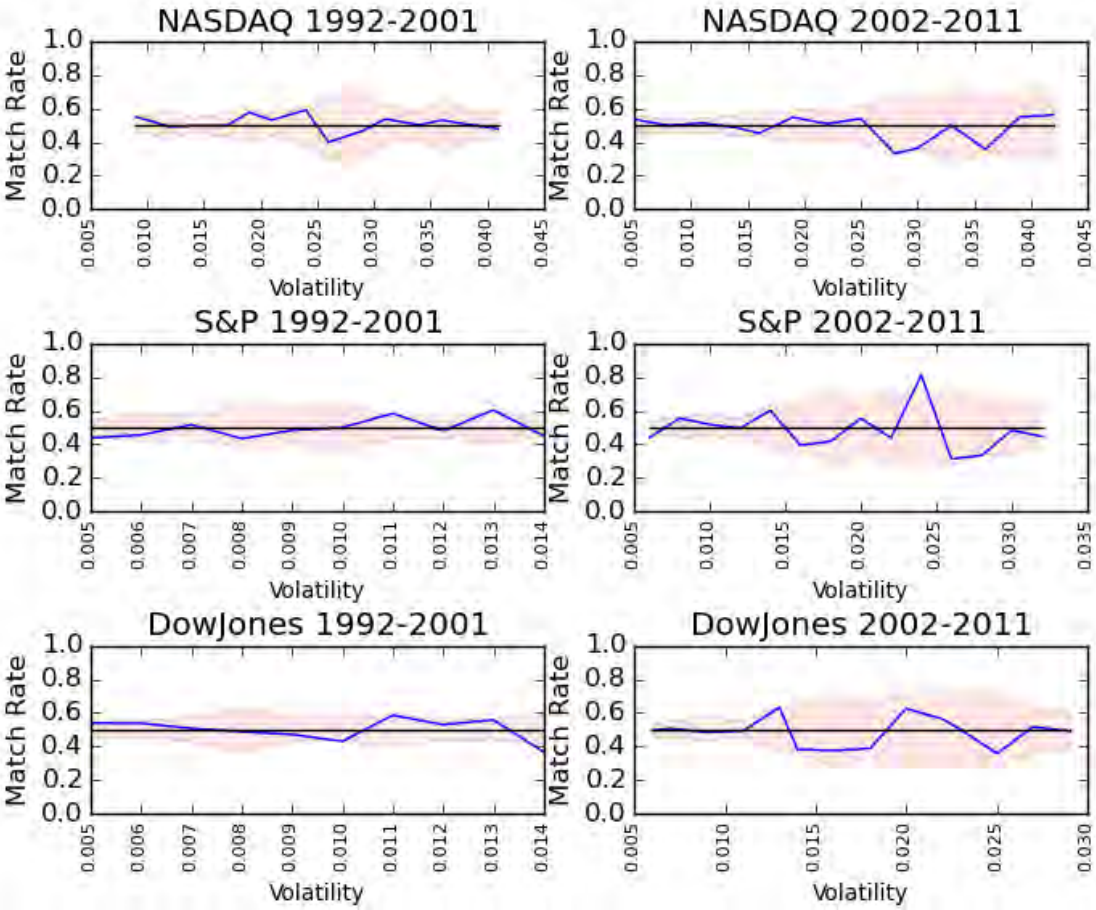

Figure 3.12. Success rates for corresponding volatility in in-sample 2 windows. Each picture analyzes a different experiment. The 6 experiments have a different combination of the subject index and time period. A 95% confidence interval is plotted on top.

Finally, we will examine the relationship between absolute real returns dur-

ing in-sample 2 periods and out-of-sample success rate of predictions. This is to determine if the index's most recent absolute return over the in-sample 2 period is a deciding factor for success rates and therefore corresponding trade decisions. Moreover, this could be interpreted as the recent market dynamics' effect on success rates. For instance, could we expect better predictions from the model when we've witnessed a sharp increase in prices during the last in-sample 2 period? Or does it work better in downswings? We've plotted the answer to these questions in figure 3.13 for all six combinations of indexes and time periods. The absolute return of each in-sample 2 period of length n is denoted

$$r_{abs} = (r_1 + 1) \cdot (r_2 + 1) \ldots \cdot (r_n + 1) - 1 \tag{3.8}$$

where r_i is the return from day i. Note that the in-sample 2 period for the NASDAQ in 2002-2011 is 20 days compared to 160 for the other five, hence the slightly shorter absolute return scale.

Essentially, there is no defining relationship between absolute returns of in-sample 2 periods and out-of-sample success rates to be found in figure 3.13. All graphs exhibit a rather flat line between these parameters, indicating that it does not matter with regard to the success rate whether prices have recently been increasing, decreasing or not changing at all.

3.2.3 Active Agents

Our next parameter of interest is the number of active agents in each trading day. Recall that each agent can choose to trade or not to trade based on the success of his most successful trading strategy. Specifically, he will choose to trade only if his most successful strategy's success rate exceeds the value of the threshold, τ. This means that the number of agents who choose not to trade varies between days. A large number of active agents would mean that the overall trading sentiment in the virtual market is positive, as agents are confident with their strategies. Conversely, a low number denotes pessimism as agents won't expect to make adequate profits and thus opt to stay out. The goal is to see if the overall number of active agents in the virtual market has any affect on success rate of predictions. Moreover, the indication of this parameter could said to be twofold. On the one hand we would be discovering if pessimism or optimism in the virtual market impacts success rates. On the other hand we would see if higher success rates are rather achieved through the collective decision of many agents or few. Both indications can said to be encoded into the active agents parameter.

For this analysis we will use the same six experiments as we did in section 3.2.2. In particular, we will find the average number of active agents in each

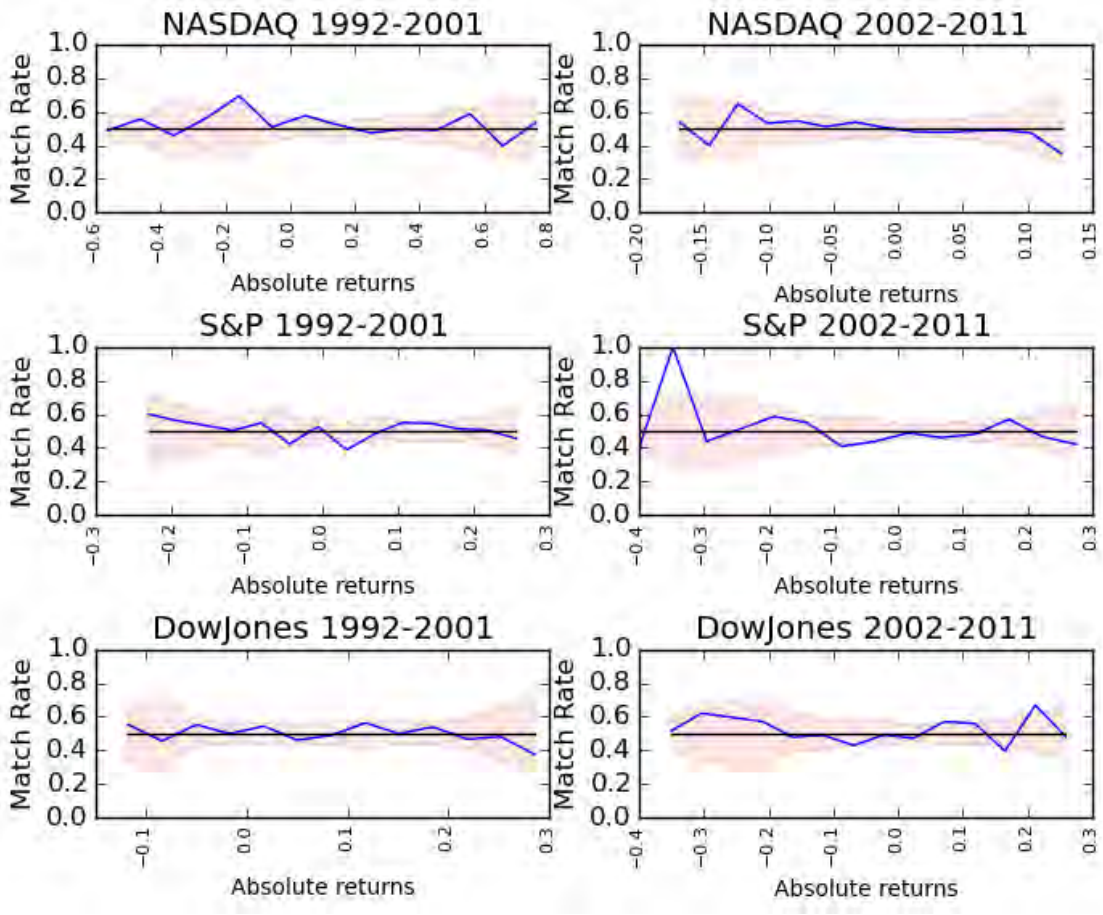

Figure 3.13. Success rates for corresponding absolute returns in in-sample 2 windows. Each picture analyzes a different experiment. The 6 experiments have a different combination of the subject index and time period. A 95% confidence interval is plotted on top.

in-sample window and plot against corresponding out-of-sample success rates in figures 3.14 and 3.15 for in-sample 1 and 2 respectively.

Results for in-sample 1 are aggregated and displayed in figure 3.14.

Upon inspection of figure 3.14, we'll find that all sub-figures except the one for NASDAQ in 2002-2011 have asymmetrically distributed window counts for number of active agents. This is reflected in the confidence intervals, who all indicate a proportionally large number of windows with an average of very few active agents. We would have expected to see the confidence intervals narrow around the half mark of active agents, indicating a normal-like shape of the distribution around the mean of half of agents. Instead,

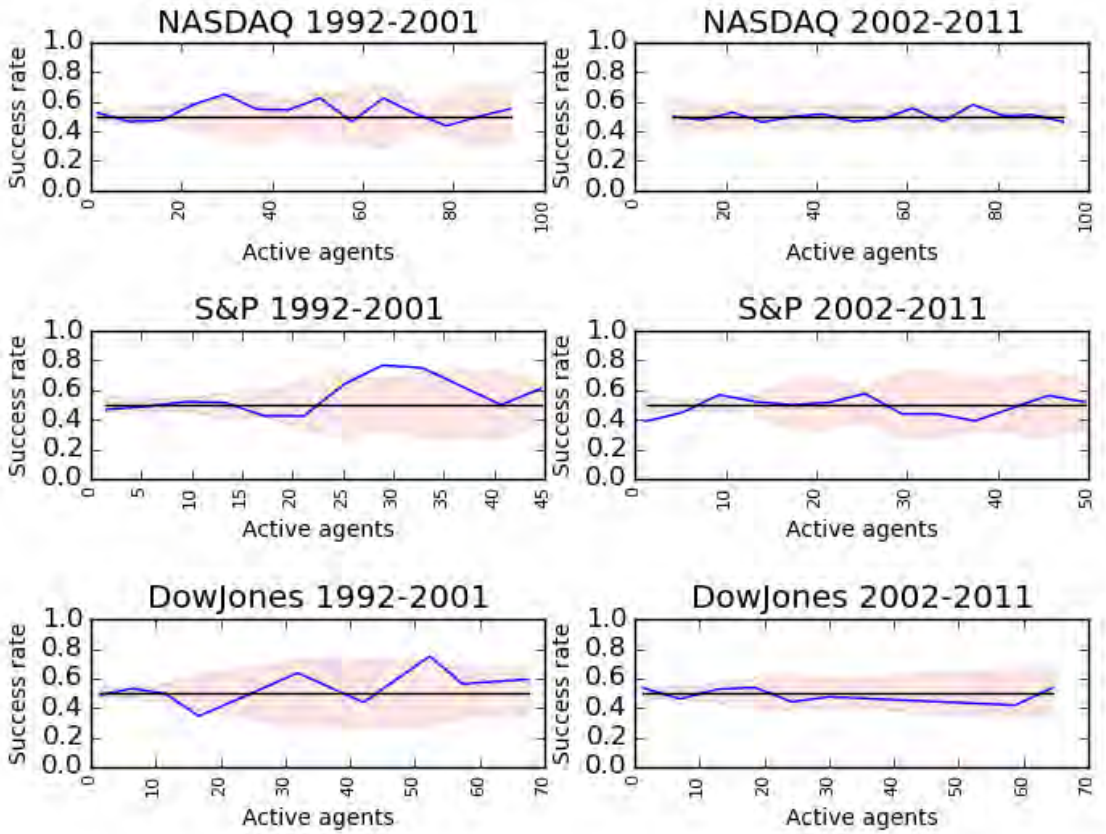

Figure 3.14. Success rates for corresponding number of average active agents in each in-sample 1 window. Each picture analyzes a different experiment. The 6 experiments have a different combination of the subject index and time period. A 95% confidence interval is plotted on top.

the overall virtual market sentiment seems to be proportionally more often in a negative state. The NASDAQ in 2002-2011 however seems to have a uniformly distributed number of windows across all values of active agents. The exact reason for this difference is unknown but could be related to difference in in-sample period lengths.

