Chapter 1

Introduction

Technological progress in telecommunications and computers over the last twenty years has created a trend towards automation of stock exchanges. Fast telecommunication networks, especially internet and mobile phones, increase the speed of information dissemination through the markets. The faster transmission of data leads to a more frequent updating of stock prices and to a larger amount of stock related information being incorporated into them, resulting in a more efficient price discovery process\(^1\). Moreover, fast connections enable remotely located entities to participate in trading, increasing the number of trading parties, traded volumes, and in general the liquidity of a market\(^2\). The concepts of market efficiency and liquidity are of fundamental importance to the markets and will be discussed in more detail in chapter 2. Computerisation of trading operations, on the other hand, speeds up stock exchange operations and allows to capture a detailed record of transactions taking place on a market. The captured high frequency\(^3\) data are gradually becoming available to researchers, revealing new insights into the aspects of trading which manifest themselves on timescales of

\(\text{\footnotesize\(^1\)The concept of market efficiency is reviewed in section 2.2.}\)

\(\text{\footnotesize\(^2\)The concept of liquidity is explained in section 2.4.}\)

\(\text{\footnotesize\(^3\)High frequency refers to temporal resolution and usually means intra-day data. The highest possible resolution is defined by a single event (transaction) and is sometimes called an ultra-high frequency. In this thesis high frequency refers to any intraday frequency, including individual transactions.}\)
hours, minutes, seconds, and single transactions. Stock exchanges with a public limit order book, such as the Australian Stock Exchange (ASX) and most of the exchanges outside the US, are order-driven markets and as such are a particularly interesting topic for study. This is due to the traditional focus of financial researchers on the New York Stock Exchange (NYSE) and other markets in the US with a quote-driven architecture\textsuperscript{4}. More importantly, however, stock exchanges with an order book capture detailed information on buy and sell orders stored in the book, and the whole contents of the book or a number of the best price levels are visible to traders. The detailed information on submitted orders and their transparency enable a precise analysis of intraday order flows, imbalances in supply and demand, and order submission strategies, as a function of the contemporaneous and past states of the limit order book. The insights obtained through scientific investigation of high frequency order book data can then be used to devise new ways of reducing the costs of trading\textsuperscript{5} by optimising the timing, size, and other characteristics of submitted orders.

This thesis analyses and builds computational models for order submissions on a market with a limit order book. We focus on a trade implementation problem faced by a trader who wants to transact a buy or sell order of a certain size. The trader would prefer to execute his trade quickly, ideally at the best price available on a given day and with minimal trading costs. He may, for example, submit a market order\textsuperscript{6} to achieve an immediate execution, or add a limit order to the order book hoping for an improved trade price at a later time. However, an immediate execution may incur an unwanted price impact cost\textsuperscript{7} if the order size is large compared to the volume stored at and near the best quotes in the limit order book. A delayed execution, on the other hand, exposes the trader to a risk of an adverse price movement and an uncertainty about future volume in the

\textsuperscript{4}Quote-driven and order-driven markets are discussed in chapter 2.
\textsuperscript{5}The costs of trading are reviewed in section 2.4.
\textsuperscript{6}Market order, limit order and some other order types are discussed in sections 2.3 and 2.5.2.
\textsuperscript{7}Price impact and other transaction costs are reviewed in section 2.4.
order book. The price impact of a large order can be reduced by breaking it up into a sequence of smaller orders and submitting them to the market over a period of time, but again, delayed orders face the risks mentioned above. The decisions about order timing, size, and type, and whether to split an order into a sequence of smaller orders, with the objective of maximising (minimising) an average price of the total sell (buy) trade and minimising the total transaction costs, are non-trivial and form a part of an optimal order submission strategy. Moreover, the multiple benchmarks for measuring transaction costs used by investors allow us to view the choice of a benchmark as another decision variable. The scope of this thesis is confined to an order-driven market, but in general, trading venues with other, traditional and innovative architectures, can sometimes be more attractive to a trader, depending on his preferences. The choice of a trading venue and the trader’s profile could thus be included as additional decision variables. The order submission optimisation problem is therefore quite complex, even in the case of a single small order in a single trading venue. This problem is of substantial economic significance to large traders, and in particular to institutional investors who frequently trade large volumes of shares representing a quarter or more of the whole market volume.

Large sets of high frequency data and inexpensive computer power which became available in recent years have created an opportunity for a data-driven research approach, also known as data mining. Data mining is a bottom-up knowledge discovery method which proceeds from low level data to a high level model. It compliments a more traditional, top-down approach. We use data mining as a tool for a systematic and statistically sound exploratory data analysis and specification search. The main strength of the data-driven approach is due to its closeness to the data and “let the data speak” attitude. Data mining can uncover new patterns or relationships where traditional methods might see outliers or noise. Sometimes it can also produce non-parametric models which are better than their existing parametric counterparts. Furthermore, the exploration
of low level data allows researchers to critically analyse the validity of constraints imposed on the data by the parametric models employed. Insights obtained from data mining may then suggest ways of refining existing models or open up entirely new avenues for addressing a given research problem. Another advantage of the data-driven approach is a capability for automatic hypothesis generation and testing, which allows the exploration of a large number of statistical hypotheses. That automation, however, does not come without a price. Testing many hypotheses on the same dataset increases the chance of finding a spurious pattern, regression, or correlation. Rigorous methods for multiple hypothesis testing must therefore be employed in order to correctly determine the statistical significance of a claimed discovery. Data mining has a long tradition in stock trading, underlying an investment style known as technical analysis⁸. More formal and scientifically sound applications of data-driven methods in finance have appeared over the last 10 years, coinciding with the dynamic development of the data mining field. The main reason for the popularity of data mining among investors is their desire for a fast and automatic discovery of profitable patterns within massive collections of financial data. The speed of data processing is essential for enabling a trader to exploit a discovered profit opportunity before the whole market reacts and the stock price is adjusted. In contrast, we apply data mining to the problem of optimal order submission, with the objective of minimising transaction costs.

To summarise, the motivation for our research on stock market order submissions stems from the recent availability of high frequency limit order book data, relative scarcity of studies employing such data, economic significance of transaction cost management, and the potential for data mining to uncover patterns and relationships not identified by the traditional top-down modelling approach.

⁸Technical analysis is discussed in section 2.2.1.
1.1 Aims of the thesis

The aims of this thesis are to analyse and build computational models for order submissions on the Australian Stock Exchange, an order-driven market with a public electronic limit order book. It needs to be mentioned, however, that the references cited in the thesis are not confined only to order-driven markets. When it seems relevant, or when the number of existing studies is small, related references for quote-driven markets may also be provided. The ASX data made available to us are a complete set of all trades and orders, transaction by transaction, with correct timestamps, as well as the full contents of the limit order book, as recorded by the stock exchange.

The focus of the thesis is on the trade implementation problem faced by a trader who wants to transact a buy or sell order of a certain size. We use two approaches to build our models, top-down and bottom-up. The traditional, top-down approach is applied to develop an optimal order submission plan for an order which is too large to be traded immediately without a prohibitive price impact. We present an optimisation framework and some solutions for non-stationary and non-linear price impact and price impact risk. The second, bottom-up, or data mining, approach is employed for trade sign inference, where trade sign is defined as the side which initiates both a trade and the market order that triggered the trade.

We are interested in an endogenous component of the order flow, as evidenced by the predictable relationship between trade sign and the variables used to infer it. We want to discover the rules which govern the trade sign, and establish a connection between them and two empirically observed regularities in market order submissions, competition for order execution and transaction cost minimisation. To achieve the above aims we first use an exploratory analysis of trade and limit order book data. We then develop a non-parametric and a parametric
trade sign model by testing multiple candidate models, while adjusting statistical significance results to account for multiple tests.

1.2 Outline of the thesis

The outline of the thesis is as follows. The background chapter (2) provides an introduction to and review of some fundamental concepts related to financial markets. The topics discussed include market architecture (2.1), market efficiency (2.2), transaction costs (2.4), and order submission strategies (2.5). The operating mechanism of a limit order book market, as implemented by the Australian Stock Exchange, is also explained (2.3). The chapter concludes with a section on data mining (2.6), which presents a short introduction to the data-driven knowledge discovery approach, challenges and pitfalls of data mining, as well as short overviews of two specific techniques used in the thesis, the self-organising map (SOM) and the k-nearest-neighbour (k-NN). The experimental part of the thesis starts with chapter 3, where we construct various order submission plans for three large stocks on the ASX. An analytic framework for minimising transaction costs is developed, and a closed-form solution is derived for simplified cost functions. To obtain solutions for the cases which are analytically intractable, we apply deterministic discrete time dynamic programming. The optimal trading plans are generated for two levels of a trader's aggressiveness. The generated optimal plans, as well as plans corresponding to some other, simpler strategies, are then evaluated by a trading simulator and compared. In the subsequent chapter (4) we undertake an exploratory analysis of trade and order book data. In particular, the self-organising map technique is employed for unsupervised clustering of multi-dimensional data to enable their visualisation. The transformation has uncovered two clusters corresponding to the areas of the SOM-processed space which are dominated by buyer-initiated and seller-initiated trades, respectively.
Chapter 1. Introduction

Chapter 5 develops a non-parametric model for trade sign inference. A local modelling method called k-nearest-neighbour is chosen for the classification task. Predictor variables are selected from among various contemporaneous and lagged trade attributes and order book data, while trade sign is the target variable. To find a classifier with the highest trade sign classification accuracy we perform a search across various predictor variable sets, training interval lengths, and values of the classifier’s parameter $k$. The performance of the k-nearest-neighbour is compared against that of the three other classifiers, linear logistic regression, trade continuation, and majority vote. The k-nearest-neighbour classifier is shown to be superior, requiring only three predictor variables and achieving an average accuracy of 71.40%, across all of the tested stocks. The best predictor variable set found for the k-nearest-neighbour classifier is then used in chapter 6 to build a parametric trade sign inference model, for the same stock data. A two dimensional projection of the three predictor variables and trade sign is constructed and allows us to discover a structure comprising six distinct regions, where each region corresponds to a particular trade sign. The boundaries between regions form a set of three straight lines, which suggests a piecewise linear parameterisation. A piecewise linear classifier is then proposed and estimated. Its classification accuracy turns out to be superior compared to the non-parametric version, with an out-of-sample value of 74.38%, across all of the tested stocks. An interpretation of the six discovered regions and the high classification accuracy is provided in terms of two behavioural regularities observed on the market, competition for order execution and transaction cost minimisation. The thesis is concluded with chapter 7. We summarise the contributions of the experimental chapters and suggest directions for future work. Supplementary material in appendices, a glossary of terms, and a bibliography are included at the end of the thesis.
Chapter 2

Background

This chapter provides background information on a number of key concepts in the areas of financial markets and data mining. The scope of the presented information is substantially broader than that actually used in the experimental chapters. In particular, we discuss types of trading venues other than an order-driven market implemented by the Australian Stock Exchange. This is done to place our work in a well defined context and to outline a rich picture of business reality for which the extended versions of the order submission problems and solutions presented in this thesis could be considered in future work. The fundamental design features of a market are referred to as market architecture. There are at least half a dozen basic market architectures employed around the world. A division into quote-driven and order-driven markets is the most popular classification, but some other architectural features are also important. Section 2.1 presents a brief introduction to this subject. More information on market design and related issues can be found in a comprehensive work by Harris [102].

As far as the operation of a financial market is concerned, the two most important characteristics are efficiency and liquidity. Their definitions, however, are not fully formalised and there are a variety of measurement methods for each of them. Market efficiency is positively correlated with the amount of public and private information reflected in stock prices. The more stock related information
is reflected in stocks’ prices, and the faster this process takes places, the more efficient a market is considered to be. Market efficiency is perhaps the single most actively studied topic in finance. It receives a lot of attention from academics and, less formally but with no less enthusiasm, from a majority of the market participants. The source of this interest lies in potential rewards from identifying an existing market inefficiency. If a particular investor finds out before the rest of the market that, for example, a stock price is going to rise, he can buy the stock before the price moves and sell it back with a profit later. Needless to say, a lot of market participants closely monitor the market and a genuine profit opportunity does not happen very often. Moreover, the time, money and the energy that need to be spent on monitoring and researching the market can erode or even completely eliminate any potential profits. We note that in this thesis we do not search for market inefficiencies with the aim of developing a profitable trading strategy. We focus instead on the issue of transaction costs, the concept of liquidity, and order submission strategies, briefly introducing them in the next paragraph. The concept of market efficiency is reviewed in section 2.2. In particular, section 2.2.1 describes three forms of efficiency. Two streams of literature critical of market efficiency, one on market anomalies and one on behavioural finance, are presented in sections 2.2.2 and 2.2.3, respectively. The last section dedicated to this topic, 2.2.4, discusses the current state of the debate.

The second fundamental aspect of a financial market’s operation is known as liquidity. In general, a stock is considered liquid if it can be easily converted into cash, and vice versa [7]. If stocks traded in a market are liquid then the market itself can be said to be liquid. However, the total number of buyers and sellers in a given market, as well as total volumes of their buy and sell orders, are always finite. Moreover, only a small part of the volume available for trading is posted at the best quoted price. This imposes a limit on a number of shares which can be traded instantaneously without incurring a price impact, a type of trading (transaction) cost. If a trader does not want to wait he will need to pay
(make) a premium (concession) in order to buy (sell) more shares than available at the best price. Such a situation is described in section 2.3 which presents the mechanism of the limit order book operation on the Australian Stock Exchange. The subject of transaction costs is reviewed in section 2.4. Two main types of transaction costs, explicit and implicit, are discussed in section 2.4.1. A popular benchmark for measuring trading costs, called implementation shortfall, is introduced in section 2.4.2, while an overview of empirical results and models is given in section 2.4.3. The problem of transaction costs has to be faced by any investor who wants to buy or sell shares, irrespective of the degree of market efficiency. These costs can be managed by formulating an optimal order submission plan (trading plan) in terms of such decision variables as order size, time horizon, trader’s preferences, a selected benchmark, and others. An introduction to the topic of order submissions is provided in section 2.5. Two important decision variables, choice of trading venue and choice of order type, are discussed in sections 2.5.1 and 2.5.2, respectively. Finally, the complex task of order submission optimisation is reviewed in section 2.5.3.

The last major topic of this chapter is data mining, presented in section 2.6. Data mining is a cross-disciplinary field and many of its concepts come from such disciplines as statistics, machine learning, databases, time series analysis, and others. We give a short introduction to the subject at the beginning of the section. A good understanding of the nature of the analysed data, probability distributions, and of potential temporal and causal relationships, is of great importance when applying a data-driven modelling approach. It is also necessary to possess a good knowledge of the proper data mining process. When used incorrectly data mining can produce spurious results, indicating, for example, a presence of a correlation or a pattern where there is none. The significance of the last problem is well recognised and the problem itself is known under the names of data snooping, data dredging, data torture, and other synonyms. Section 2.6.1 gives examples of data snooping mistakes and some measures for
avoiding them. A review of the data mining field is concluded with overviews of two techniques used in the thesis, the self-organising map and the k-nearest-neighbour, in sections 2.6.2 and 2.6.3, respectively. The chapter is concluded with a short summary (2.7).

## 2.1 Market architecture

Stock exchanges have various architectures, with some of the main distinguishing characteristics being presence or lack of intermediation, level of automation, and continuous or periodic trading [154, 155]. Intermediated markets employ market makers, dealers, or specialists, who determine price quotes and act as a counterparty in each trade. Due to the quote setting function of the dealers, those markets are referred to as quote-driven. In non-intermediated markets, trading does not involve market makers. Instead, submitted orders are stored, matched, and executed via a so-called limit order book. The whole trading process is determined by submitted orders, which is why such markets are referred to as order-driven. The Australian Stock Exchange and most of the markets outside the US are order-driven markets without intermediaries. Some markets, such as the New York Stock Exchange (NYSE), are a combination (hybrid) of both market types, and use market makers and a limit order book. As far as level of automation is concerned, a traditional trading floor populated by a crowd of traders is at one end of the spectrum, while a fully computerised screen-based system is at the other end. There is a trend, however, to augment or replace the former with the latter. The third characteristic, execution time, determines if trades are executed continuously during a trading session or only at certain points in time. The two modes are called a continuous double auction and a periodic auction, respectively\(^1\).

\(^1\)A overview of the continuous double auction and the periodic call auction, as implemented on the Australian Stock Exchange, is presented in section 2.3.
2.2 Market efficiency

The concept of market efficiency came to prominence in the mid-1960s. At that time, market efficiency was defined in terms of the informational content of stock prices, and various measures were proposed to calculate how well stock prices reflect a company’s “fundamentals”. In general, a market is called efficient if it does not allow investors to obtain abnormal returns, also referred to as beating the market, on a regular, long-term basis. Seminal papers by Fama [79] and Samuelson [193] showed that successive price changes (returns) are almost independent and follow a so-called random walk. This finding was seen as proof that a return process has no memory and therefore past returns have no predictive power for the current return and can not lead to abnormal returns.

2.2.1 Three forms of market efficiency

Early evidence seemed to confirm the efficiency of real financial markets, a view which became known as the Efficient Market Hypothesis (EMH). According to Fama et al. [83], the rapid speed of price adjustment to new information is one of the key characteristics of market efficiency. As far as types of informational content of stock prices are concerned, Fama [80] provides definitions and analyses of three forms of the EMH: weak, semi-strong, and strong. An efficient market is assumed to fully reflect in its prices all available information. The three efficiency forms are based on different information sets and a notion of unpredictability of the current price, given one of the information sets. A weak form uses historical prices only and implies that historical prices do not allow to predict the

\footnote{Fundamentals capture a financial condition of a given company and include such data as revenue, expenses, debt, a firm’s size, and others.}

\footnote{A return is the difference between a later value and an earlier value of a given variable, usually price. Values of the variable are measured at a selected time interval such as a day, month or a quarter. An abnormal return means an above-average return. There is no single definition of the abnormal return, but one popular approach defines it as an extra return above a simple strategy of buying of shares and then holding them over the selected time interval. The value of the abnormal return may need to be risk adjusted, where risk is a measure of a return’s uncertainty.
current price. Semi-strong form employs historical prices and all other publicly available information. The last of the three, the strong form, allows all stock related information, including private (monopolistic) information of well informed traders.

Much of the financial research during the last three decades of the twentieth century was devoted to testing of the three forms of market efficiency. It was recognised early on that not all traders possess the same depth of information [18], with company directors, mutual fund managers, and other insiders having access to private information not available to the market at large. Traders without privileged information act on an incomplete information set, while trading by insiders is either forbidden or strictly regulated. This would imply that stock prices do not fully reflect all stock related information and therefore the strong form of market efficiency does not hold, a conclusion confirmed by empirical research [198]. The status of the other two efficiency forms, semi-strong and weak, is less clear. Refs. [79, 158] do not see any practical value in technical trading. In contrast, Ref. [40], among many other studies, provides evidence that technical trading can generate abnormal returns. These results, however, have been confirmed by Ref. [27], and disputed by Ref. [187]. On the other hand, an automated pattern recognition approach proposed in Ref. [152] shows that some technical analysis indicators may be of practical value even if they are not profitable. Compared to trading on pure technical data, the inclusion of fundamental information and other macroeconomic variables might offer a greater chance of successful prediction. While the contradictory results for technical trading could perhaps be interpreted as evidence supporting the weak form of the EMH, there are indications that the semi-strong form does not hold [81].

---

Technical trading, or technical analysis, uses technical information to decide when to buy, hold, and sell stocks. Technical information usually refers to information other than fundamentals, such as past prices. Ref. [182] is a comprehensive handbook of technical analysis.

In Ref. [81] Fama slightly changed the definition of the weak form by allowing some non-technical variables. We use the original definitions as in Ref. [80].
2.2.2 Market anomalies

A particular stream of literature questioning the EMH has been dedicated to the discovery and documentation of so-called market anomalies. It has been empirically discovered that prices and other variables exhibit regularities which can not be explained by existing models of stock prices. Moreover, some of the anomalies appear to offer opportunities for easy profits. For example, the “January effect” refers to abnormal returns observed for small capitalisation stocks during the first trading days in January [124, 189]. It would seem plausible that if investors bought stock in late December then they could profitably sell it back in early January. Unfortunately the effect seems to apply only to historical data which precede its discovery [159]. This may be the result of data snooping (to be discussed in section 2.6.1). Seasonal effects around the turn of the week and the turn of the month have also been reported [143]. The above and other market anomalies are reviewed in more detail in Refs. [159, 196].

2.2.3 Behavioural finance

Other attacks on the EMH come from a relatively new field of behavioural finance. Research in this area focuses on investor psychology and other factors that contribute to the observed market inefficiencies but are outside the scope of traditional finance. Specifically, investors have been shown to sell rising stocks too early and to hold onto losing stocks for too long, the latter behaviour being explained as aversion to loss realisation [200]. Traders overreact to good and bad news on a timescale of over one year [64], but for shorter time horizons, between 3 and 12 months, a momentum effect has been observed, indicating an underreaction [116]. Individual investors also make below-average returns by trading too frequently, most probably due to overconfidence [19]. As far as market-wide effects are concerned, an overly bullish sentiment among investors can lead to
a prolonged overvaluation of stocks resulting in a price bubble. Market partici-
pants may be aware of the bubble but still unable to take advantage of it because
of high risk, limited capital, and other constraints [201]. The story of Internet
stocks in the late nineties of the last century serves as a good example here.
Behavioural finance has been subject to some criticism, especially by market
efficiency proponents [82], but the wealth of supporting evidence suggests that
the field has already matured [199]. The above is just a short selection of the
phenomena studied by behavioural finance. More information can be found in
Ref. [20, 106], which provide recent surveys of the field, and in Ref. [199], a
comprehensive textbook accessible to non-specialists.

2.2.4 The modern view

The main problem in testing market efficiency is caused by unobservability of ef-
cient prices, i.e. prices which would prevail if the EMH were true. To overcome
this problem researchers employ models of stock prices [44]. As a consequence
market efficiency tests are also tests of assumed stock pricing models and there-
fore become a joint-hypothesis problem [81]. Results of such tests indicating
that some anomaly exists may be due to a combination of a market inefficiency
and a misspecified model, and the split (in terms of weights) between the two
hypotheses is not obvious [81]. A detected anomaly represents a market ineffi-
ciency only if it is economically significant, which means that it can be used to
trade profitably, after paying transaction costs\(^6\) and adjusting for the associated
risk [117]. It is also argued that if information is not free than prices can not
fully reflect all available information because otherwise investors spending money
on market research would not be compensated [94]. Traders vary in the level of
information they possess and in their rationality, and the poorly informed and
irrational ones are referred to as noise traders [66]. Surprisingly, noise, mean-
ing lack of, insufficient, or incorrect information, is claimed to be the necessary

\(^6\) Transaction costs are reviewed in section 2.4.
ingredient which allows the trading process to happen [33]. Moreover, it is also known now that “unforecastable prices need not imply a well-functioning financial market with rational investors, and forecastable prices need not imply the opposite” [150, chapter 1].