Regarding out-of-sample success rates in figure 3.14, we can't say that there is a clear trend in any of the figures. The average number of active agents in an in-sample 1 period therefore does not seem to have a deciding influence on the success rate.

When we examine figure 3.15, we'll find similar results. A proportionally

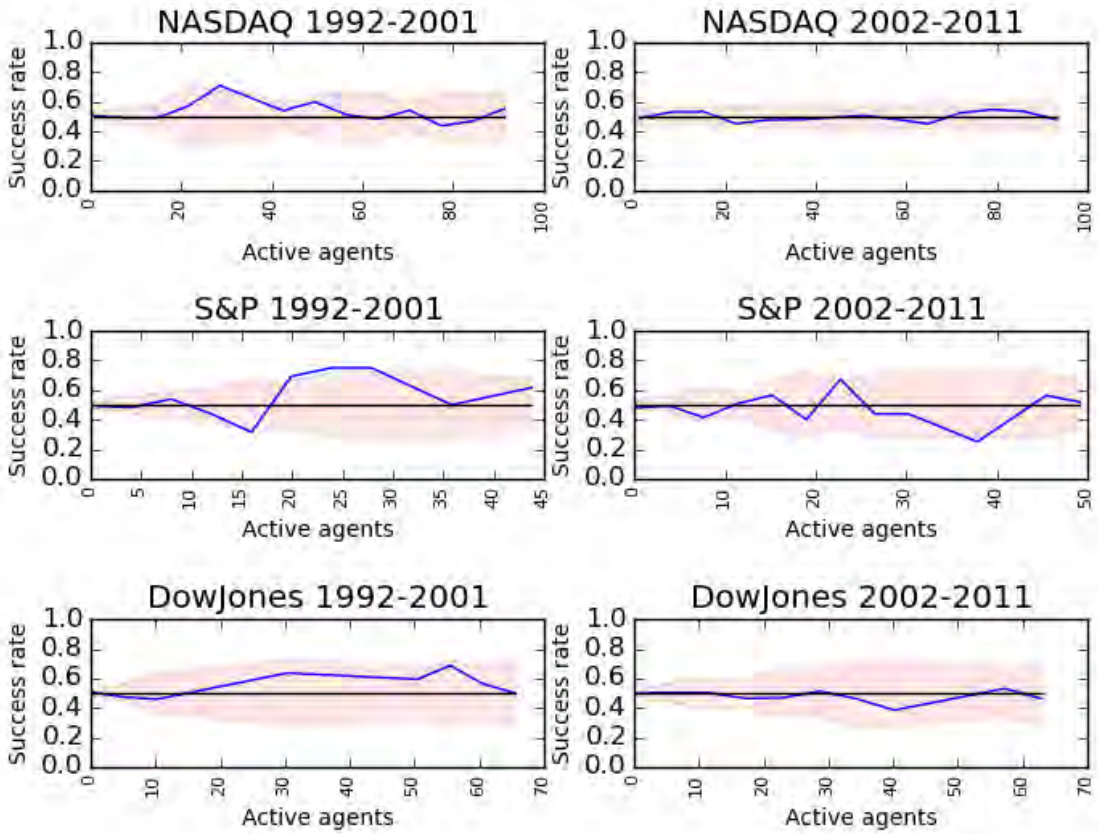

Figure 3.15. Success rates for corresponding number of average active agents in each in-sample 2 window. Each picture analyzes a different experiment. The 6 experiments have a different combination of the subject index and time period. A 95% confidence interval is plotted on top.

large number of windows does indeed seem to have a very low average of active agents, apart from the NASDAQ in 2001-2011. Each picture also exhibits no obvious trend between these parameters, thus giving us no reason to believe that number of active agents during in-sample periods is a viable prediction indicator. Curiously, however, when we compare figures 3.14 and 3.15 we'll see that the out-of-sample success rate line takes on a very similar shape in the graphs for S&P 500 in 1992-2001, with a high around 30 and 25 active agents respectively. Furthermore, this similarity between in-sample 1 and 2 figures is somewhat apparent for the NASDAQ in 1992-2001 but not for the Dow Jones or other time periods. It is hard to say whether this is a useful prediction indicator or not, as the graphs do not all characterize the

relationship in this way, nor as explicitly rising or falling. In light of this ambiguity, we'll rest our analysis of these parameters here.

3.2.4 Trends

So far we have centered our research around the model's ability to predict a correct return signal for each day. The reason for that are promising results reported by Zhang (2013) who tested this ability with a trading strategy which traded on a daily basis according to the model's prediction. In this section on trend based prediction indicators, we would like to explore the possibility that we can identify when the model is more likely to predict the correct overall move over few or several subsequent days, instead of for individual days. We think this might yield different results because the model's ability to predict a trend correctly does not necessarily have to equal its ability to predict individual days. In fact, it might even be better at predicting trends than it is at predicting individual day returns. This might be possible because a simple success rate only incorporates the return direction, not its magnitude. So in the case of an out-of-sample window with 50% of successfully predicted days, the trend prediction rate could still be 100% as the magnitude of returns could be higher in days where the prediction is correct. Which in turn means that the virtual market moves in the same direction as the real market during the trend. This of course only holds for certain ways of estimating trends in real returns and predicted returns and is consistent with our way. In detail, our way of calculating trends will be to measure the absolute return during the trend period for both predicted and real returns. We can then measure how often the predicted trend's and real trend's directions match, as well as go further and measure to what extent the predicted trend captured the real trend. We thus have 2 measures for trends. We will call the first one **trend success rate** and the second one **trend capture**, denoting the two measures of trends respectively.

Let us first examine trend success rate and trend capture in chosen experiments. Direct comparison can then be made to earlier numbers for success rates that were achieved by predicting each day. This chapter will obviously only provide an introduction into the model's potential for predicting trends. We won't recreate all previous analyses with trend success rates, but instead choose the most interesting parameter to do an initial examination on, namely, the success in in-samples.

We'll start by collecting average success rates and trend success rates in out-of-sample windows and comparing them. As before, we will pick one experiment from each of the three indexes and two time periods. This time, however, all experiments will have the same in-sample and out-of-sample lengths of 160 and 16 respectively. We won't use our usual experiment on the NASDAQ in 2001-2011 here as it has an out-of-sample length of 2. That

is simply too short for the trend lengths we want to examine. We will also report trend success rates for trends of varying length for comparison purposes and mark the most successful trend length for each experiment in green. Also worth mentioning here is that days with zero predictions are not considered and trend periods who entirely consist of zero predictions are therefore excluded. This filtering also applies to individual day success rates by excluding out-of-sample windows who only have zero predictions. The results of this analysis are displayed in table 3.1, where the OS column denotes individual day success rates and the Trend columns denote trend success rates for varying trend lengths.

Table 3.1. Comparison of individual day success rates and trend success rates in out-of-sample periods for various experiments and trend lengths. All experiments have the same in-sample and out-of-sample lengths and all numbers are percentages. The seven trend columns display trend success rates with trend periods from length 16 days to 4 days. Ticker NDX denotes the NASDAQ, GSPC denotes the S&P 500 and DJI denotes the Dow Jones.

Experiment (ticker, period)	OS	Trend 16	Trend 14	Trend 12	Trend 10	Trend 8	Trend 6	Trend 4
NDX, 1992-2001	51.7	45	43.1	50.9	46.9	51.6	49.9	49.8
NDX, 2002-2011	49.1	48.4	51.3	49.4	44.7	45.1	48.9	50.5
GSPC, 1992-2001	49.2	40	43.3	39.1	41.1	42.8	42.7	46
GSPC, 2002-2011	50.6	37.9	52.1	52.5	50.3	50	47.2	47.2
DJI, 1992-2001	51.1	47.9	49	49.7	45.5	45.1	49.6	49.6
DJI, 2002-2011	49.3	48.7	50.7	46.1	54.1	48.5	47.5	48.4

When we take a close look at table 3.1, we find that the most successful trend lengths (marked with green) only beat our traditional way of calculating success rates (OS column) in three out of six experiments. These results certainly undermine our earlier arguments that the model would potentially be better at predicting trends than it would be at predicting individual days. However, curiously enough the three experiments for which the model predicts trends better than days are all done on indexes for the period 2002-2011. This could indicate that the model is more suitable for predicting trends in this period and more suitable for predicting days in the earlier period. We can however not make that assumption at this point as statistics across numerous experiments would be needed, a potential subject for later research. Whether our results here are coincidental or represent actual difference in prediction capability in terms of time periods and method (trend or individual day), there is still some lack of consistency with regard to trend period lengths. This is apparent when we examine the position of the green cells in

table 3.1. Only two experiments report the same "winner" in trend period length, whereas otherwise no specific trend period length seems to be best for more than 1 experiment. Like before, we can probably not infer anything concrete from these results as more statistics are needed for that. We can only say that these results indicate ambiguous importance of trend lengths and possible relationship between time periods and method of calculating success rates (trend or day).

Next, let us do some initial testing for prediction indicators on the same experiments we used in table 3.1. However, since we didn't get notably better results using trends, we won't expect a big difference in the relationship between in-sample and out-of-sample here. The trend period length in subsequent analyses will be 4, due to the fact that the column for this particular trend length in table 3.1 has the highest success rate on average. Even though it only "wins" for one experiment (GSPC 1992-2001), it has similar results for them all, around and slightly below 50%. This check will be for correlation between in-sample 1, 2 and out-of-sample performance. Our measure of success this time will be trend capture, like we introduced before. In detail, trend capture reports how closely each trend period reported by the model matches the real trend period. It does so with the quotient of the absolute returns of both the model- and real trend periods. This means that if the virtual market produces absolute returns of say 0.05 during the 4 day period, and real absolute returns in the same period amount to 0.1, the trend capture will be 50%. Similarly it could have been -50% if the model produced absolute return would have been -0.05. We will further confine the trend capture parameter within the interval $[-1, 1]$. So if the absolute of the model's trend is equal or larger than the real one, the trend capture gets a value of 1 or -1 accordingly, denoting 100% or -100% trend capture respectively.

For displaying results in a descriptive manner, we'll stick to a similar format as in chapter 3.2.1. For that, average trend capture in each in-sample 1, 2 and out-of-sample window is calculated, a histogram of each in-sample's average is made, and each respective pillar's average out-of-sample trend capture is plotted on top. Results are displayed for all six aforementioned experiments at once, in two figures, for relating in-sample 1 and in-sample 2 to out-of-sample trend capture respectively. Results are displayed in figures 3.16 and 3.17

Unsurprisingly, the overall average of out-of-sample trend capture in each graph on figure 3.16 is around zero. That means that on average the model's predicted out-of-sample trend captures close to none of the real trend across values of in-sample 1 trend captures. These results are also consistent with our earlier results in table 3.1. Moreover, apparently from figure 3.16, there seems to be no meaningful increase or decrease in the relationship between

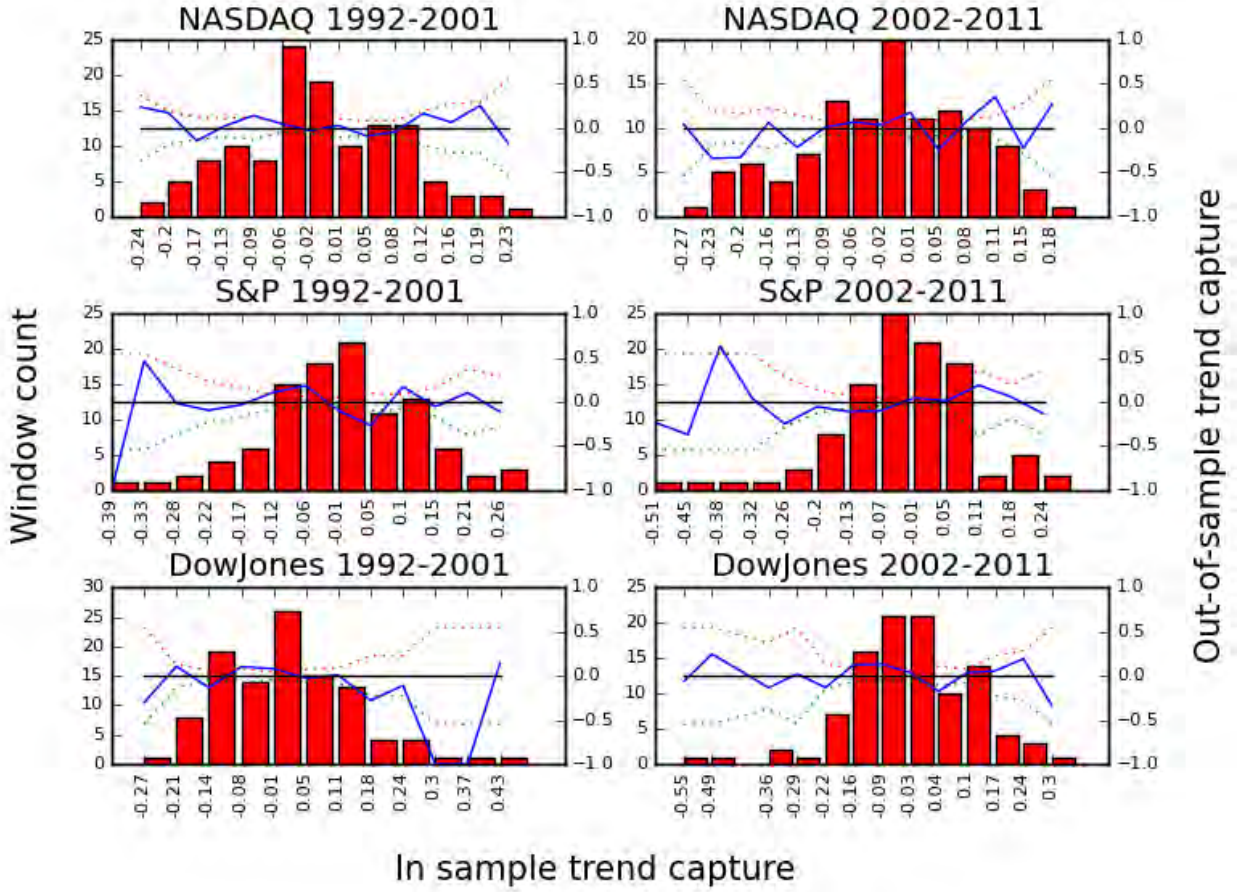

Figure 3.16. Histogram of in-sample 1 trend captures (red) with average corresponding out-of-sample trend capture on top (blue). Data is from six experiments on different indexes and time periods. A 95% Agresti-Coull confidence interval is plotted on top.

in-sample 1 and out-of-sample trend capture. In other words, the success of in-sample 1 trend capture does not seem to dictate success in out-of-sample trend capture, based on data in these experiments. We would of course need larger sample sizes from more experiments if we were to verify these results. We do however see, if we look closely, a slightly upwards sloping curve in the blue line on the graph of S&P 500 in 2002-2011 between in-sample 1 trend captures of -0.26 to 0.11. We will still leave the question of whether this slope is relevant up to later research on larger data sets.

On figure 3.17 there is a similar situation. No obvious increase or decrease seems to characterize the relationship of in-sample 2 and out-of-sample

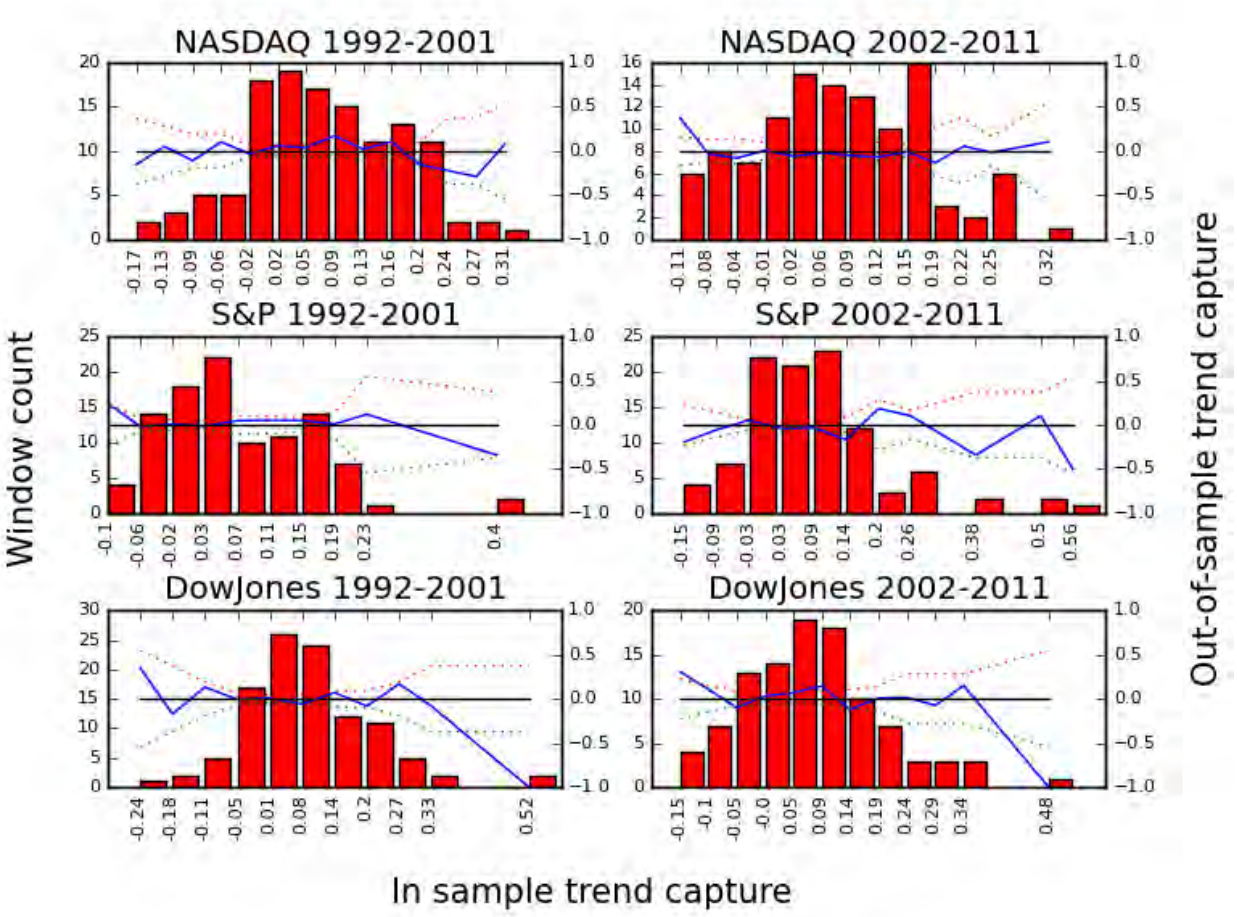

Figure 3.17. Histogram of in-sample 2 trend captures (red) with average corresponding out-of-sample trend capture on top (blue). Data is from six experiments on different indexes and time periods. A 95% Agresti-Coull confidence interval is plotted on top.

trend captures. As expected, in-sample 2 trend captures are also a bit higher than in-sample 1 on average. There is not much more we can say based on these figures, as their purpose is only to provide an initial review of trends as prediction indicators.

Chapter 4

Simple calibration

The desirable yet elusive objective of many financial market participants is to foresee the coming price movements, whether that is their sole purpose of participation or any other. The ABM we have explained in detail in earlier chapters approaches this problem from a standpoint of market inefficiency. Zhang (2013) argues that financial markets are inefficient to an extent that positive risk-adjusted abnormal gains can be made from trading with past price information only, in direct conflict with the weak form EMH. Crucial for the ABM to exploit these inefficiencies, however, is its ability to calibrate parameters such that the model's collective dynamic takes on the real market's characteristics. It does so by way of minimizing the difference between the ABM produced returns and the market's real returns. This way, the ABM determines parameters for the behavior of virtual agents on the micro level by incorporating only information from real markets on the macro level. Therefore, as Zhang (2013) calls it, the model disentangles the complex socio-economic system that is financial markets and delivers insight into the underlying mechanism at work. Such procedure of reverse engineering a real financial asset's return time series is however not as straightforward as it may seem. The problem is very computationally intensive, making it especially difficult for performing on long time series (Zhang et al., 2013). The biggest culprit is the model's large set of parameters for which its output has a highly non-linear response to (Wiesinger et al., 2012). This makes it difficult to navigate the solution space effectively in search for improvement. As a consequence, a genetic algorithm was designed for identifying a good solution, which understandably requires considerable processing power in order to explore enough parameter combinations. Nevertheless, this has been done successfully using ETH's super cluster, Brutus, yielding results we have based our analyses on in earlier chapters.

As we have seen in chapter 2, some characteristics of agents have been considerably simplified in order to make the calibration procedure feasible us-

ing the current method. This includes reducing agents' strategy sets to binary, only dependent on the direction of the market's move. Such a simplification has been common practice for researchers of agent based modeling in the context of financial markets such as Andersen and Sornette (2003); Challet and Zhang (1997); Wiesinger et al. (2012); Challet D. and Zecchina (2001); Chen et al. (2008); Marsili (2001); Jefferies et al. (2001); Johnson N. F. (2001) to name a few. Moreover, although still intensive, this way the calibration procedure is simplified enough for allowing simulations on modern CPU cores (Zhang et al., 2013). As agents' beliefs can hardly be simplified more without potentially losing some of the properties that make up the dynamics of the virtual market, we must seek ways to make the calibration sequence more practical in implementation. Calibrating the model partly revolves around finding each agent's strategy set, F_i, such that the market produces the desired output. For one particular set of strategies, the agent then uses the in-sample 1 periods for identifying the best performing strategies to be used. The one strategy he chooses to follow at each time thus defines that particular agent's part in the collective decision. Our impression is that the collective decision could possibly be molded in a more deterministic way. Furthermore, to that end, we are keen on testing alternate, simplified ways of optimizing agents strategies, especially with a reduced population. For that purpose, we will take a large step back and examine in detail how an agent makes a decision in each time step. The next few sub-chapters will focus on this subject. Specifically, in chapter 4.1 we will investigate a reverse engineering of a real return time series with one agent only. Similarly, in later chapters, we will attempt the same using three agents who interact in a virtual market place.

4.1 One agent model

We recall from section 2 that agents have an exogenously determined memory which represents their intellectual limit. They are thus not capable of incorporating the entire information history into their trade decision and thus do not necessarily make the optimal decisions at each time point. The memory limit impacts agents' behavior through their sets of strategies. The strategies an agent has to choose from only incorporate information from m time steps back. When an agent chooses one specific strategy to follow, he is defining his part in the aggregate decision and thus the virtual market's return. Since our problem is to make the virtual market's return resemble the real return, let us consider a simplified setting where the aggregate decision consists of that of only one agent. Moreover, let our one-agent virtual market be different from our previously defined market such that instead of the initial random set of strategies to choose from, $F_i := \{f_i^1, \ldots, f_i^s\}$, our agent would be endowed with only one strategy. Considering such vastly

simplified settings is obviously not for the purpose of capturing complex dynamics, but rather to demonstrate the power of one agent, or say a group of agents where the aggregate decision is dictated by a like-minded majority, to unravel the complexity embedded in real return time series. The extent to which this is possible might in turn indicate how calibrating a crowded virtual market place could be made more efficiently.