2.3 The limit order book

This section presents the main principles of the continuous double auction and the call auction in the order-driven market implemented by the Australian Stock Exchange (ASX). A more thorough account can be found in Ref. [10], while a comprehensive description of order-driven markets is presented in Ref. [102]. A recent surge of interest in the electronic call auction is reflected in the collection of new and reprinted papers published in Ref. [194]. The ASX is a limit order market operating an electronic limit order book. The electronic limit order book, further referred to as the limit order book or the book, is a mechanism for collecting, storing, and matching of buy and sell limit orders submitted by market participants, as well as a mechanism for trade execution. Apart from the ASX, a limit order book without market makers is employed by most stock exchanges outside the US, for example the Paris Bourse, the Tokyo Stock Exchange, and the Singapore Stock Exchange.

The limit order book consists of two queues, called a buy (bid) side and a sell (ask) side, which store buy and sell limit orders, respectively. Each stock (security) traded on the exchange has its own order book. Limit orders are orders to trade, either to buy or to sell, with a specified size (number of shares) and a limit price. The latter is a constraint imposed on the actual trade price. Buy orders are called bids, while sell orders are called asks (offers). The limit order price constraint requires that, in the case of bids, a trade may happen at a price no higher than the specified limit price. For asks, on the other hand, the actual trade price may not be lower than the specified limit price. A bid in
Chapter 2. Background

<table>
<thead>
<tr>
<th>Sell side</th>
<th>Sell side</th>
<th>Sell side</th>
<th>Sell side</th>
<th>Sell side</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.55: S10,000</td>
<td>12.55: S10,000</td>
<td>12.55: S10,000</td>
<td>12.55: S10,000</td>
<td>12.55: S10,000</td>
</tr>
<tr>
<td>12.51: S3,000; S800</td>
<td>12.51: S3,000; S800</td>
<td>12.51: S3,000; S800</td>
<td>12.51: S3,000; S800</td>
<td>12.51: S3,000; S800</td>
</tr>
<tr>
<td>12.49: S200; B700</td>
<td>12.49: S200; B400</td>
<td>12.49: S200; B400</td>
<td>12.49: S200; B400</td>
<td>12.49: S200; B400</td>
</tr>
<tr>
<td>12.48: S300; S50</td>
<td>12.48: S50</td>
<td>12.48: S50</td>
<td>12.48: S50</td>
<td>12.48: S50</td>
</tr>
<tr>
<td>Spread &gt; 0</td>
<td>Spread = 0</td>
<td>Spread &gt; 0</td>
<td>Spread &lt; 0</td>
<td>Spread &lt; 0</td>
</tr>
<tr>
<td>12.47: B100; B200</td>
<td>12.47: B100; B200</td>
<td>12.47: B100; B200</td>
<td>12.47: B100; B200</td>
<td>12.47: B100; B200</td>
</tr>
<tr>
<td>12.46: B300; B1,000</td>
<td>12.46: B300; B1,000</td>
<td>12.46: B300; B1,000</td>
<td>12.46: B300; B1,000</td>
<td>12.46: B300; B1,000</td>
</tr>
<tr>
<td>12.44: B750; B500</td>
<td>12.44: B750; B500</td>
<td>12.44: B750; B500</td>
<td>12.44: B750; B500</td>
<td>12.44: B750; B500</td>
</tr>
<tr>
<td>12.42: B10; B50; B70</td>
<td>12.42: B10; B50; B70</td>
<td>12.42: B10; B50; B70</td>
<td>12.42: B10; B50; B70</td>
<td>12.42: B10; B50; B70</td>
</tr>
<tr>
<td>Buy side</td>
<td>Buy side</td>
<td>Buy side</td>
<td>Buy side</td>
<td>Buy side</td>
</tr>
<tr>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>(T1)</td>
<td>(T2)</td>
<td>(T3)</td>
<td>(T4)</td>
<td>(T5)</td>
</tr>
<tr>
<td>Buy 700 @12.49</td>
<td>Trade 300 @12.48</td>
<td>Trade 50 @12.48</td>
<td>Trade 200 @12.49</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2.1: States T1 to T5 of the limit order book for a fictitious stock and currency.

the order book with the highest limit price is called the best bid, while an ask with the lowest limit price is called the best ask. The best bid and the best ask have the highest trading priority. If two bids (asks) have the same limit price then the one which was entered into the book first has priority. These two rules together constitute a price and time priority rule. For reasons of either clarity or brevity limit orders in the book may be later referred to as orders, an order’s size as order size or number of shares, total number of shares as share volume or volume, and limit price as price.

The price difference between the best ask and the best bid in the order book is called the spread. If the spread is positive then the price in the middle of the spread, equal to an average of the best ask price and the best bid price, is defined and called the midpoint price. Bids and asks are entered into the limit order book by market participants throughout a trading day, with prices and sizes of their choice. Orders are stored in the book until they are amended, deleted or traded. Trading can take place continuously or at specific points in time. The
former is called a continuous double auction, while the latter is a call auction. The ASX employs a continuous double auction throughout the day, Monday to Friday, from around 10 am\textsuperscript{7} to 4 pm., while a call auction is executed at the daily open, daily close, and occasionally during the day after trading halts. During the continuous auction, a trade takes place whenever a new bid (ask) arrives with a limit price equal to or higher (lower) than the limit price of the best ask (bid) in the limit order book. Such an order, called a marketable limit order, creates an overlap between the prices of the best bid and the best ask, thereby changing the spread from positive to zero or negative. The non-positive spread triggers the order matching mechanism which will execute a trade at a price equal to the limit price of an existing order in the book. During the call auction, limit orders are stored in the book regardless of the value of the spread, and the order matching mechanism is activated at a specific point in time to match orders in the book and generate trades at a single price\textsuperscript{8}. The single market clearing price is determined in a way that maximises the traded volume [72], and results in the execution of bids (asks) with the same or higher (lower) limit price.

A trade during the continuous auction can also be triggered by submitting a market order. The market order is an unpriced order which is executed immediately against the best order, and if more volume is needed, the next best orders, on the opposite side of the limit order book, until all of its volume has been traded. A market order can therefore “walk the book”, as long as there is enough volume in the order book. On some exchanges, however, market orders are implemented via limit orders priced for immediate execution, and known as marketable limit orders. We note that throughout the experimental chapters of the thesis we do not make a distinction between market orders and marketable limit orders. Submitters of market orders are called liquidity demanders, while

\textsuperscript{7}Groups of stocks take part in the morning single price auction between 9:59:45 am and 10:15:00 am, in an alphabetic order by stock name, with groups earlier in the alphabet opening before groups later in the alphabet.

\textsuperscript{8}The call auction is also referred to as a single price auction.
submitters of limit orders stored in the book are called liquidity providers. An order, either a market order or a marketable limit order, which triggers a trade is considered a trade initiator. A trade is called buyer-initiated if the initiating order was a bid (buy order), and seller-initiated if the initiating order was an ask (sell order). The trade initiator is therefore the same as the initiator of the order which triggered the trade, which allows to use the former to determine the latter. Trades executed during the call auction are classified as exchange-initiated.

The trade initiator can be treated as another trade attribute in the form of a binary variable, besides price and size. The trade initiator variable is alternatively referred to as trade sign, trade direction, trade indicator, or buy/sell indicator. We will use the second term, trade sign, throughout the rest of the thesis. If both the trade initiating order and the matched order have the same size then this will be the actual size of the executed trade, and subsequently both orders will be removed from the order book. If sizes of the two orders are different then the trade size will be equal to the smaller of the two sizes, while the smaller order and the matched part of the larger order will be removed from the limit order book. The remaining shares of the trade initiating order will be subject to the continued matching process as long as the spread is not positive, potentially generating more trades. Once the spread becomes positive the matching process terminates and any unexecuted volume stays in the book as another limit order.

Figure 2.1 depicts states of the limit order book for a fictitious stock and currency, at the arrival of a new bid (buy order). For each side of the book, only orders at several prices closest to the spread are shown. Each rectangular cell on the buy side and the sell side represents orders at a single price in the order book. The price is shown on the left of each cell, with corresponding orders on the right. Buy (sell) limit orders are marked with B(S) followed by order size. The left-most order at a given price has time priority. The first state, T1, shows

---

A trade aggregation procedure (described later in section 5.2) is necessary for mapping trades to orders.
the book with a positive spread and the arrival of a new order. The order is a bid for 700 shares with a limit price of 12.49. After the new bid’s arrival the book’s state is changed to T2. Because of the way the new order is priced it is classified as a marketable limit order, which means that at least a part of its volume can be matched and traded immediately. A thick frame indicates an area of overlapping bids and asks, with a non-positive spread. At T2, the order matching mechanism matches orders S300 and B700, executing a trade of 300 shares at 12.48. Then, order S300 is removed, while order B700 is amended to B400. Subsequently two more trades are executed, as illustrated by the transitions of the order book through states T3 to T5.

The orders are matched in a sequence determined by the price and time priority rule. All three trades are classified as buyer-initiated. Eventually the positive spread will be restored, while the unmatched 150 shares of the new buy order will stay in the book as a bid at the price of 12.49. Consequently, the prices of the best bid and the best ask in the book will also change. It should be noted that out of the three trades generated, the first two are executed at the price of 12.48, while the third one at 12.49. The volume (size) weighted average price for the whole sequence is calculated by dividing the total dollar value by the total trade volume (total number of shares) for the three trades, and is equal to 12.4836. If the new buy order had requested no more than 350 shares, being the total volume of all asks at the price of 12.48, it could have been executed at a single price of 12.48.

The above description presents the basic operational principles of a market with a limit order book. In practice, however, stock exchanges with limit order books offer more trading options. For example, traders may be allowed to conceal the total size of large limit orders by submitting undisclosed limit orders [11, 29]. Such orders consist of two components, usually a small component whose size is visible in the order book and a larger hidden component with a size known only to the order submitter. Another type of order, called a crossing, can be
used when a buyer and a seller in a given trade are represented by the same broker [10]. As far as large price fluctuations are concerned, exchanges around the world employ additional mechanisms [131] to ensure the smooth evolution of stock prices. Some popular measures include price variation limits [96, 105], special handling rules for market orders [29, 96], and trading halts [10, 60]. Furthermore, to protect market participants against insider trading and other forms of market manipulation, authorities responsible for the securities industry enforce special regulations and conduct market surveillance [38, 42].

2.4 Transaction costs

Trades in an ideal market could perhaps be assumed to take place costlessly\(^{10}\) and instantaneously, regardless of the trade size. In a real market, however, there exist a number of factors, collectively referred to as friction [203], which impose costs and time delays as a function of the traded volume. The costs of trading, also known as transaction costs, can significantly reduce the performance of an otherwise successful trading strategy [175]. Institutional investors who frequently trade high volumes are particularly vulnerable. The climate of stiff competition in financial markets coupled with diminishing profit margins has created a need for careful management of transaction costs. Automatic tools for pre-trade cost estimation, which allow the analysis of alternative trading scenarios before actual transactions take place, are particularly sought after, but very difficult to build. The difficulty is due to the fact that some of the transaction costs, discussed below, are not directly observable. Moreover, there is no single way (benchmark) of calculating transaction costs, and such factors as an investor’s aggressiveness (risk profile), investment style, and trade complexity should also be considered.

Last but not least, if pre-trade estimates are required, then one needs to develop some forecasting models of future price fluctuations (volatility) and of future

\(^{10}\)Without any extra costs apart from the value of shares traded.
share volumes made available for trading by other market participants (liquidity), a rather hard task in highly efficient markets. A good overview of transaction costs is presented in Ref. [127].

2.4.1 Explicit and implicit costs

The costs of trading can be divided into two groups, explicit and implicit [127]. The former, explicit costs, are relatively easy to calculate and model. They may include a broker’s fee (commission), exchange fee, tax, and possibly other charges. The second group, implicit costs, comprises the spread, price impact (market impact), and opportunity cost. The spread, as explained in section 2.3, is the difference between the best ask and the best bid. Every market order, except for crossings, incurs a cost of at least the half-spread. The other two implicit costs, price impact and opportunity cost, represent the main challenge in the analysis and forecasting of transaction costs. They are not directly observable and have to be estimated through appropriate benchmarks and theoretical models.

Price impact cost can be understood as the price of immediacy which is caused by a limited stock liquidity. Specifically, price impact is a change in price resulting from an execution of a given trade, or in other words, a difference between the trade price and a hypothetical price that would have existed if the trade had not taken place [127]. If, for example, a trader wants to immediately transact more shares than there are available at the best ask (bid) in the limit order book then he needs to pay (make) a premium (concession), as is the case

---

11 We define transaction costs in terms of the mechanisms employed by an order-driven market. Corresponding definitions for quote-driven markets are very similar and therefore we cite references without distinguishing between market types. Some differences are mentioned in the footnotes.

12 In the case of a quote-driven market there are several definitions of the spread [127].

13 A crossing trade can take place if a buyer and a seller are represented by the same broker. The rules governing the execution of a crossing trade are quite complicated [10].

14 Liquidity in a market with a limit order book can be measured as volumes of orders on both sides of the book.

15 Either the best bid or the best ask, and consequently the midpoint price.

16 A similar effect is observed in the upstairs markets in the US which specialise in large trades and do not use a limit order book [125, 156]. A broker representing an investor who
for the buy order for 700 shares described in section 2.3. The change in price may also affect subsequent trades, whether executed by the same trader or not. Furthermore, the initial change, called the temporary price impact, diminishes in time until it reaches a permanent residual referred to as the permanent price impact [109, 110, 125, 137]. The temporary price impact is associated with liquidity shortages, while the permanent price impact is due to new information conveyed by the trade [125].

To avoid a large price impact, a block of shares can be broken up into a number of smaller orders to be transacted over time. A potential disadvantage of this approach, however, is caused by stock price volatility\textsuperscript{17}. Deferred trades may be subject to adverse price movements and obtain worse execution prices than earlier trades. The resulting losses are referred to as the opportunity cost and are defined as the cost of unexecuted or incorrectly timed trades [127]. It must be mentioned here that there is some hypothetical aspect in counting unexecuted trades as opportunity cost because we can not be sure how the price would have evolved if the trades had taken place earlier. To reduce the opportunity cost one could perhaps trade faster, but it would come at the expense of a higher price impact. A trade off between price impact and opportunity cost is also reflected in an investor’s choice between market orders and limit orders. A market order provides an immediate execution but no price certainty\textsuperscript{18}. A limit order, on the other hand, guarantees that the price will not be worse than the specified limit price, but the order may sometimes be executed only partially or not at all. Another problem associated with a limit order is the so-called winner’s curse [88]. If an execution does take place, but not immediately, then there is a possibility that the stock price has evolved in the direction of the limit price and beyond it.

\textsuperscript{17}Price volatility is a measure of a price’s variability.

is trying to transact a large block of shares searches around for counterparties which together could absorb the volume.

\textsuperscript{18}An example of a marketable limit order execution is described in section 2.3.
In such circumstances the limit order would be executed at the stale limit price, missing an opportunity for a better deal and incurring an opportunity cost.

2.4.2 Implementation shortfall

To calculate total transaction costs, one can compare prices of executed trades against some reference price [175]. The reference price represents a price below which one wants to buy and above which one wants to sell. There are many possible candidates for the reference price, for example, an opening or closing price [21] on a given day, a price just before or just after each trade, or a volume weighted average price (VWAP) of all daily trades [23]. If an average price difference between the reference price and the realised trade prices is positive for buys, and negative for sells, then the total transaction costs, before adding broker’s fees and other explicit costs, will be negative, indicating a good trade implementation strategy. Such method of calculating transaction costs, however, has several disadvantages. The main problem is a lack of accounting for unexecuted trades, hence the opportunity cost associated with them is unknown. Furthermore, if a trader who executes the trades knows which reference price is used to evaluate his performance he may “game” the benchmark [57]. For example, if the closing price is used as the reference price, the trader may concentrate his trades towards the end of the day, even if better prices existed earlier on that day. This again will result in an unaccounted opportunity cost. Using prices just before or just after each trade, on the other hand, does not properly capture the price impact and the opportunity cost incurred by an order implemented as a sequence of trades. The last of the mentioned reference prices, VWAP, does not allow to properly account for unexecuted trades either.
A method which overcomes the above disadvantages compares the performance of a real portfolio\(^\text{19}\) against a “paper portfolio”. The method was proposed in Ref. [209], and further developed under the name of implementation shortfall by Perold [175]. The real portfolio is the actual portfolio maintained by an investor, while the paper portfolio is a paper record of stock names and the corresponding numbers of shares. At the beginning of a measurement period the paper portfolio is assumed to have the same total value as the real portfolio. At this point in time the initial value and composition of each portfolio, that is all stock names, stock prices, and numbers of shares, are also recorded. During the measurement period the real portfolio undergoes changes, with stock shares being sold or bought, while the paper portfolio remains intact. At the end of the measurement period the performance of each portfolio is determined as a difference between the final and initial values of that portfolio. This difference represents a gain in value of a given portfolio. Subtracting the real portfolio gain from the paper portfolio gain produces the implementation shortfall which is equal to a sum of the price impact cost and the opportunity cost. The last two costs are calculated as sums of price impact costs and opportunity costs for individual stocks, respectively. A price impact cost for a given stock is found as a sum of price impacts of all trades in that stock. Finally, a trade price impact is determined by multiplying the number of shares traded by the difference between the trade price and the initial stock price in the paper portfolio. On the other hand, an opportunity cost for a given stock is equal to a product of the unexecuted (remaining) number of shares and the difference between the final and the initial stock price in the paper portfolio.

The implementation shortfall method presented by Perold [175] correctly captures price impact and opportunity cost over a whole sequence of trades that implements a particular stock order [127]. This method is not vulnerable to

\(^{19}\)A portfolio is a collection of financial instruments used for risk diversification, i.e. averaging investment risk by spreading it across various investments.
gaming, as long as the initial and final dates of the measurement period as well as other details are accurately recorded. It needs to be remembered, however, that a large implementation shortfall does not necessarily mean a bad investment strategy, while a small shortfall is not equivalent to profitability. The implementation shortfall can be high if an investor is very confident, likes to take a risk, or possesses superior information. An investment style, such as the technical or fundamental analysis, is also an important factor. A technical trader may be tracking and exploiting stock price trends over very short, daily or intradaily time horizons. His profit opportunities may be frequent but not large, and therefore the total number of executed trades and associated transaction costs need to be managed very carefully. A fundamental (value) trader, on the other hand, has a much longer investment horizon, ranging perhaps from one year to up to ten and more years. He does not trade often and any transaction costs incurred by him are insignificant relative to the long-term gains achieved by his stock picking skills.

Unforeseen events and the difficulty of a trade should also be taken into account when evaluating a particular implementation shortfall. If a private investor wants to buy a stock when its price is trending down, he may decide to concentrate his trading towards the end of the day, for example after 3.30 pm while the market closes at 4.00 pm. However, if an institutional investor unexpectedly submits a large buy order at 3.30 pm then the order’s price impact will result in a rapid price rise, and the private investor will incur a substantial opportunity cost. A comparable difficulty is faced by a trader responsible for implementing a large buy or sell order when market liquidity is low and price volatility is high. His implementation shortfall is going to be large regardless of the speed of his trading. We would thus like to stress the need for the careful application of transaction costs benchmarks by quoting from Ref. [126]: “transaction costs can not be studied in isolation but must be analysed with respect to the trader’s
underlying investment strategy”. The implementation shortfall is perhaps a superior measurement method but the ultimate judge of investment performance is profitability.

2.4.3 Empirical results and models

Empirical research on transaction costs has been mainly focused on institutional trades due to large orders (block orders) transacted by institutional investors and the associated large implicit costs. Depending on the study, various estimates of price impact have been reported. In general, such differences can be attributed to differences in stock samples, historical periods, trading venues, and benchmarks. For example, Ref. [48] finds that the permanent price impact is small and equal to 0.34% for buys and -0.04% for sells\(^\text{20}\). Corresponding values determined by Ref. [49] are higher, 1% and -0.3%, respectively\(^\text{21}\). The former study, however, calculates price impacts for every single trade, while the latter looks at block trades, or sequences of individual trades aggregated together. Interestingly, results from a number of studies support an asymmetric price impact, where absolute price responses for buys and sells are significantly different\([9, 48, 49, 51, 126]\). There is no agreement as to the source and the direction, i.e. whether the skew is towards buys or sells, of the observed asymmetry. Ref. [49] speculates that sells are more likely to be intermediated, entailing extra compensation for brokers. Alternatively, it argues that buys may have a higher informational content, hence a higher price impact. Ref. [51], on the other hand, finds that the direction of the asymmetry reflects a prevailing market sentiment during a particular historical period, with bullish (bearish) markets characterised by a higher buy (sell) price impact. An entirely different and very intriguing explanation is proposed by Ref. [195]. It suggests that buy-side money managers outsource some of their services to the sell-side and pay for these services with

\(^{20}\)The specific definition of the benchmark used can be found in the cited reference.

\(^{21}\)As above.
their clients’ money. However, the extra payments become unobservable to the clients by “bundling” the expenses with transaction costs. Furthermore, a flawed trading costs benchmark, VWAP\textsuperscript{22}, is used in order to prevent detection of higher implicit costs.

In terms of explanatory variables, Ref. [51] claims that trading costs are positively correlated with trade size and negatively correlated with a firm’s market capitalisation\textsuperscript{23}. The two variables, trade size and market capitalisation, are major determinants of trade difficulty. Other workers found a similar [125, 126] or weaker [48, 49] relationship between trade difficulty and transactions costs. In contrast, there seems to be a strong consensus that the identity of a money manager or broker who submits an order is a significant predictor of the transaction costs [9, 48, 49, 126, 156]. The identity reflects such factors as the investment style, trading strategy, skill, and the reputation for non-informational trading. Refs. [51, 119] show that orders which take several days or several brokers to get executed have higher trading costs. Price volatility and trading in emerging markets have also been associated with higher transaction costs [51, 71]. Market-wide returns and the urgency of trading, the latter reflected in the percentage of market orders used, seem to have some explanatory power as well [9].

As far as analytical models are concerned, the shape and formula of the price impact function are still an open problem. Refs. [89, 115, 125], among others, propose that price impact is a concave\textsuperscript{24} function of trade size. A similar shape has been suggested when the explanatory variable is a measure of order imbalance [128, 178]. The concavity of price impact also finds support in empirical evidence, for example Refs. [148, 153, 180]. Other studies, however, postulate functions with a convex shape [59, 77, 136]. One of the probable causes of this difference in shape may be the choice of trading venue and hence market type.

\textsuperscript{22}VWAP stands for a volume weighted average price of all daily trades.

\textsuperscript{23}Market capitalisation is a total value of all company shares.