As our optimization problem for calibrating a one agent model to a piece of the real return time series is obviously much simpler than before, we do not require a genetic algorithm. Instead of looping through every possible version of the strategy in search for the optimal one, specifically 2^{2^m} possibilities, we would simply examine the in-sample period's real returns and assign the optimal strategy to the agent. For example, let us check how far an agent is able to predict correctly if we tailor his trading strategy to the real return time series for each day. We find that when we reduce the NASDAQ (2002-2011) time series down to just binary indicators for up and down moves, and strategically assign a correct decision to the agent for each rolling window of past moves of length m, the agent's average number of correct prediction days exceeds his memory length, m. For detailing this procedure, let us borrow from Satinover and Sornette (2012), who researched cycles and determinism in binary time series. If our agent has memory length $m = 3$, the binary form of the NASDAQ is broken down into a series of "events" of length 3, each to which our agent has one specific action according to his strategy; buy or sell. For example, the rolling windows of length 3, or "events", for the first seven days in the NASDAQ would be:

$$\{1,1,0,1,0,1,0\} \rightarrow \{(1,1,0),(1,0,1),(0,1,0),(1,0,1),(0,1,0)\} \tag{4.1}$$

Each event's possible transition into other events can then be illustrated in a binary De Bruijn graph of the 3rd order, displayed on the left side of figure 4.1. The right side of figure 4.1 displays the three distinct events that happen in the first seven days of the NASDAQ.

Continuing our example with the first seven days of the NASDAQ, it is easy to see what our agent's optimal strategy would partly look like. If he is to predict correctly for each of the four days he participates (he starts participating after the first event), his strategy must have the actions that correspond to the three events as shown on the right side of figure 4.1. Specifically, three out of eight events in his strategy must therefore trigger the following actions: $\{1,1,0\} \rightarrow 1, \{1,0,1\} \rightarrow 0$ and $\{0,1,0\} \rightarrow 1$. By determining the agent's strategy in this way for each day, he has already predicted correctly four days in a row. It therefore becomes clear that as days pass he will add to his strategy every new event that happens and the corresponding correct prediction. This might even work for an extended period of time, as long

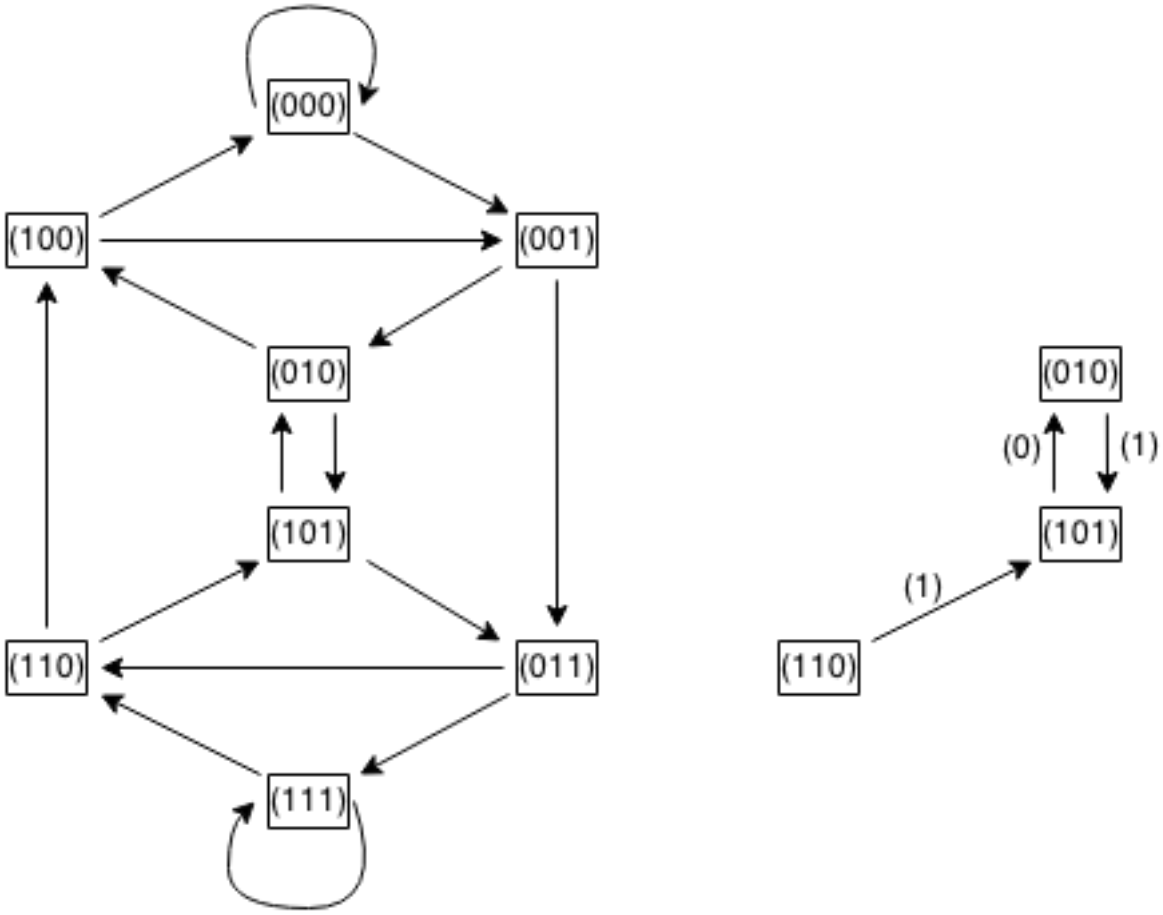

Figure 4.1. Binary De Bruijn graph of the 3rd order. The left part displays all possible state transitions for an agent with memory of length 3. The right part displays the state transitions that happen in the first seven days of the NASDAQ, as well as the agent's correct predictions.

as the agent encounters new events he can add to his strategy, or old events that happen again have the same next day move as the agent's strategy dictates. The agent's winning streak however stops as soon as he re-encounters an event whose next day move is different from what his strategy predicts. He is therefore ill-equipped to handle events that happen again and again, as they might not have the same next day move as his strategy foretells. Let us now test the one agent model by checking how far on average he is able to predict correctly. We'll detail this test more accurately in definition 4.1.

Definition 4.1 *An agent has memory m and an empty strategy set. The agent starts at t_0 in a time series of returns. The agent adds the first event to his only*

strategy along with the corresponding correct trade decision. He then moves to t_1 and does the same. He continues until he re-encounters an event that now has a different trade decision than his strategy previously registered. The agent stores how many days he was able to predict correctly and starts this sequence again, now from t_1 with an empty strategy set. The agent does this until he reaches the end of the time series, at which point results for average length of runs are returned. The approximate algorithm for this sequence is as follows:

Algorithm 2: Test One-Agent Model

```
m ← memory ;                                  /* Determine memory size */
TS ← NASDAQ ;                          /* Assign NASDAQ as time series */
for t = 1:end-m do
    initialize strategy set;
    for i = t:end-m do
        event ← TS_{i:i+m} ;                              /* Current event */
        if event not in strategy then
            Add event to strategy;
            Assign correct action ;   /* Correct action is TS_{i+m+1} */
        else
            if Agent's action is correct then
                Continue;
            else
                results ← Count and store how far agent got;
                Break inner for-loop;
            end
        end
    end
end
Return average of results;
```

Let us now run this test multiple times, using a different memory length each time. The results from this are displayed in table 4.1.

Table 4.1. Average number of correct prediction days in a row. The time series is NASDAQ in 2002-2011. Results are for various memory lengths. The number of possible different events is shown for comparison.

Memory Length	Number of possible events	Average number of correctly predicted days in a row
3	8	5
6	64	13.7
8	256	28.5
10	1024	54.9
12	4096	101.3

The results in table 4.1 show that if our agent for instance has a memory length of 8, and our in-sample period is of length 20, over half of in-sample periods could have a 100% success rate of predictions. We can further see, as we mentioned above, that all averages for the number of correctly predicted days in a row exceed the memory length, m.

Let us also test the model in a slightly different way. Specifically, we'll calibrate the one agent model to the NASDAQ on in-sample periods, and report results for average in-sample success rates for various in-sample lengths. A more accurate definition of this test is in definition 4.2.

Definition 4.2 *An agent has memory m. His strategy is created from scratch in each in-sample window, such that he'll trade as often as possible like the real returns. In-sample windows are then shifted with out-of-sample window length. In each in-sample window, the agent's strategy is the one that minimizes the sum of squared differences between the agent's binary decision, r^{agent}, and the real market's binary move, μ:*

$$Minimize: \sum_{i}^{i+W_{is}} (\mu_i - r_i^{agent})^2 \tag{4.2}$$

where W_{is} denotes the in-sample length and i is incremented on out-of-sample window length. Immediately, following each in-sample window, comes an out-of-sample window. Since the agent's strategy is created only for one specific in-sample period of limited length at a time, the strategy might not include all 2^m events. Consequently, he will only trade in the out-of-sample window if the event there has been included in his trading strategy. If we denote one possible event in the time series to be $e : \{0,1\}^m$, we can write all events in in-sample window i as his strategy space for that window:

$$f_i := \{e_1, \ldots, e_n\}, \tag{4.3}$$

where $n \leq 2^m$. Consequently, the agent will only trade in out-of-sample window i when

$$e_i^{out} \subset \{e_1, \ldots, e_n\}, \tag{4.4}$$

where e_i^{out} is the set of events in the i-th out-of-sample window.

The strategy optimization for each in-sample window will undoubtedly yield higher in-sample success rates than using our earlier approach of creating the strategy day for day. We'll also report the average out-of-sample success rate from all windows where trades were made. Moreover, we will compare the agent's out-of-sample prediction success rate to that of 10.000 random strategies who predict at the same days as our agent, with probabilities of buy or sell decisions of 50%. Note that agent's memory length, m, is fixed at 3 for all in-sample period lengths and out-of-sample period length is fixed as one[1]. Results from this analysis are displayed in table 4.2

Table 4.2. One agent model's in-sample and out-of-sample success rates for different in-sample lengths. The time series being reverse engineered is NASDAQ in 2002-2011. Agent memory length is fixed at 3 and out-of-sample length is 1 day. Last column shows the proportion of the 10.000 random strategies that perform worse than the one-agent model.

In-sample Length	Avg. in-sample success rate	Avg. out-of-sample success rate	Random strategies worse than the model
20	74%	52%	97.3%
60	64.8%	51.6%	94%
100	62%	52.6%	99.4%
150	60.2%	51.5%	92.4%
200	59%	50.9%	79%

The results clearly indicate that such a deterministic calibration to the time series can yield high in-sample success rates of predictions. Further, we can see that the one agent model is able to predict the return signal in the one day following each in-sample period better than the majority of random strategies can.

Some additional statistics for this test is displayed in table 4.3. Note that the proportions are only calculated from the out-of-sample windows where the agent makes a trade, and therefore values for the proportion of up-moves in the real time series are not the same for all in-sample length values. The

[1]We chose to have a short out-of-sample length since such a simplistic approach will certainly not produce any descriptive complex dynamics, and is therefore only expected to have some predictive capabilities in the immediate time steps, if any.

proportion of up-moves in the entire time series is 53.7%. In general, up-moves are more frequent than down-moves for all in-sample lengths, both for the real time series as well as the predicted values. Also, the model is clearly better at predicting up-moves than down-moves, as reflected in the proportion of caught up-moves versus down-moves.

Table 4.3. One agent model out-of-sample statistics. The time series being reverse engineered is NASDAQ in 2002-2011. Agent memory length is fixed at 3 and out-of-sample length is 1 day.