\textsuperscript{24}The existence of a negative (non-negative) second derivative is a sufficient condition for a function’s concavity (convexity).
selected for a particular analysis. The upstairs markets\textsuperscript{25} facilitate large trades motivated by liquidity needs, and not by informational motives. The upstairs brokers know each other and work to maintain a reputation for non-informational trading by screening their clients and ensuring that the clients do not plan to take advantage of the other side of a trade. A genuine liquidity motivated trading allows them to achieve a decreasing extra impact cost per additional unit of volume, and consequently results in a concave price impact [125, 156]. On the other hand, markets where trading is anonymous, also referred to as downstairs markets, can not rely on trust and reputation. A trader who wants to transact a large order is assumed to possess some information, and the larger the volume the more valuable the information. Other market participants respond by, presumably, demanding an increasing extra premium per additional unit of volume. This results in a convex price impact function, which in the case of a limit order book market could perhaps be confirmed by the finite liquidity in the book [77, 155]. The total trading costs for upstairs and downstairs markets, however, may not be very different, but the upstairs markets probably enable otherwise unfeasible trades [156].

\section{2.5 Order submission strategies}

An investor who wants to buy or sell shares of a particular stock faces a number of choices. Assuming that a stock has already been picked the main problem to be solved is order implementation with constraints on transaction costs and trading duration. We also assume that an order is handled by a broker, whether trading for his own account or on behalf of his client. To be implemented, an order has to be submitted to a trading venue, such as, for example, a stock exchange, electronic communication network (ECN), or a crossing system. The choice of a trading venue depends on the selected stock, order size, urgency to

\textsuperscript{25}Also discussed in section 2.5.1.
transact, and other factors. A related yet different choice concerns a preferred order type. If a trader is patient he may choose a limit order, while an information motivated investor may use a market order to achieve an immediate execution. The choice of an order type is also dictated by what is offered by the selected trading venue, with possibly other options besides limit and market orders. An order requesting a small number of shares, relative to available liquidity, can be implemented by a submission of a single market order resulting in an execution of a single trade. Alternatively, if the number of shares being sought exceeds what is available without incurring an unwanted price impact, an order may be broken up into a sequence of smaller orders which are submitted for execution over a certain period of time. Development of an optimal submission plan for a sequence of orders, however, is not trivial and requires a joint optimisation with a cost function that captures price impact, risk of adverse price movements due to stock price volatility, and liquidity risk due to liquidity volatility. The following sections discuss in more detail the problems of trading venue choice, order type choice, and optimal order submission.

2.5.1 Choice of trading venue

Trading venues vary according to stocks traded on them, speed of trade execution, supported transaction sizes, degree of trader’s anonymity, source of stock price, hours of operation, and other characteristics. A particular group of stocks may be primarily traded in a market which specialises in them, as is the case for high technology stocks traded on the NASDAQ. Some stocks may also be traded in more than one market, including markets located in different countries and different time zones. A stock in a particular market is considered liquid if the

\footnote{An order can be matched against more than one of a counterparty’s orders, producing a sequence of trades with the same timestamp. We consider such a sequence as a single aggregate trade.}

\footnote{Liquidity volatility is a measure of liquidity’s variability.}

\footnote{The NASDAQ is a national market of the National Association of Securities Dealers in the USA.}
market attracts a large volume of trades and orders in that stock. It is important to know on which markets the stock is highly liquid because high liquidity is usually associated with fast trade execution and low transaction costs. The speed of execution also depends on a market’s order matching mechanism. As explained in section 2.3 where the operation of the ASX is discussed, orders during a continuous double auction are stored and compared in a continuous fashion, and any matching orders trigger an immediate trade execution, resulting in one or more trades. In a call auction, on the other hand, order matching is not continuous but periodic, and orders get matched in batches, generating a series of trades at a single market clearing price. The continuous double auction is employed throughout the whole trading day by the ASX, New York Stock Exchange (NYSE), and many other stock exchanges. In contrast, call auctions usually take place at the daily open and at the daily close, and after trading halts. Some markets, however, use the call auction over the whole day, as is the case at the Taiwan Stock Exchange [50]. Compared to the continuous double auction, the call auction has a lower price volatility and slower trade execution [50, 74, 157].

A trader who wants to transact a small order, relative to available liquidity, can achieve an immediate execution by submitting a single market order. In the case of an order which is too large to be executed instantaneously without an unwanted price impact, the trader’s action will be influenced by his motivation for trading. If trading is informationally motivated then it will have to be carried out gradually in a venue which offers anonymity. Most stock exchanges facilitate anonymous trading. Some exchanges, such as the NYSE, have a so-called downstairs market, which is anonymous\(^\text{29}\). The trader may break up the large order into a sequence of smaller orders and submit them to the market over a period of time. An anonymous and patient submission strategy will reduce price impact by hiding from other market participants the fact that all those orders

\(^{29}\)A trading floor with a crowd of traders, if present, is a part of a downstairs market which is not fully anonymous because individual traders may know one another.
were originated by the same trader [75]. A trader whose motivation is not special information but rather a liquidity need, for example a need to convert cash into shares as a long term investment, or a need to liquidate a stock position because of a shortage of cash, can submit his order to an upstairs market. Large liquidity motivated trades are facilitated by upstairs markets, with a well known one being located at the NYSE. Those markets operate on the basis of information disclosure and broker’s reputation. A broker renowned for liquidity motivated trading can find counterparties willing to trade large volume with a lower price impact than in a downstairs markets [125]. The total costs of trading in an upstairs market, however, are not necessarily lower than the costs in a downstairs market, but upstairs markets may enable otherwise unfeasible trades [156].

The revolution in telecommunications and computer technologies of the 1990s, coupled with increasing trading volumes and the increasing price volatility, as well as a growing demand for transaction costs management have lead to the creation of alternative trading systems. These new markets can be divided into two types, electronic communication networks (ECNs) and crossing systems [58]. The ECNs\(^\text{30}\) operate as virtual meeting grounds for stock brokers. They allow to anonymously search for and access liquidity pools at negotiated prices. The second type of venue, crossings systems\(^\text{31}\), are proprietary implementations of the electronic call auction. These systems also preserve anonymity, while their distinguishing characteristic is the lack of an independent price discovery process. The single execution price used by them is derived from traditional markets, but its precise definition depends on a particular system. It can, for example, be a daily closing price, an intraday price, or a volume weighted average price on the same day on a major stock exchange. The alternative trading systems operate when the traditional markets are open and after hours. Dealings on the ECNs and crossing systems are not visible to other markets, which together with the

\(^{30}\text{The oldest and largest ECN in the US is Instinet.}\)

\(^{31}\text{A well known crossing system is POSIT.}\)
guaranteed anonymity and process automation contribute to lower transaction costs for trading in these venues [58, 165]. The new venues have the potential to further reduce the trading costs by employing innovative technological solutions to better capture and match investors’ trading preferences [55]. The growing popularity of alternative trading systems, however, is believed by some to cause market fragmentation and to divert liquidity from the traditional markets [165]. There may be a need for new legislation, and the US Securities and Exchange Commission (SEC), for example, has already taken some measures to make trading in the new venues more integrated with other markets [58]. Various hybrids of the new and traditional systems already exist and it seems natural to expect that market competition and regulatory pressure will make the hybrids increasingly popular [172].

2.5.2 Choice of order type

The following is a summary and an extension of what is presented in sections 2.3 and 2.4.1, with an emphasis on features common across various markets. More comprehensive information can be found in Refs [10, 102]. As already discussed, there are two main order types, a market order and a limit order. A market order is an unpriced order which is executed immediately against the best order, and if more volume is needed, the next best orders, on the opposite side of the limit order book, until all of its requested volume has been traded. A market order can “walk the book” as long as there is enough volume in the order book. On some exchanges, market orders are implemented via limit orders priced for an immediate execution, also known as marketable limit orders. Limit orders, on the other hand, are orders with a specified limit price. They are stored in the order book until they can be executed. The limit price is a constraint imposed

\footnote{Most stock markets use a limit order book, even if a specialist is employed. In some markets the book may be visible only to the specialist. The specialist can sometimes execute a market order against another market order. In some circumstances he can also offer a price improvement, which is a better price than the best bid and ask quotes.}
on the trade execution price, where a buy (sell) limit order can be executed at a price no higher (lower) than the specified limit price.

Market and limit orders differ in execution price certainty and in execution probability. A market order provides an immediate execution but no price certainty. The lack of price certainty is caused by potential changes to the limit order book in a period between the market order submission and the trade execution. Multiple events can happen within a fraction of a second, with some examples being submissions of market orders by competing traders and possible revision of the price and volume at the best quote. A limit order, on the other hand, guarantees that the price will not be worse than the specified limit price, but the order may sometimes be executed only partially or not at all. Another problem associated with a limit order is the so-called winner’s curse [88]. If an execution does take place, but not immediately, then there is a possibility that the stock price has evolved in the direction of the limit price and beyond it. In such circumstances the limit order will be executed at the stale limit price, missing an opportunity for a better deal and incurring an opportunity cost. If a trader is not averse to the risk of non-execution then a limit order can seem more attractive than a market order because the former does not incur a price impact cost.

To date, empirical studies of order type choice and the probability of limit order execution have yielded a number of interesting results. It has been found that if one side of the order book is dominant, where the dominant side is the one with more depth, then there is an imbalance between supply and demand, and limit orders on the dominant side face a longer time to execution [14] and a higher risk of an adverse price movement leading to non-execution. Consequently, traders on the same side of the market as the dominant side of the book are more

\[\footnote{An example of a marketable limit order execution is described in section 2.3.}\]

\[\footnote{Relative to the transaction type, i.e. either buy or sell.}\]

\[\footnote{Depth is measured as the volume (total number of shares) on a given side of the limit order book, usually at a single price (best price) or at a number of prices closest to the midpoint price.}\]
likely to submit market orders to achieve an immediate execution [45, 93, 171, 185, 210]. The behaviour of buyers and sellers, however, may not be perfectly symmetrical [105, 185]. Some authors also speculate that a large volume on the opposite side of the order book may encourage the submission of a large market order [180]. It has also been reported that limit orders are submitted more frequently when the spread is large [29] and when price volatility is high [54]. As far as the probability of a limit order execution is concerned there is evidence that it is primarily determined by the distance of the limit price from the best quote. Orders closer to the best quote have a higher execution probability and a shorter time to execution [52, 151].

Apart from market and limit orders stock exchanges with a limit order book may offer more trading options. Undisclosed limit orders [11, 29], also known as hidden or iceberg orders, allow traders to conceal the total size of a large limit order. Such orders consist of two components, a small component whose size is visible in the order book and a larger hidden component with a size known only to the order submitter. The hidden component is exposed to the market gradually through execution of the visible part of the order. Undisclosed limit orders are used by submitters of large orders, like institutional investors, to protect them against a type of front running called quote matching [11, 102]. If, for example, a large buy (sell) limit order is placed in the order book, fully visible to the market, other traders might decide to take advantage of it by placing their own buy (sell) limit orders just in front of it, i.e. with a minimally higher (lower) price. The rationale of quote matchers is based on the assumption that the presence of the large buy (sell) limit order will move the price upwards (downwards), and once their initial buy (sell) orders are executed they will be able to sell (buy) their shares back with a profit. In case the price moves against them the quote matchers can sell (buy) back with only a minimal loss by trading against the large order they tried to front run. The quote matching is considered

\[^{36}\text{The exact minimum size of an undisclosed order depends on the stock exchange.}\]
undesirable by stock exchanges and undisclosed limit orders provide a measure to curb this practice.

Another type of transaction, called a crossing, can be used when a buyer and a seller in a given trade are represented by the same broker [10]. The broker can represent two different clients or be a counterparty of a single client. Two orders are crossed against one another when the broker submits them to the market and executes a trade at a price between the best bid and ask. There is a short delay between the submission and trade execution to give other market participants a chance to participate in the trade [10]. The crossing trade offers a low price impact cost for both trading parties. Some other types of orders, such as a stop order and a stop limit order, are equivalent to a “latent” market order and a “latent” limit order, respectively, which become activated once a stock price reaches a pre-determined level. More complicated conditions activating the contingency orders can be envisaged, involving price, time of day, and other factors. The actual monitoring and evaluation of the conditions is performed by a broker or the stock exchange’s computer system. The progress in computer and software technology gives traders a choice of increasingly sophisticated order types. As an example we would like to mention an interesting generalisation of a market and a limit order presented in Ref. [191]. The two order types are considered as extreme cases of a trading preference profile, and by adopting a fuzzy logic framework a continuum of preference profiles between the two extremes becomes available. The proposed generalisation provides a much more precise tool for expressing trading preferences and allows for an improved optimisation of the trading process.

2.5.3 Optimal order submission

The process of order submission can be optimised in respect to a variety of criteria including transaction costs, trading speed, trade difficulty, risk, investor’s aggressiveness, and others. The optimisation should produce an optimal trade
plan for a single order, for one or more stocks. If the order is large than it may
need to be broken up into a sequence of smaller orders submitted over a period
of time. The break up is motivated by a desire to limit the price impact and to
hide the large order from other market participants. The length of the trading
period or the trade plan duration should be as short as possible, while achieving
a preferred trade off between price impact and opportunity cost. Specifically,
small orders may be expected to be executed within a single day, while large
orders may take several days or longer. The duration can be an arbitrarily im-
posed parameter or it can be optimised to satisfy selected performance criteria.
The optimisation of the process of stock position accumulation or liquidation is
of particularly great interest to institutional investors, due to high transaction
costs resulting from the large volume of their trades. That interest is also mo-
tivated by the fact that institutional investors look after other people’s money
and therefore have more stringent rules of management and operation compared
to an individual investor risking his own money. The problem of optimal order
submission, however, is quite complex and even a modest goal of a single order
optimisation for a single stock in a single trading venue is not an easy challenge.
There already exist several dozen academic and professional press papers devoted
to various forms of the problem, varying in the scope of simplifying assumptions
and the resulting level of realism. The state-of-the-art solutions are believed to
have considerable economic value and are probably kept confidential until their
potential has been realised. Nonetheless, the published solutions, some of which
are discussed below, are becoming increasingly sophisticated and offer valuable
insights into the trading process. They also produce benchmarks against which
the performance of methods used in business can be compared.

As mentioned earlier, a single order for a single stock can be transacted imme-
diately by submitting a single market order. The price impact of the trade will be
minimal (half-spread) if the order size does not exceed the volume available at the
best quote. Alternatively, a limit order can be posted, offering a chance of a zero
price impact, but incurring a risk of partial execution, no execution at all, or an execution at the stale price (winner’s curse). Some theoretical models of choice between a market order and a limit order can be found in Refs. [46, 88, 173]. It has been reported that limit orders submitted at the best quote or inside the spread have lower transaction costs than market orders [103]. Similarly, Ref. [93] recommends placing buy (sell) limit orders at the best bid (ask) as an optimal strategy for minimising the implementation shortfall. An optimal submission strategy depends on the probability of order execution, risk of transacting with a better informed counterparty, and order price [108]. The strategy also depends monotonically on the private value of a stock as determined by a trader [108]. As far as order timing is concerned, Ref. [41] argues that small, liquidity motivated trades achieve the best price when executed during the morning single price auction. Ref. [176] analyses a choice between a market order and a marketable limit order, and finds that investors choice of an order is driven by lower predicted transaction costs. Moreover, a strong preference for price impact minimisation of market orders is reported by Refs. [34, 36]. An innovative generalisation of an order type is presented in Ref. [191] which considers a market order and a limit order as two extreme cases of a trading preference profile. A proposed fuzzy logic framework offers a continuum of preference profiles between the two extremes and provides a much more precise tool for expressing trading preferences, enabling a better optimisation of the trading process.

Orders which are too large to be executed via a single market order without an unwanted price impact, or via a single limit order without being exposed to quote matching, can be broken up into a sequence of smaller orders that are submitted to a market over a period of time. Smaller orders will have a lower price impact, but delayed execution will expose them to potential adverse price movements and a possibility of incurring an opportunity cost. The problem of generating an optimal trade plan which will achieve a desired balance between price impact and opportunity cost belongs to a class of multiperiod decision
problems. Such problems have been studied by the field of operational research (operations research) [207] since the second world war. An optimisation method particularly suitable to tackling the large trade problem is called dynamic programming [24, 155]. Some of the theoretical and empirical results obtained with this method are presented in Refs. [25, 26, 56, 101]. Specifically, Ref. [101] derives optimal submission plans for small trades without price impact, for three types of traders motivated by liquidity, information, and value, respectively. Ref. [25] demonstrates that an optimised strategy can yield up to 40% lower transaction costs compared to a naive approach of dividing a single order into a number of smaller orders of the same size. Ref. [26] considers a portfolio of stocks, while Ref. [56] incorporates non-stationary price impact and liquidity risk into the objective function. Despite its name, the dynamic programming is often used for static optimisation, i.e. the obtained solution is rigid and does not allow a trader to respond to changing market conditions. More recently, however, new algorithms have been developed which enable path dependent optimisation, where a trading plan is conditional on a price trajectory observed during its implementation [43, 138].

A number of optimal closed-from solutions have also been obtained analytically [16, 17, 107, 135]. Ref. [16] has incorporated non-linear price impact and liquidity risk into the objective function. In a related approach, Ref. [107] turns the trade duration into an endogenous parameter, but assumes a constant speed of trading. An analysis for a portfolio of stocks is provided in Ref. [17], while Ref. [135] presents optimal trading plans for cases with single and multiple stocks, with the trade duration dependent on order size. Analytical solvability often requires that simplifying assumptions are made, rendering a resulting solution less realistic. Nonetheless, the closed-form solutions provide interpretable insights into the mechanism of the trading process and serve as reference points for other methods. The correlation between market volume and stock price [47, 53, 120], as well as intraday variations of trading volume [1, 146] are empirically confirmed.
factors which should also be taken into account when constructing an optimal trading plan. Ref. [134] seems to meet both requirements by allowing for price-volume correlation and by using a volume-weighted average price as a benchmark. The paper’s analysis, however, applies to small trades only due to the assumed zero price impact. Some other interesting approaches include an optimal split of an undisclosed order into an exposed and hidden component [78], and an innovative framework with trade preference profiles for an alternative trading system [55]. On a different level, Ref. [119] reports that institutions are “sophisticated traders who adjust their trading strategies to accommodate differences in market characteristics”. Ref. [165] finds that, compared to the traditional market, simulated transaction costs for stocks traded through alternative trading systems are lower. A regulatory aspect of price impact minimisation, presumably referring to a potential stock price manipulation, is mentioned in Ref. [155]. More studies on optimal order submission can be found in Refs. [111, 163, 208], but their discussion is beyond the scope of this chapter.

### 2.6 Data mining

Data mining is a data-driven, bottom-up, inductive knowledge discovery method. As a scientific and engineering field, data mining is located at the interface of several disciplines, primarily statistics, machine learning, and computer databases. A data-driven approach to knowledge discovery is at least as old as the hypothesis-driven, deductive approach. However, massive databases and high speed computers, which have become available recently, have created a qualitatively new opportunity and a demand for the data-driven methods. An opportunity lies in the automated data mining tools that can be run on the fast computers and search through large databases. Given enough computer power and ingenious algorithms, this process should yield some valuable knowledge
nuggets. At the same time, the exponential growth in the amount of information produced by our IT society leads to an information overload. The existing databases are already so vast that it is beyond human ability to analyse them manually. Consequently, an opportunity offered by the data mining tools is, and will continue to be, paralleled by a high demand for them.

As far as scientific and engineering literature is concerned, the interdisciplinary character of data mining means that there is not a single reference work which provides a comprehensive treatment of the subject. There exist, however, a number of good textbooks presenting the field from the perspective of a particular discipline. For example, an introduction to data mining based on statistics is provided in Ref. [98]. The database-centric view, on the other hand, is presented in Ref. [97], while the third discipline, machine learning, is the framework of the approach employed in Ref. [216]. Reports of practical applications, in a variety of fields, are published in the proceedings of at least half a dozen conferences dedicated to the subject of data mining. The cross-disciplinary approach employed in this thesis can be found in more specialised conferences, dedicated to finance and such fields as data mining, supercomputing, neural networks and others. To date, some of the major conferences of that type have been Computational Intelligence for Financial Engineering [112, 113], Computational Finance [2], and Neural Networks in the Capital Markets [188, 214]. The remaining part of this section discusses the problem of data snooping, and two techniques used in the thesis, the self-organising map and the k-nearest-neighbour.

### 2.6.1 Data snooping

The field of data mining offers powerful tools for data-driven knowledge discovery. The promise of these tools, however, is matched by a potential for their incorrect application leading to spurious findings and illusory profits. The naive or unskilled practice of data mining is referred to by various authors as data
snooping, data dredging, data fishing, data torture, and other names. The pitfalls of data snooping were recognised long ago in the discipline of statistics. Interestingly, until recently some disciplines used the term “data mining” as equivalent to data snooping. This can still be the case in finance and econometrics [87, 161]. Probably the best known example of data snooping is overfitting without proper out-of-sample evaluation and without accounting for multiple tests. If a large number of models is constructed, or a model is very complex (flexible), or a model is built in a data-driven way, it is possible to find a good fit to virtually any data, including purely random fluctuations [99, 149, 161, 215]. This well known problem has been addressed by developing significance tests for multiple hypotheses [177, 184], resampling methods like bootstrap [204, 215], criteria for calibrating model complexity [31], out-of-sample tests [31], and cross-validation [31]. Recently available massive datasets pose new statistical challenges. When the number of data points in the studied sample is of the order of millions or billions then it is relatively easy, due to the amount of data points, to show that some hypotheses are or are not true with a statistical significance of 0.05 [92]. Consequently, when we test whether, for example, two means are not equal it may be more productive to analyse the practical significance of the difference between the means, or to adjust the required statistical significance level downwards below 0.05. Another problem is somewhat in contrast to the previous one. If the number of measured variables in a dataset reaches hundreds or thousands, then one would need an astronomical number of data points, more than any database can hold, in order to properly capture all joint probabilities in the input space [99]. This curse of dimensionality means that an analysis of a high-dimensional database may find spurious correlations or patterns, due to missing data.