In-sample length	Up-moves: real time series	Up-moves: predicted time series	Up-moves caugh	Down-moves caught
20	53.6%	65.8%	66.6%	35.2%
60	54%	64.3%	64.8%	36.1%
100	54.2%	64.8%	66%	36.7%
150	54.4%	65.4%	65.5%	32%
200	54.5%	67.3%	66.7%	32%

It is obvious from the numbers in the second column of table 4.3, that they are all larger than the average out-of-sample success rates in table 4.2. That essentially means that we would get a higher out-of-sample success rate by simply predicting up in each out-of-sample window, thus also catching 100% of up-moves and 0% of down moves. Furthermore, a buy-and-hold strategy for the entire index, would yield a success rate of 53.7%, trumping most out-of-sample success rates in table 4.2. This is however not the only way of measuring the prediction power of our model. The above reported success rates infer little about absolute and risk-adjusted returns from a trading strategy using the model. Direct testing of a one agent model trading strategy would reveal more of the model's potential for practical application. Such a test would furthermore need to be properly bench-marked, in order to identify true skill from luck. Therefore, this inspires a comparison in the spirit of Daniel et al. (2008), using random trading strategies with the same number of trades, duration in the market and randomized entry points. The following four tests of strategies will be made:

1. A strategy employing the one agent model
2. Averages from 1000 random strategies with the same number of trades, duration in the market and randomized entry points.
3. A strategy employing the one agent model - with trade percentage fee of 0.5 basis points per trade.
4. Buy-and-hold strategy for the NASDAQ. Stock is bought at the beginning and kept until the end.

We'll report both absolute returns and Sharpe ratios for all strategies for various in-sample lengths. Memory of the agent will be fixed at 3 like before. The strategies will be tested with back-testing on an ETF[2] tracking the NASDAQ between 2002-2011. We'll use the algorithmic trading module PyAlgoTrade in Python for achieving this goal. Results from this are displayed in table 4.4.

Table 4.4. Comparison between the trading strategies. Strategies are back-tested on the NASDAQ in 2002-2011. Agent's memory is 3 and in-sample lengths vary. The values for random strategies are averages and the values in brackets are standard deviations.

In-sample length		1 agent	Random strat.	1 agent w. fees	Buy and hold NDX
20	Sharpe	-0.217	0.025 (0.265)	-0.559	0.203
	Abs returns	-30.4%	20.2% (67.6%)	-62.8%	49%
60	Sharpe	0.267	0.044 (0.259)	-0.016	0.222
	Abs returns	70.1%	24.9% (75.7%)	-2.3%	56.2%
100	Sharpe	0.286	0.048 (0.257)	0.008	0.283
	Abs returns	75.6%	25.3% (69.4%)	3.6%	79.8%
150	Sharpe	-0.035	0.096 (0.262)	-0.324	0.391
	Abs returns	-1.8%	38.5% (75%)	-41.1%	128.8%
200	Sharpe	-0.125	0.091 (0.257)	-0.417	0.385
	Abs returns	-14.9%	36.3% (70.4%)	-48.4%	123%

Table 4.4 shows that the one-agent model has better Sharpe ratios and absolute returns than the averages of random strategies for in-sample lengths of 100 and 60. This advantage is however eliminated when trading fees are included.

In figure 4.2, the return history of the most successful one agent model trading strategy without fees (in-sample length 100) is presented visually. The figure is in three parts, displaying the NASDAQ itself, returns from each trade and accumulated returns from the strategy (from top to bottom respectively).

[2]PowerShares QQQ Trust, Series 1. (ETF: exchange traded fund)

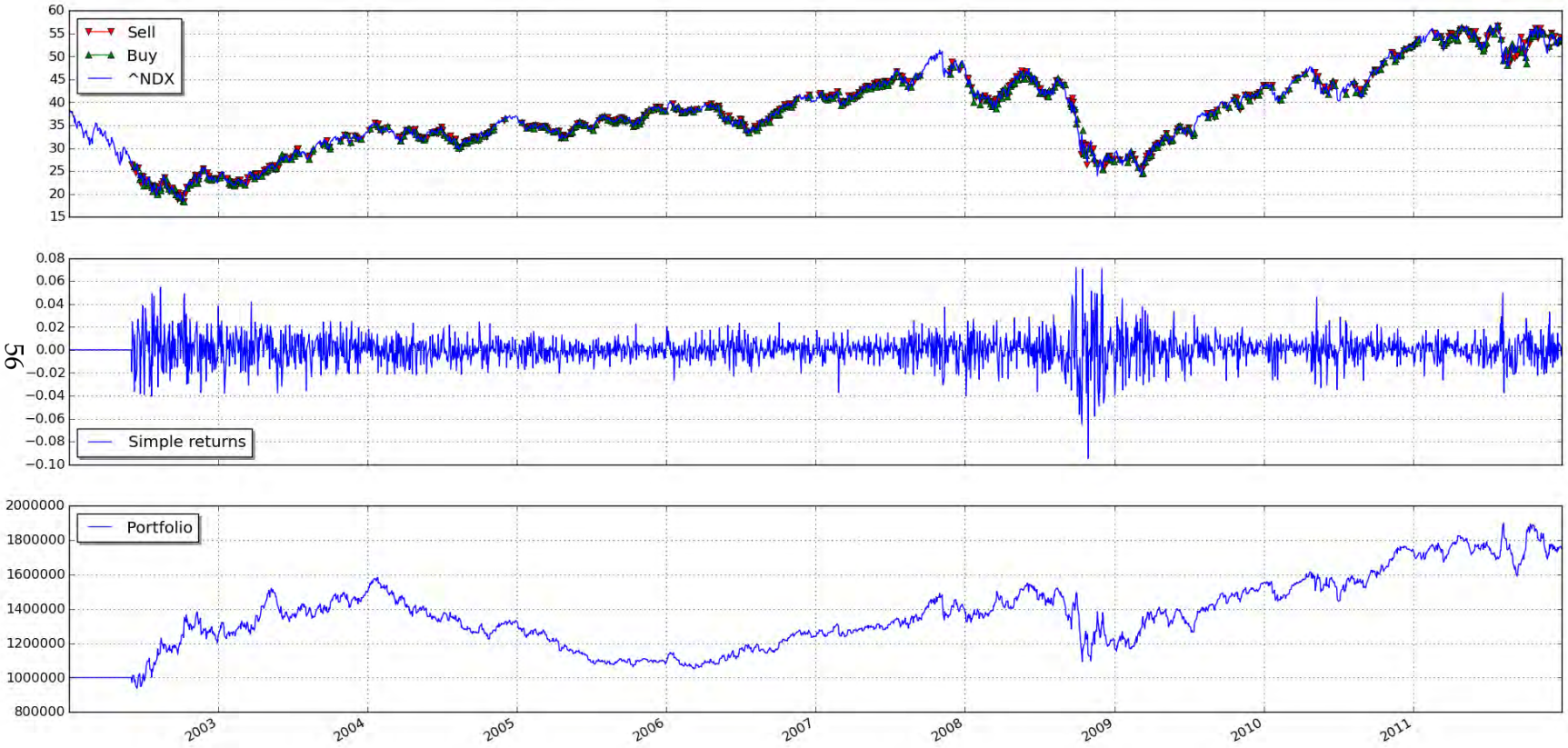

Figure 4.2. Performance of the one agent model trading strategy on the NASDAQ in 2002-2011. The top figure shows the the index and when buy/sell orders where made. The middle figure shows returns for each buy/sell order. The bottom figure shows returns from the trading strategy. Memory and in-sample lengths are 3 and 100 respectively.

4.2 Three agent model - Majority Game

Imagine a virtual marketplace populated by 3 agents only. Again here, this model is not created for the purpose of constructing the complex dynamics and stylized facts of stock markets through a bottom up approach as in chapter 2. Our interest in the three agent virtual market is that of accurately simulating real market returns. We are convinced that by allowing three agents to interact, and by accurately defining their behavior, they will have a great potential for reverse engineering a long piece of real financial time series. As before, we will explain and demonstrate this capability with examples done on the NASDAQ in 2002-2011.

The three agent's advantage in emulating real returns is related to the additional complexity that their interaction produces. Moreover, to add onto that complexity, one of the three agents will be endowed with 2 strategies in stead of one. All strategies are of course carefully constructed such that the virtual market produces the correct output. The reason for one agent having two strategies is for the virtual market to overcome recurring events in the time series who have different subsequent move signal. Let us again use the beginning of the NASDAQ (2002-2011) for explaining this occurrence. If we now consider the first eight days of the NASDAQ, $\{1,1,0,1,0,1,0,0\}$, and examine them using the relevant part of a binary De Bruijn graph as before, we'll see that this is exactly what happens for the event $\{0,1,0\}$.

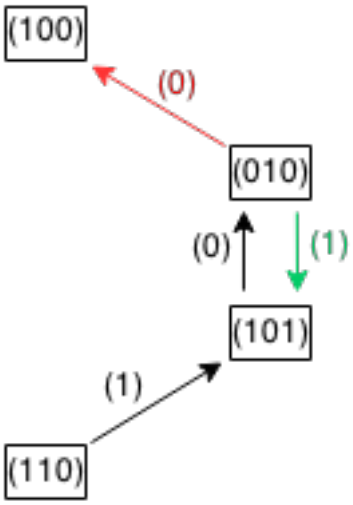

Figure 4.3. Four states of a binary De Bruijn graph. The figure displays the two possibilities following the event $\{0,1,0\}$, both of which occur in the first eight days of the NASDAQ. The first state transition from the recurring event is denoted in green and the second one in red.

The event $\{0,1,0\}$ happens twice in the first eight days of the NASDAQ. The first occurrence is followed by an up-move in the market, denoted 1. The next occurrence, however, is followed by a down-move, or 0. If we were using a one-agent model and creating the agent's strategy day after day, the strategy would already have the action $\{0,1,0\} \rightarrow 1$ when it came to the second occurrence of $\{0,1,0\}$. As figure 4.3 shows, the two states $\{1,0,1\}$ and $\{1,0,0\}$ can both follow $\{0,1,0\}$, but our agent's strategy only predicts $\{1,0,1\}$. He would therefore predict falsely on the fifth day. With the three agent model, we are able to overcome one recurring event, with different next day move, per run. As a consequence, this allows the model to reverse engineer a time series 100% correctly for a longer time than the one agent model could. We achieve this by making the agent with two strategies switch between them before the second occurrence of the recurring event. He thus alters his trade decision from the false prediction to the correct one while still having made the correct prediction at the first occurrence.

This virtual market model is of course extremely deterministic as all strategies for agents are specifically tailored for the time-series in question. This time, we do however have some form of agent decision making, as the agent with two strategies will evaluate both strategies' performance and choose which one to use for each day. We only endow him with the correct strategies (created day for day), which he will then choose to employ based on how they have payed off in the past. In chapter 2, we detailed four different ways to evaluate payoffs from trading strategies. Here, our switching agent will be playing the Majority Game, which means that his derived payoff for strategy i is as follows

$$\pi^{MajG}(f_i(\mu_t), \text{sign}(r_t)) = f_i(\mu_t)\,\text{sign}(A_t). \tag{4.5}$$

The other two agents, who have only one strategy each, have no choice and thus do not need to track their strategy's past payoff. We can therefore say that they aren't playing any game at all.

The specifics of each agent's behavior is as follows. One agent will have a similar role as the agent in the one-agent model we detailed before. Specifically, events will be added to his strategy each day and he will trade in exact accordance to the real return time series. This goes on throughout the second occurrence of a recurring event, at which point he will obviously make a wrong prediction. We will call this agent agent one. Another agent, endowed with two strategies, will also trade in accordance with the real returns. His first strategy will in fact be exactly the same as agent one's, while his second strategy will differ only to the first on the recurring events. Specifically, the first strategy will predict correctly at the first occurrence of the recurring event, while the second strategy will predict correctly at the

second occurrence. We'll call this agent agent two. The third agent will trade somewhat exactly opposite to the real returns. This means that he will be wrong at the first occurrence of the recurring event, and correct at the second occurrence. We'll call this agent agent three.

Although this procedure seems very straightforward, there are some adjustments that need to be made for the virtual market to predict correctly through a recurring event with a different next day move. Although we have explained how it is possible, we haven't given agent two any reason to switch strategies at the correct point in time, which is essential for this to work. For agent two to switch from the first strategy to the second at the time of the second occurrence, we modify strategy one's past decisions such that it predicted wrongly in the most recent two time steps. If we denote the cumulative payoff from an accurate strategy to be p, strategy one's payoff at that time would be $p - 2$. Agent two's strategy two, however, is not modified. At the time point before the recurring event, it had accumulated payoff of $p - 1$, where one is withdrawn because of the wrong prediction at the first occurrence of the recurring event. Thus, at the time step before the second occurrence of the recurring event, agent two will find that strategy two has performed better overall and therefore switches just in time to make a correct decision at the next time step. If P_1 and P_2 are the respective strategies' payoffs, $P_1 < P_2$ must be satisfied before the switching day.

Since there are three agents, it is the majority decision that decides the virtual market's return signal. At each time step, we therefore require two out of three agents to predict exactly like the real returns. Moreover, our method for making agent two switch strategies is backward looking. That is, everything goes on unchanged until an event that has already happened happens again, with a different next day return signal. At that point we will adjust agent two's decisions in the past, so his strategies predict like we define above. Since we are changing decisions in the past, the number of agents predicting like the real returns also changes. Therefore, to make sure that two out of three agents always predict correctly, we must use agent three. At the time step before the recurring event, agent three had always predicted opposite of the real returns. This works well until that time point, as agent three's prediction is always in the minority and thus ignored. However, when the second occurrence of a recurring event happens, we must use agent three in the two most recent time steps to fix the majority decision. Since we suddenly changed agent two's predictions into being wrong, agent three must predict correctly at those time points in order for the market to predict correctly as a whole. We therefore also use a backward looking approach for agent three, and reverse his decisions in the two most recent time steps.

We have now explained all three agents' strategies, decision making and

how we use a backward looking approach to adjust previous decisions. Let us now test the model with similar methods we employed with the one agent model. We'll define the first test in definition 4.3. Note however that even though our test of the three agent majority game model here will be on large pieces of time series, which includes both up-trends, down-trends and regime changes, the majority game is still only intended to capture market dynamics in up- and down trending markets. A model with the majority game payoff in times of regime shifts is therefore unrealistic, which should be kept in mind when reviewing results in this chapter. These and other possible shortcomings are further discussed in closing remarks.

Definition 4.3 *Three agents have memory m and empty strategy sets. The agents start at t_0 in a time series of returns. The agents add the first event to their strategy along with the corresponding trade decision. They then move to t_1 and do the same. They continue past one recurring event with different next day return. When that happens again, they will predict incorrectly, store how many days they were able to predict correctly and start this sequence again, now from t_1 with empty strategy sets. The agents do this until they reach the end of the time series, at which point results for average length of runs are returned. The approximate algorithm for this sequence is as follows:*

Algorithm 3: Test Three-Agent Model

```
m ← memory ;                                  /* Determine memory size */
TS ← NASDAQ ;                          /* Assign NASDAQ as time series */
for t = 1:end-m do
    for i = t:end-m do
        event ← TS_{i:i+m} ;                              /* Current event */
        A1 ← Fetch Agent one's decision ;
        A2 ← Fetch Agent two's decision ;
        A3 ← Fetch Agent three's decision ;
        if Majority's action is correct then
            Continue;
        else
            results ← Count and store how far agent got;
            Break inner for-loop;
        end
    end
end
Return average of results;
```

Our results for this test are displayed in table 4.5. We can immediately see

that these numbers for average correctly predicted days in a row are considerably higher than for the one agent model. For instance, with memory length eight and in-sample period of 40, we could potentially be able to predict over half of windows 100% correctly.

Table 4.5. Average number of correct prediction days in a row - 3 agent model Majority Game. The time series is NASDAQ in 2002-2011. Results are for various memory lengths. The number of possible different events is shown for comparison.

Memory Length	Number of possible events	Average number of correctly predicted days in a row
3	8	7.3
6	64	21
8	256	44.2
10	1024	80.5
12	4096	154

For completeness, let us also check how the three agent model fares when we transform the NASDAQ in 2002-2011 into trends instead of individual day return signals and employ the test in definition 4.3. The definition of a trend will be similar to the one we used before and a trend time series is defined as follows.

Definition 4.4 *Denote trend period length as l and asset prices in a time series at time t as p_t. Returns in trend period i of length l can then be written:*

$$r_i^{trend} = \frac{p_{t+l}}{p_t} - 1 \tag{4.6}$$

We can now transform an entire time series of prices into its corresponding trend return time series, by calculating r^{trend} for all t in the sequence $\{xl\}_{x=0}^{x=\frac{n}{l}-1}$, also denoted as:

$$t_x = xl \quad \forall x \in \{0, \ldots, \frac{n}{l} - 1\}, \tag{4.7}$$

where n is the length of the original time series. We then also transform the returns in the trend time series into binary form; 0 for negative and 1 for positive.

For this test let us also report results for various trend and memory lengths. Results for average number of correct prediction days for varying trend and memory lengths are displayed in table 4.6.

Table 4.6. Average number of correct prediction days in a row - 3 agent model majority game with trends. The time series is NASDAQ in 2002-2011. Results are for various trend and memory lengths.

	Avg. number of correctly predicted days in a row		
	Trend length		
Memory length	3	6	10
3	7.3	6.91	6.82
6	21.9	21	22.7
8	41.8	40	44
10	77.14	74.46	69.14
12	118.14	175	239

When tables 4.5 and 4.6 are compared, we can see that no trend length yields consistently better results than before. The averages in table 4.6 who are different from their corresponding values in table 4.5, statistically significant to the 5% level, are marked in green.

Similarly to the chapter on the one agent model, here we will also see how the three agent model manages to reverse engineer the NASDAQ on in-sample windows of various lengths. We'll report these results with in-sample and out-of-sample success rates. The test is defined as follows.

Definition 4.5 *Three agents have memory m and empty strategy sets. Their strategies are created anew in each in-sample window by including events day after day (strategies are therefore not optimized for each in-sample window). They then trade in in-sample windows, who shift with out-of-sample window length. If we use our earlier notation of events, $e : \{0,1\}^m$, that happen in in-sample window i, we can write an agent's strategy space for that window as before:*

$$f_i := \{e_i, \ldots, e_n\}. \tag{4.8}$$

Again here, since $n \leq 2^m$, not all events always occur in each in-sample window. And so an agent can only trade in the out-of-sample window if it has occurred, or:

$$e_i^{out} \subset \{e_1, \ldots, e_n\}, \tag{4.9}$$

where e_i^{out} is the set of events in the i-th out-of-sample window.

For making this experiment comparable to the one agent model, let us also report results for a fixed memory length of 3, as well as for the same in-

sample period lengths. Additionally, we will compare the three agents' out-of-sample prediction success to the one from 10.000 random strategies by reporting the proportion of random strategies our three agents are able to beat. Our results for this test are displayed in table 4.7.

Table 4.7. Three agent model majority game in-sample and out-of-sample success rates for different in-sample lengths. The time series being reverse engineered is NASDAQ in 2002-2011. Agent memory length is fixed at 3 and out-of-sample length is 1 day. Last column shows the proportion of the 10.000 random strategies that perform worse than the three agent model.

In-sample length	Avg. in-sample success rate	Avg. out-of-sample success rate	Random strategies worse than the model
20	72.9%	53.1%	99.8%
60	59%	50.7%	75.3%
100	56.1.2%	51.7%	95%
150	54.3%	50.3%	62%
200	53.5%	50%	47.8%

Careful examination of table 4.7 reveals similar results as for the one agent model. Average in-sample success rate is of course directly dependent on the period length, and is slightly lower than for the one agent model. This is understandable as the one agent model is optimized on each in-sample, and that becomes more important as in-sample lengths increase for such a small memory length. Additionally, average out-of-sample success rates are consistently above 50%, but slightly worse than for the one agent model. Same applies to the proportions of random strategies underperforming the three agent model.

Further statistics for this experiment is displayed in table 4.8. Note again that these proportions are only for the out-of-sample days where our agents made a trade, hence various values in the second column. Apparently, both the real- and model predicted moves are more often positive. The proportion of real up-moves that are correctly predicted is furthermore always between 54-56% while proportion of down moves correctly predicted is always lower than 51%.

A similar test of trading strategies as we did for the one agent model is also applied here for the same purpose. As before, the first trading strategy will be the three agent majority game model itself. The second one will be averages from 1000 random strategies with same amounts of trades, duration in the market and random entry points. The third is the three agent majority game model with trade fees of 0.5 basis points per trade. A comparison

Table 4.8. Three agent model majority game out-of-sample statistics. The time series being reverse engineered is NASDAQ in 2002-2011. Agent memory length is fixed at 3 and out-of-sample length is 1 day.

In-sample length	Up-moves: real time series	Up-moves: predicted time series	Up-moves caught	Down-moves caught
20	53.6%	52.7%	55.5%	50.4%
60	54%	53.7%	54.1%	46.7%
100	54.2%	54.7%	55.8%	46.7%
150	54.4%	54.7%	54.6%	45.3%
200	54.5%	55.3%	54.8%	44.1%

to the buy-and-hold strategy for the NASDAQ will also be provided. The strategies are back-tested on an the same ETF tracking the NASDAQ from 2002-2011. Results for absolute returns and Sharpe ratios are displayed in table 4.9.

Table 4.9. Comparison between the trading strategies. Strategies are back-tested on the NASDAQ in 2002-2011. Memory length is fixed at 3 and in-sample lengths vary. Values for random trading strategies are averages and values in brackets are standard deviations.

In-sample length		3 agent majg	Random strat.	3 agent majg w. fees	Buy and hold NDX
20	Sharpe	0.008	-0.053 (0.299)	-0.357	0.203
	Abs returns	4.4%	7% (72.7%)	-46.9%	49%
60	Sharpe	0.489	-0.052 (0.284)	0.131	0.222
	Abs returns	155.6%	4.7% (63.1%)	31.2%	56.2%
100	Sharpe	0.231	-0.046 (0.28)	-0.117	0.283
	Abs returns	57.9%	6.4% (64.5%)	-17.1%	79.8%
150	Sharpe	-0.059	-0.048 (0.283)	-0.4	0.391
	Abs returns	-5.7%	6.9% (60.9%)	-48.3%	128.8%
200	Sharpe	-0.385	-0.043 (0.28)	-0.752	0.385
	Abs returns	-44.4%	8.5% (59%)	-69.9%	123%

Return development for the best performing three agent majority game model trading strategy (in-sample length of 60, without fees) is displayed in figure 4.4.