The literature of statistics is rich in descriptions of, and remedies for, many other pitfalls awaiting a data analyst. This knowledge can greatly benefit the mainstream of the data mining field [92]. As far as financial data are concerned
there are also a number of unique challenges which need to be taken into account to avoid data snooping and inflated claims. As we discussed in the section on market efficiency (2.2) genuine profit opportunities are rather infrequent. Profitable strategies have a limited lifetime because imitation by other market participants leads to the erosion and eventual disappearance of abnormal returns. This is why banks and financial advisors use a disclaimer that past performance of a stock is not an indicator of its future performance. As a consequence, no amount of data mining performed on historical data can guarantee that a discovered strategy will work in the future. Data mining can become more realistic by including transaction costs [161], risk level, as well as quality and scope of information available when an investment decision is made. The speed of information transmission may also be important, if one wants to operate on an intraday timescale. A comparison against some simple strategies, such as buy and hold, should reveal if the new approach is worth the effort. Testing on simulated data, on the other hand, and on purely random data in particular, may detect hidden biases [197]. These can result from a limited dataset, collected when the market was trending up or down. Moreover, if the proposed strategy seems profitable when tested on the random data, one should suspect that the algorithm looks up future values of the time series. The risk level mentioned above is used to adjust the rate of return so as to compare it against strategies with a different level of risk [161]. More importantly however, the risk determines how much money one should be prepared to lose while pursuing a particular strategy. A large potential profit tomorrow will be no consolation for an investor who cannot maintain his stock position today because he ran out of money and is broke. Last but not least, a trading strategy needs to make sense when interpreted in terms of financial mechanisms operating on a given market [161]. Without a sound theoretical explanation, the discovered strategy may be just a result of some fleeting pattern specific to the studied dataset.
2.6.2 The self-organising map

The self-organising map (SOM) is a biologically inspired method that combines data projection and data quantisation. It implements “a nonlinear, ordered, smooth mapping of high-dimensional input data manifolds onto the elements of a regular, low-dimensional array” [133, chapter 3]. This technique is used in chapter 4 for unsupervised clustering of trade level data. It was conceived by Kohonen in the early 1980s, and since then has attracted a lot of interest, as evidenced by the bibliography with over 5,000 papers for the period until the end of 2001 [122, 170]. The seminal work by Kohonen [133] is a comprehensive reference on the subject, while a short introduction is given in Ref. [132]. The SOM achieves a simultaneous projection and quantisation of input data by mapping them onto an output space represented by a grid of interconnected nodes, according to a certain set of rules. The grid is usually a two dimensional lattice, with square or hexagonal interconnections between the nodes. Each node in the grid has a reference (prototype, codebook) vector associated with it, which has the same dimensionality as the input space. Data vectors from the input space are quantised into reference vectors in the nodes of the output grid. However, the quantisation procedure takes into account the neighbourhood relationships in the grid, applying a type of smoothing kernel over a particular node and its neighbours. This procedure promotes mapping of similar input vectors onto the same node or neighbouring nodes in the output space, leading to some preservation of the topology of the input space. The resulting map can be thought of as a clustered similarity graph [132].

There are two methods for learning a SOM, or calculating reference vectors, an incremental method and a batch method. The incremental method is the original SOM algorithm. It computes reference vectors in an iterative fashion, according to a number of formulae given by Kohonen [132, 133]. The main
Chapter 2. Background

The updating rule has the following form:

\[ \mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{c(x),i}[\mathbf{x}(t) - \mathbf{m}_i(t)] \] (2.1)

The values of the input vector \( \mathbf{x} \) and the \( i \)-th reference vector \( \mathbf{m}_i \) at a discrete time step \( t \) \((t = 0, 1, \ldots)\) are denoted as \( \mathbf{x}(t) \) and \( \mathbf{m}_i(t) \), respectively. Both vectors belong to the input space, \( \mathbf{x} \in \mathbb{R}^n \) and \( \mathbf{m}_i \in \mathbb{R}^n \). The neighbourhood function \( h_{c(x),i} \) acts as a smoothing kernel and should satisfy the condition \( h_{c(x),i} \to 0 \) when \( t \to \infty \). The function has the general form of \( h_{c(x),i} = \alpha(t)h(d,t) \), where \( \alpha(t) \) is the learning rate, while \( d = \|\mathbf{r}_c - \mathbf{r}_i\| \) is an Euclidean distance between two vectors, \( \mathbf{r}_c \in \mathbb{R}^2 \) and \( \mathbf{r}_i \in \mathbb{R}^2 \). These vectors represent locations on the output grid of the nodes associated with the reference vectors \( \mathbf{m}_c(t) \) and \( \mathbf{m}_i(t) \), respectively. The subscript \( c = c(x) \) identifies the reference vector \( \mathbf{m}_c(t) \) that is closest to the input vector \( \mathbf{x}(t) \):

\[ \forall i : \|\mathbf{x}(t) - \mathbf{m}_c(t)\| \leq \|\mathbf{x}(t) - \mathbf{m}_i(t)\| \] (2.2)

The neighbourhood function \( h_{c(x),i} \) controls the smoothness of the mapping via the factor \( h(d,t) \), while the learning rate of the SOM is determined via \( \alpha(t) \), where \( 0 < \alpha(t) < 1 \). Two popular functional forms of \( h_{c(x),i} \) are the Gaussian function and a rectangular function. The Gaussian \( h_{c(x),i} \) is defined as:

\[ h_{c(x),i} = \alpha(t)\exp \left( -\frac{d^2}{2\sigma^2(t)} \right) \] (2.3)

The symbol \( \sigma(t) \) represents the width of the smoothing kernel. Both \( \alpha(t) \) and \( \sigma(t) \) are monotonically decreasing with time \( t \), regardless of the functional form of \( h_{c(x),i} \). This means that the fastest learning rate and the largest local neighbourhood are used at the beginning of the learning process. As the learning progresses, the learning rate and the size of the local neighbourhood are gradually decreased, so as allow the SOM to converge to a solution. The second popular form of \( h_{c(x),i} \), the rectangular function, is defined simply as \( h_{c(x),i} = \alpha(t) \) when

\[37\text{Our symbols and notation, used in the SOM description and formulae, are the same as, or very close to Kohonen’s [132, 133].}\]
\( d < d(t) \), and \( h_c(x) = 0 \) otherwise, where the function \( d(t) \) is a monotonically decreasing distance from the node \( r_c \). Consequently the rectangular function creates a local neighbourhood within which all nodes are assigned the same weight, and the size of this neighbourhood shrinks with time.

The second method for learning a SOM is the batch method. This method is much faster than the incremental method described above, and it does not require the learning rate \( \alpha(t) \). The corresponding formulae and the derivation of the batch method can be found in Refs. [132, 133]. An interesting question concerns the initial values of the reference vectors, \( m_i(0) \). In principle, the self-organising properties of the SOM allow the use of any initial values, but in practice a faster convergence can be achieved if the initial values are derived from the input vectors \( x \). Some options here include random samples from the input dataset, or regular sampling from a subspace defined by the two largest principal components of the input space [132].

When used with a two dimensional output grid, the multi-dimensional data transformed by the SOM can be visualised on a plane. Moreover, the quantisation, or clustering, function of the SOM emphasises the major structures present in the input data. The level of detail, or the resolution, of the clustered image depends on the number of nodes in the grid. These characteristics make self-organising maps a useful tool for exploratory data analysis [67, 121, 211]. However, even though the algorithmic description of the SOM transformation is not difficult, the mathematical analysis of its properties, such as, for example, convergence or learning rate, has not been presented yet, due to the heuristic nature of this technique [32]. Consequently the data experiments conducted with the SOM are primarily of a qualitative nature. Some improvements and extensions to the original SOM technique have been proposed by various authors. Ref. [212] develops an improved clustering algorithm, where data are pre-processed with the SOM before being subject to a clustering procedure. A similar goal is pursued in Ref. [130]. Some statistical methods employing bootstrap for testing the
quality of the SOM transformation are developed by Ref. [63], while Ref. [186] devises an extended self-organising map that can grow dynamically and encode hierarchical relationships among data. An alternative to the SOM, based on theoretical principles and free of some of its predecessor’s deficiencies, is proposed in Ref. [32]. As far as practical applications are concerned, references for specific fields can be found in the compiled SOM bibliography [122, 170]. The applications are very diverse, and include, among many others, remote sensing [213], gene expression analysis [168], and corporate bankruptcy analysis [123].

2.6.3 The k-nearest-neighbour

The k-nearest-neighbour is a local non-parametric memory based modelling technique that can be used for classification and regression. This technique is used in chapter 5 to develop a non-parametric model for trade sign inference. There exists a large literature devoted to the study of its properties and various extensions. The most comprehensive treatment to date is provided in Ref. [62]. It contains a survey of almost 140 studies, but was published some 15 years ago. A more recent overview of the nearest-neighbour method is presented in Ref. [73]. The basic version of the k-nearest-neighbour technique requires that only the number of neighbours $k$ is specified and that the distance metric $M$ is defined. No assumptions about data distribution, nor the type of relationship between input (predictor) variables and an output (target) variable, are made. Models are built by simply storing all training data points, comprising input vectors $x_i, i = 1 \ldots p$, and their associated output values $y_i$, in the model’s memory. Computations are deferred until a new input vector $x'$ is to be evaluated. The evaluation consists of finding in the model’s memory $k$ vectors $x_j$ that are closest to the vector $x'$, and using their associated output values $y_j, j = 1 \ldots k$, to calculate the output value $y'$. The closeness is defined as a distance in $n$ dimensional space occupied by the input vectors $x_i, i = 1 \ldots p$, according to the selected distance metric $M$. The output value $y'$ can be calculated as, for example, an average of the output values
$y_j, j = 1 \ldots k$, if the output variable is continuous and the model is a regression. In the case of a classification model, the output variable is discrete, and $y'$ may be set to the most frequent value among the chosen $k$ output values $y_j$. The fact that all calculations are deferred to the evaluation stage is the reason why this technique belongs to the class of so-called lazy learning methods.

The basic formulation of the k-nearest-neighbour given above is straightforward and easy to implement. The $k$ vectors closest to a given vector $x'$ represent a local neighbourhood of $x'$. The size of the neighbourhood is fixed to $k$ neighbours. In terms of dimensional units however, the size of the local neighbourhood varies. This is in contrast to kernel methods, where the size of the kernel, or the local neighbourhood, is fixed in dimensional units, but varies if measured by the number of vectors which occupy it. The concept of the local neighbourhood is quite intuitive, at least for low dimensional space ($n \leq 3$) and the Euclidean distance metric. As we have already mentioned, the parameter $k$ determines the size of the neighbourhood, and hence how local and how smooth the mapping between the input and the output is. Low values of $k$ result in small neighbourhoods and flexible, detailed mappings, but may sometimes not generalise well to new data, due to overfitting. Larger values of $k$, on the other hand, sacrifice local features for smoother mappings. The choice of $k$ therefore, influences the bias/variance tradeoff. The second parameter for the basic k-nearest-neighbour technique is the distance metric $M$. The metric determines how a distance between two vectors is calculated. Some popular metrics include the Euclidean ($L_2$), Manhattan ($L_1$), and Minkowski distance ($L_k$). The problem of constructing metrics with weighted dimensions or features, as well as the challenges of distance measurement in high dimensional spaces ($n > 10$), are the subject of ongoing research [13, 28, 70]. As far as the classification performance is concerned, Ref. [61] showed that the error rate of one-nearest-neighbour classifier is less than, or equal to, twice the Bayes error rate, for the case with unlimited training data. A more recent analysis can be found in Ref. [73].
The main disadvantages of the k-nearest-neighbour are memory and computational time requirements, as well as poor interpretability. In its basic form this technique does not create any summary of the data. All \( p \) training data points are stored in the model’s memory. This may not be an issue for small \( p \), like several hundred or several thousand. There also exist memory technologies for storing much larger datasets. External memory devices, in the form of magnetic and optical disks and tapes, can readily handle gigabytes and terabytes of information. These devices, however, are too slow to be used as primary storage for intensive data processing. Data mining applications in particular require that all heavily used data reside in the system’s fastest memory, known as RAM. The size of RAM is usually much smaller than the size of an external memory device. This is because the cost of RAM, per unit of stored information, is significantly higher than the corresponding cost of memory disks or tapes. Processing a dataset which is larger than the available RAM therefore requires access to an external memory, which slows down the whole process and can make it unfeasible. The second disadvantage of the k-nearest-neighbour is long computational time. This is caused by the “lazy” approach to model building that defers all computation to the evaluation time. An evaluation of a new vector \( \mathbf{x}' \), for the simplest case of a model with \( k = 1 \), entails calculating \( n \) dimensional distances to all \( p \) vectors in the model’s memory, and retrieving the vector with the shortest distance to \( \mathbf{x}' \). The computational time is thus \( O(np) \) for a single input vector, and \( O(np^2) \), or quadratic in \( p \), for \( p \) vectors.