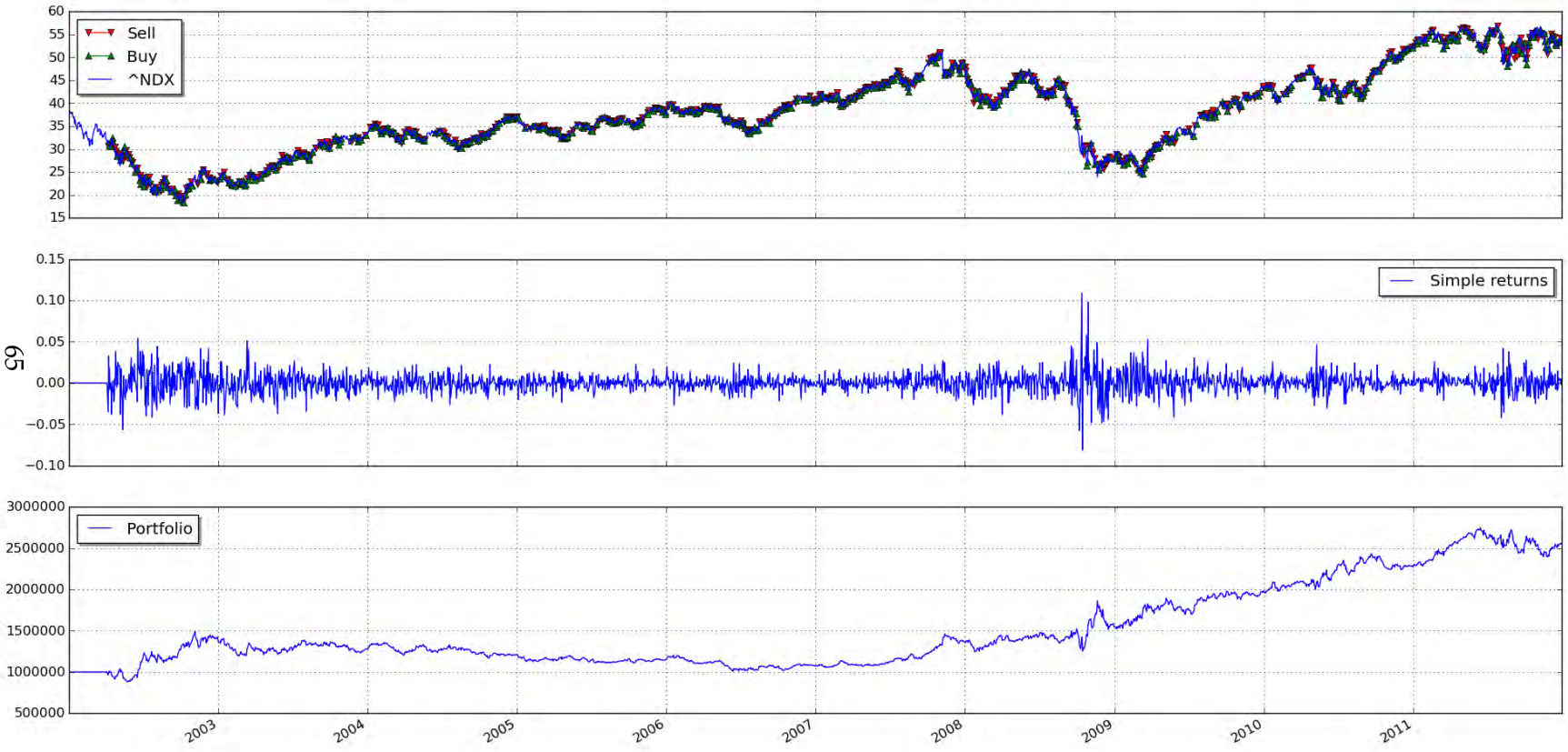

Figure 4.4. Performance of the three agent majority game model trading strategy on the NASDAQ in 2002-2011. The top figure shows the the index and when buy/sell orders where made. The middle figure shows returns for each buy/sell order. The bottom figure shows returns from the trading strategy. Memory and in-sample lengths are 3 and 60 respectively.

4.3 Three agent model - Minority Game

The three agent model majority game allowed us to reverse engineer a time series through at least one recurring event, thus considerably increasing the length of a perfectly reverse engineered in-sample period. In a very similar way, the three agent model minority game achieves the exact same goal, but with some slight differences.

Agents' strategies are created day by day and then corrected as soon as a recurring event happens with a different next day move. Agent one's strategy gets trade decisions according to the real returns. Agent two has two strategies to choose from like before. His first strategy is identical to Agent one's strategy. His second strategy is also the same, except for trades on days where the recurring event happens. On these days, he has the opposite trade decision, resulting in wrong trade decisions in all days of the recurring event, until it happens with a different next day return. At that time point he will predict correctly. Agent three gets a trading strategy that chooses the opposite trade decisions compared to Agent one.

Agent two now evaluates his strategies according to the Minority Game payoff:

$$\pi^{MG}(f_i(\mu_t), \text{sign}(r_t)) = -f_i(\mu_t)\,\text{sign}(A_t). \tag{4.10}$$

This preference changes the switching process. Since he now constantly wants to be in the minority decision, agent two prefers trading opposite to the real returns. This means that agent two's strategy one will accumulate the lowest payoff possible as it always predicts with the majority. Strategy two, however, will clearly predict wrongly at the first occurrence of the recurring event, thus getting a higher payoff for that day. Consequently, agent two will switch to strategy two after that day, and therefore predict correctly at the second occurrence of the recurring event, where the next day return is different. This sequence is quite straightforward and requires no backward looking adjustments, as long as the recurring event only happens once before the occurrence with the different next day move. When it happens multiple times, some backward looking adjustments must be made in order for agent two to switch between strategies on the correct day. Furthermore, because of this, we must also use agent three's strategy to fix the aggregate decision such that it is always in accordance with the real returns. This is also achieved with a backward looking approach as before, by reversing some of agent three's past trade decisions from being incorrect to correct.

We will use the same methods for testing this model. Specifically, the test in definition 4.3 is used for determining the model's average number of correctly predicted days in a row for different memory lengths. Thereafter, we

will test the model on trends in order to see if there is a tangible difference in average number of predicted days from the previous approach. Finally, we will reverse engineer a time series using in-sample periods of different lengths and out-of-sample periods of length one. The underlying real time series will always be the NASDAQ in 2002-2011 as before. As with the three agent majority game model, the minority game model is not very realistic in all parts of the NASDAQ. This payoff is really only realistic in times of regime shifts, when agents want to be in the minority. That should therefore also be considered when reviewing the results. Especially since the NASDAQ in years 2002-2011 is characterized to a large extent by an upwards trend. One could therefore say that the minority game model is less applicable for this time period than the majority game model.

Our results for the average number of correctly predicted days in a row are displayed in table 4.10.

Table 4.10. Average number of correct prediction days in a row - 3 agent model Minority Game. The time series is NASDAQ in 2002-2011. Results are for various memory lengths. The number of possible different events is shown for comparison.

Memory Length	Number of possible events	Avg. number of correctly predicted days in a row
3	8	7.0
6	64	20.9
8	256	44.1
10	1024	80.4
12	4096	153.9

Clearly, the results in table 4.10 show that the three agent model minority game can emulate perfectly longer pieces of real return time series than the one agent model. It does in fact yield very similar results to the three agent majority game model, as expected. The only value that differs, statistically to the 5% level, between the three agent minority- and majority models is for memory length of three. The majority model has an average of 7.3 correctly predicted days in a row while the minority model has an average of 7. The reason for the minority game's lower result is not immediately obvious, although it could be related to our earlier reasoning regarding the applicability of the model in times other than regime shifts.

Results for the same test, but with trends of varying lengths are displayed in table 4.11. The definition of trends is in definition 4.4. As before, cells are marked with green if their respective average number of correctly predicted days in a row is statistically different to the 5% level from values for no

trends (table 4.10).

Table 4.11. Average number of correct prediction days in a row - 3 agent model minority game with trends. The time series is NASDAQ in 2002-2011. Results are for various trend and memory lengths.

	Avg. number of correctly predicted days in a row		
	Trend length		
Memory Length	3	6	10
3	6.83	6.55	6.49
6	21.64	20.81	22.2
8	41.73	39.83	43.94
10	77	75.45	69.14
12	118.1	175	239

Let us now reverse engineer the NASDAQ in 2002-2011 on varying in-sample lengths, fixed out-of-sample length of one and fixed memory length of 3. The definition of this test is in definition 4.5. As before, 10.000 random strategies will make trades on the same out-of-sample days that our agents trade. We then have a better picture of how the three agent model performs in comparison to chance. These results are displayed in table 4.12.

Table 4.12. Three agent model minority game in-sample and out-of-sample success rates for different in-sample lengths. The time series being reverse engineered is NASDAQ in 2002-2011. Agent memory length is fixed at 3 and out-of-sample length is 1 day. Last column shows the proportion of the 10.000 random strategies that perform worse than the three agent model.

In-sample length	Avg. in-sample success rate	Avg. out-of-sample success rate	Random strategies worse than the model
20	70.3%	48.8%	12.1%
60	56.2%	48%	1.9%
100	53.7%	49.6%	34.4%
150	52.1%	48.7%	9.6%
200	51.5%	48.9%	13.2%

Results in table 4.12 conform with our earlier findings for the three agent minority game model. That is, average in-sample success rates, as well as out-of-sample success rates are generally lower than for the three agent majority

game model. The difference is perhaps most noticeable for out-of-sample success rates, who are now all below 50%. Consequently, the proportion of random strategies underperforming the model is also extremely low.

Statistics for the out-of-sample windows where the agents have a trade decision is displayed in table 4.13.

Table 4.13. Three agent model minority game out-of-sample statistics. The time series being reverse engineered is NASDAQ in 2002-2011. Agent memory length is fixed at 3 and out-of-sample length is 1 day.

In-sample length	Up-moves: real time series	Up-moves: predicted time series	Up-moves caught	Down-moves caught
20	53.6%	52.9%	51.5%	45.7%
60	54%	52.3%	50.2%	45.2%
100	54.2%	52.5%	52%	46.8%
150	54.4%	52%	50.6%	46.4%
200	54.5%	53.3%	52%	45.2%

Our comparison of trading strategies like before now follows. The first strategy is the three agent minority game model, the second is the average from 1000 random strategies like before, and the third is the minority game model with fees. The fourth is the buy-and-hold strategy for the entire NASDAQ (2002-2011). All strategies will be back-tested on an ETF tracking the NASDAQ between 2002-2011, with memory length 3 and varying in-sample lengths. Sharpe ratios and absolute returns are reported in table 4.14.

Values for Sharpe ratios and absolute returns are obviously generally lower for the minority game model when compared to the majority game model. The reason behind this difference might again be related to the fact that the minority game model could be said less relevant in general for the NASDAQ between years 2002 and 2011 than the majority game model.

Returns for the best performing three agent minority game model strategy (in-sample length 60) is displayed in figure 4.5.

Table 4.14. Comparison between the trading strategies. Strategies are back-tested on the NASDAQ in 2002-2011. Memory length is fixed at 3 and in-sample lengths vary. Values for random trading strategies are averages and values in brackets are standard deviations.

In-sample length		3 agent min.	Random strat.	3 agent min w. fees	Buy and hold NDX
20	Sharpe	-0.537	-0.047 (0.294)	-0.931	0.203
	Abs returns	-65.2%	7.1% (65.9%)	-82.3%	49%
60	Sharpe	0.082	-0.072 (0.289)	-0.266	0.222
	Abs returns	19.5%	1.8% (67.1%)	-37.7%	56.2%
100	Sharpe	-0.073	-0.066 (0.29)	-0.42	0.283
	Abs returns	-9.8%	3.2% (62.7%)	-52.4%	79.8%
150	Sharpe	-0.3	-0.067 (0.283)	-0.643	0.391
	Abs returns	-38.6%	2.8% (56.4%)	-66.5%	128.8%
200	Sharpe	-0.637	-0.078 (0.279)	-0.986	0.385
	Abs returns	-64.5%	1.9% (58.9%)	-80.5%	123%

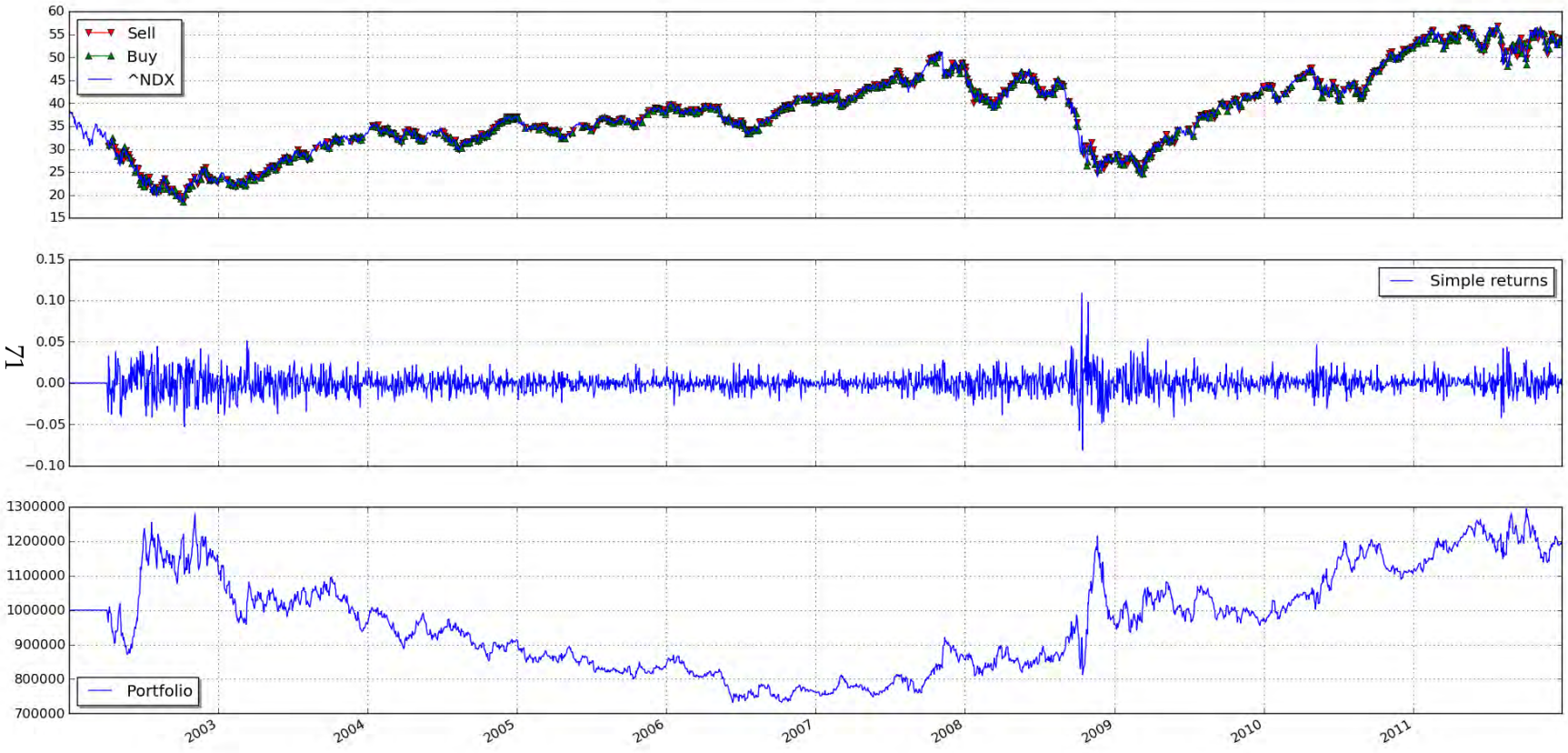

Figure 4.5. Performance of a three agent minority game model trading strategy on the NASDAQ in 2002-2011. The top figure shows the the index and when buy/sell orders where made. The middle figure shows returns for each buy/sell order. The bottom figure shows returns from the trading strategy. Memory and in-sample lengths are 3 and 60 respectively.

Chapter 5

Concluding Remarks

Although the ABM in section 2 surely undermines the weak form efficient market hypothesis, its practical benefit in real time trading remains to be seen. Our overarching goal, creating abnormal risk-adjusted returns using historic prices, spurred our efforts to further improve the model's predictions and thus increase the potential of subsequent real time trading tests. Results from a broad scoped search of reliable prediction indicators in the model's output were interesting in that parameters' relationships seemed to rely largely on the underlying asset as well as time period. For instance, out-of-sample success rates do not seem to depend on in-sample success rates for the NASDAQ in 2002-2011. However, the same cannot be said for the S&P 500 in 1992-2001, where a clear upwards trend characterizes the relationship between in-sample 1 and out-of-sample success rates. Search for prediction indicators in predicted- and real returns, number of active agents as well as trends and trend capture yielded similar results in that either there was no consistent prediction indicator across assets and time periods, or a very weak indicator. One such weak indicator is predicted returns. Our analysis of the relationship between predicted returns and out-of-sample success rates indicated a weak but fairly consistent positive relationship between them. However, it is difficult to infer any meaningful prediction indicators from this that are worth using in a real time trading test.

As our exploration of this complex agent based modeling method failed to deliver decisive prediction indicators, we somewhat digressed from our original intentions, while staying true to the overall purpose. Without impactful prediction indicators, we are unable to extend the model's "alpha" generating capability through an enhanced trading strategy. As a consequence, instead of further pursuing this model's true ability to predict asset prices, through real time trading with an enhanced trading strategy, we went back to the beginning and contemplated agents' individual decision making. Specifically, our aim was to replace the other model's computationally inten-

sive calibration process with deterministic strategy allocation that delivered high in-sample success rates. Furthermore, we suspected this method to be capable of high out-of-sample success rates as well, and thus serve our overall purpose of creating positive risk-adjusted abnormal returns.

We tested our assumptions in three different models, one of which had only 1 agent, and two which had 3 agents. The 1 agent model, although very simplistic, reported high in-sample success rates as well as the highest average out-of-sample success rate among the three models. It further managed to beat both the random strategies as well as the buy-and-hold strategy in terms of Sharpe ratios for in-sample length of 60 and 100 and memory length of 3.

The two 3 agent models either used the majority- or minority game to achieve their purpose of perfectly reverse engineering longer time series. It is however somewhat misleading to create two different models for these games as they would ideally be used in one model, which would employ them according to the current market regime. Results for success rates as well as Sharpe ratios and absolute returns for the two 3 agent models are therefore often not comparable. We find that they do however illustrate the potential of a model which combines the two games. In particular, the 3 agent majority game model consistently had out-of-sample success rates higher than 50% and beat both the random strategies as well as the buy-and-hold strategy for in-sample length 60 and memory length 3. The 3 agent minority game model reported notably worse results, likely because the minority game is not very relevant during the long up-trend of the NASDAQ in 2002-2011, when real market investors strive to be in the majority.

Overall, our simplified 1 and 3 agent models can be extended in numerous directions. We think they represent a new way of calibrating ABMs to real time-series, while delivering promising prediction capabilities.

Bibliography

Alan Agresti and Brent A. Coull. Approximate is better than "exact" for interval estimation of binomial proportions. *The American Statistician*, 52(2): pp. 119–126, 1998. ISSN 00031305. URL http://www.jstor.org/stable/2685469.

JV Andersen and D Sornette. The $-game. *European Physical Journal B*, 31(1): 141–145, Jan 2003. doi: 10.1140/epjb/e2003-00017-7.

L. Barras, O. Scaillet, and R. Wermers. False discoveries in mutual fund performance: Measuring luck in estimated alphas. *The Journal of Finance*, 65(1):179–216, February 2010.

Eric Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7280–7287, 2002. URL http://www.pnas.org/content/99/suppl_3/7280.abstract.

Markus K. Brunnermeier. *¡p¿Asset pricing under asymmetric information: bubbles, crashes, technical analysis, and herding¡/p¿*. Oxford University Press, 2001.

D. Challet and Y.-C. Zhang. Emergence of cooperation and organization in an evolutionary game. *Physica A: Statistical Mechanics and its Applications*, 246(3):407–418, Dec 1997. doi: 10.1016/S0378-4371(97)00419-6.

M. Marsili Challet D., A. Chessage and Y-C. Zecchina. From minority games to real markets. *Quantitative Finance*, 1(1):168–176, 2001. doi: 10.1080/713665543.

Fang Chen, Chengling Gou, Xiaoqian Guo, and Jieping Gao. Prediction of stock markets by the evolutionary mix-game model. *Physica A: Statistical Mechanics and its Applications*, 387(14):3594 – 3604, 2008. ISSN 0378-

4371. doi: http://dx.doi.org/10.1016/j.physa.2008.02.023. URL http://www.sciencedirect.com/science/article/pii/S037843710800188X.

Gilles Daniel, Didier Sornette, and Peter Wohrmann. Look-ahead benchmark bias in portfolio performance evaluation. (08-33), 2008. URL http://ssrn.com/abstract=1289222.

Eugene F. Fama. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 25(2):383–417, 1970. doi: 10.1111/j.1540-6261.1970.tb00518.x.

Eugene F. Fama. Efficient capital markets: Ii. *The Journal of Finance*, 46(5):1575–1617, 1991. doi: 10.1111/j.1540-6261.1991.tb04636.x.

Eugene F. Fama and Kenneth R. French. Common risk factors in the returns on stocks and bonds. 33:3–56, Feb 1993. doi: 10.1016/0304-405X(93)90023-5.

Eugene F. Fama and Kenneth R. French. Luck versus skill in the cross section of mutual fund return. *Journal of Finance*, 65(5):1915–1947, 2009. URL http://ssrn.com/abstract=1356021.

J. Doyne Farmer. Market force, ecology and evolution. *Industrial and Corporate Change*, 11(5):895–953, Nov 2002. doi: 10.1093/icc/11.5.895.

C.H. Hommes. Modeling the stylized facts in finance through simple nonlinear adaptive systems. *Proc. Nat. Acad. Sci. USA*, 99(Suppl. 3):7221–7228, 2002.

C.H. Hommes. Heterogeneous agent models in economics and finance. *Handbook of Computational Economics (Elsevier B.V.), Edited by Leigh Tesfatsion and Kenneth L. Judd*, vol. 2, chapter 23:1109–1186, 2006.

P. Jefferies, M.L. Hart, P.M. Hui, and N.F. Johnson. From market games to real-world markets. *The European Physical Journal B - Condensed Matter and Complex Systems*, 20(4):493–501, 2001. doi: 10.1007/s100510170228.

Narasimhan Jegadeesh and Sheridan Titman. Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of Finance*, 48(1):65–91, 1993. ISSN 1540-6261. doi: 10.1111/j.1540-6261.1993.tb04702.x. URL http://dx.doi.org/10.1111/j.1540-6261.1993.tb04702.x.

Narasimhan Jegadeesh and Sheridan Titman. Profitability of momentum strategies: An evaluation of alternative explanations. *The Journal of Finance*, 56(2):pp. 699–720, 2001. ISSN 00221082. URL http://www.jstor.org/stable/222579.

P. Jefferies M. L. Hart S. Howison Johnson N. F., D. Lamper. Application of multi-agent games to the prediction of financial time series. *Physica A*, 299:222–227, 2001.

A.S. Kyle. Continuous auctions and insider trading. *Econometrica*, 53(6): 1315–1336, May 1985.

T. Lux and M. Marchesi. Scaling and criticality in a stochastic multi-agent model of a financial market. *Nature*, 397:498–500, 1999.

Burton Gordon Malkiel. A random walk down wall street : The time-tested strategy for successful investing. 2003.

Matteo Marsili. Market mechanism and expectations in minority and majority games. *Physica A: Statistical Mechanics and its Applications*, 299(1–2):93 – 103, 2001. ISSN 0378-4371. doi: http://dx.doi.org/10.1016/S0378-4371(01)00285-0. URL http://www.sciencedirect.com/science/article/pii/S0378437101002850. Application of Physics in Economic Modelling.

Ariel Rubinstein. *Modeling Bounded Rationality*. The MIT Press, Dec 1997. ISBN 0262681005.

J.B. Satinover and D. Sornette. Cycles, determinism and persistence in agent-based games and financial time-series i. *Quantitative Finance*, 12(7):1051–1064, 2012.

J. Wiesinger, D. Sornette, and J. Satinover. Reverse engineering financial markets with majority and minority games using genetic algorithms. *Computational Economics DOI 10.1007/s10614-011-9312-9 (http://ssrn.com/abstract=1553821)*, Jan 2012. doi: 10.1007/s10614-011-9312-9.

Qunzhi Zhang. *Disentangling Financial Markets and Social Networks: Models and Empirical Tests*. PhD thesis, 2013.

Qunzhi Zhang, D. Sornette, and J. Satinover. Mixed-game virtual stock markets combining minority, delayed minority, majority and dollar agent-based models. *working paper ETH Zurich*, 2013.