Many extensions to the basic k-nearest-neighbour technique have been proposed to improve its characteristics, especially the memory and computational time requirements discussed in the previous paragraph. To reduce the computational time during new data evaluation, it is desirable to limit the number of visited data points, so that a retrieval of \( k \) nearest neighbours does not entail scanning the whole contents of the model’s memory. A natural solution involves
Chapter 2. Background

shifting some of the computational load from the evaluation phase to the training phase. This can be achieved by structuring the model’s memory as a search tree, or an index, resulting in faster data retrieval [22, 69, 95, 160]. The distance calculation algorithm can also be modified to calculate partial distance only [73]. Further improvements in speed can be realised by decreasing the size of the training dataset. For this purpose various authors suggest data reduction by compression, editing, or clustering [91, 95, 139, 190]. Approximate methods may also offer some solutions, for both memory and computational time management, although at a potential cost of an increased error value (rate) [69, 114, 167].

The increase in the computer power and in the size of RAM memory, theoretical advances, as well as the simplicity and flexibility of the k-nearest-neighbour technique have contributed to its great popularity. Apart from being a primary solution for a particular task this non-parametric method may be employed for the initial exploration of the problem, and also as a benchmark against which to evaluate other methods. There exist a large number of papers reporting diverse applications of the k-nearest-neighbour, and we would like to mention only a few of them. This method has been successfully applied to forecasting chaotic time series by Refs. [85, 118, 140]. Good results have also been obtained for various financial time series, such as, for example, forecasting currency exchange rates [90] and stock indices [86, 218]. Some other interesting applications include handwritten character recognition [202], avalanche forecasting [183], and Internet site classification [141].

2.7 Summary

This chapter provided a background information on a number of key concepts in the areas of financial markets and data mining. In particular, we discussed such topics as market architecture, market efficiency, transaction costs, and order
submission strategies. The details of the limit order book operations, as implemented by the Australian Stock Exchange, have also been presented. The last section of the chapter was dedicated to data mining, with a focus on the problem of data snooping and two techniques used in the thesis, the self-organising map and the k-nearest-neighbour. The next chapter is the first of the four experimental chapters in this thesis. It presents an order submission optimisation framework.
Chapter 3

Optimisation of order submission

In this chapter we develop order submission plans for three large stocks on the Australian Stock Exchange (ASX). An analytic approach for minimising transaction costs over a single day is developed first. The approach allows to balance price volatility risk against price impact and price impact risk. A closed-form solution is derived for simplified transaction cost functions. Next, to obtain solutions for the cases which are analytically intractable, we apply deterministic discrete time dynamic programming. The optimal trading plans are generated for two levels of a trader’s aggressiveness. The generated optimal plans, as well as plans corresponding to some other, simpler strategies, are then evaluated by a trading simulator and compared.

We consider the problem of optimal order submission on the Australian Stock Exchange (ASX), which is an order-driven market with a limit order book. We seek to minimise transaction costs for a stock order realised over a single trading day, balancing price volatility risk against price impact and price impact risk\(^1\). The optimisation method we propose is an extension of the framework developed by Almgren [15, 16], who in turn built on an earlier work described in Refs. [17, 25]. The original framework and our extensions of it are described in section 3.1. A survey of transaction costs in the US context is provided in

\(^1\)Price impact risk is a measure of a price impact’s uncertainty.
Ref. [127]. As far as other studies on optimal order submission are concerned, Ref. [111] minimises the mean and variance of the cost of buying a block of shares, where the objective function is similar to the one in Ref. [16], but refers to the total cost of trading, not only transaction costs. A solution for a non-stationary price impact function is provided as a recursive formula. However, the model in Ref. [111] does not include price impact risk when price impact itself is stochastic. Ref. [107] adopts the transaction costs model developed in Ref. [17], but changes the objective function to match the Value at Risk (VaR) framework. The authors point out that the traditional VaR framework does not include price impact cost and liquidity constraints. By incorporating these factors they formulate a new framework, called Liquidity-adjusted Value at Risk (L-VaR). They obtain closed-form solutions for optimal execution time with linear and non-linear price impact costs. The main limitation of their model however, is the assumption that the trading speed is constant. Numerical examples with L-VaR and traditional VaR calculations are presented, suggesting that the traditional approach overestimates the risk for very liquid stocks, while underestimating it for illiquid stocks. A stochastic price impact model is also considered, with solutions found through numerical methods. More information on transaction costs and optimal order submission can be found in the reviews in the background chapter, in sections 2.4 and 2.5, respectively.

We propose an order submission optimisation method which is an extension of the mean/variance minimisation framework presented in Ref. [16]. We extend it by allowing intraday non-stationarity in price volatility, price impact, and price impact risk. A closed-form solution is derived for simplified transaction cost functions. The full formulation, however, is analytically intractable. Numerical solutions are obtained via deterministic discrete time dynamic programming [162]. Optimal trading plans are generated for three large stocks on the Australian Stock Exchange, for two levels of the trader’s aggressiveness. To compare performance, we also generate trading plans for three other trading
strategies, referred to as one-interval, uniform and VWAP\textsuperscript{2}. The performance of trading plans is evaluated by a trading simulator. The results are compared using two benchmarks, execution shortfall and distance from market VWAP. Finally, we show the connection between the mean/variance optimisation and the Value at Risk (VaR) approach.

### 3.1 An optimisation framework

The order submission optimisation method which we present is an extension of the framework presented by Almgren in Ref. [16]. We will therefore briefly review the approach proposed by that author. The problem to be solved concerns submission of a sell\textsuperscript{3} order in a way that minimises execution shortfall. The original order for \(X\) shares is broken up into a sequence of \(N\) orders which are submitted to the market over a period of time \(T\), within equal time intervals \(\tau\), where \(\tau = T/N\). The sizes of \(N\) orders to be traded within intervals\textsuperscript{4}, denoted as \(n_1, \ldots, n_N\) need to be determined so that the assumed objective function is satisfied. The trade execution shortfall \(sf\) is defined as follows,

\[
    sf = XS_0 - \sum_{k=1}^{N} n_k \tilde{S}_k
\]

\[
    X = \sum_{k=1}^{N} n_k,
\]

where \(S_0\) is the stock price at time zero (beginning of trading), \(\tilde{S}_k\) is the realised trade price for the \(k\)-th order, while \(n_k\) is the number of shares to be traded within interval \(k\). The initial value of the whole order is \(XS_0\), the captured value is \(\sum_{k=1}^{N} n_k \tilde{S}_k\), whereas the execution shortfall is equal to the difference between these two values. Next, a price process for the stock is defined as a

\textsuperscript{2}VWAP stands for a volume weighted average price.
\textsuperscript{3}In this framework the problem for a buy order is symmetrical, i.e. instead of being subtracted from, transaction costs are added to the initial value of the order.
\textsuperscript{4}The details of order implementation within a single interval are not considered here.
discrete time arithmetic random walk. This is an acceptable approximation to the more realistic geometric random walk over the intraday trading horizon that we assumed\(^5\). The price process is given by,

\[ S_k = S_{k-1} + \sigma \tau^{1/2} \xi_k, \]

(3.2)

where \( S_k \) is the stock price at time \( k \) (after the \( k \)-th interval), \( \sigma \) is the price volatility, and \( \xi_k \) is a set of IID zero mean unit variance normal random variables. The realised trade price \( \tilde{S}_k \) in equation 3.1 can now be defined as a function of the stock price \( S_{k-1} \) at the beginning of the interval \( k \) and two other functions, temporary price impact \( h(v) \) and temporary price impact risk \( f(v) \). Those last two functions depend on the speed of trading \( v_k \) during the interval \( k \), where \( v_k = \frac{n_k}{\tau} \). The trade price \( \tilde{S}_k \) is then specified as,

\[ \tilde{S}_k = S_{k-1} - h(v_k) + \tau^{-1/2} f(v_k) \tilde{\xi}_k, \]

(3.3)

where \( \tilde{\xi}_k \) are zero mean, unit variance normal random variables independent of the \( \xi_k \). We note that in contrast to Ref. [16] we assume that permanent price impact is zero, which means that a given trade price does not depend on previous trades. This is a valid assumption as long as the trades are small and do not perturb the prevailing stock price. Alternatively, if permanent price impact is linear in the speed of trading, which is a reasonable assumption, then it has no influence on the optimal order submission plan \( n_1, \ldots, n_N \) [16]. After substituting the right side of equation 3.3 for \( \tilde{S}_k \) in the trade shortfall equation 3.1 the expected value and variance of the shortfall can be determined with respect to the random innovations of the price process \( \xi \) as,

\[ E(x) = \tau \sum_{k=1}^{N} v_k h(v_k) \]

(3.4)

\(^5\)The difference between the two random walks over the time horizon considered is small enough to be neglected [17].
\[ V(x) = \sigma^2 \tau \sum_{k=1}^{N} x_k^2 + \tau \sum_{k=1}^{N} v_k^2 f(v_k)^2 \] (3.5)

where \( x_k \) stands for the number of shares held after the \( k \)-th interval, and \( x_k = x_{k-1} - n_k \). Detailed derivations of formulae 3.4 and 3.5 are provided in appendices D and E, respectively. To perform a mean/variance optimisation a utility function \( U(x) \) is constructed in the following way,

\[ U(x) = E(x) + \lambda V(x) \] (3.6)

where \( \lambda \) is proportional to the trader’s risk aversion, or inversely proportional to the trader’s aggressiveness. For \( h(v) \) of the form \( h(v) = \eta v^i \), where \( i \) is an integer, \( U(x) \) can be minimised analytically ([16]). The above is a summary of the order submission optimisation framework presented by Almgren in Ref. [16]. In this chapter we make two practical extensions to the Almgren’s model. Firstly, we observe that each trade, no matter how small, incurs a price impact\(^6\) of at least half of the bid-ask spread. Furthermore, if a volume requested by a trade does not exceed some critical volume \( h_{v0} \) available at the best quoted price, then the price impact will stay fixed at the value of the half-spread (see also Figure 3.1). To account for these we define \( h(v) \) as,

\[ h(v) = \begin{cases} \eta (v - h_{v0}) + \epsilon & v > h_{v0} \\ \epsilon & v \leq h_{v0} \end{cases} \] (3.7)

The same reasoning can be applied to the price impact risk by introducing the bid-ask spread and some critical volume \( f_{v0} \) into \( f(v) \). The price impact risk thus assumes the following form,

\[ f(v) = \begin{cases} \beta (v - f_{v0}) + \alpha & v > f_{v0} \\ \alpha & v \leq f_{v0} \end{cases} \] (3.8)

---

\(^6\)Price impact per share traded.
Our second extension to the Almgren’s model allows non-stationarity in price volatility and price impact. Specifically, parameters $\eta$ and $\sigma$ become functions of time, $\eta(k)$ and $\sigma(k)$, respectively. The time is measured by the interval index $k$.

### 3.2 An analytical solution

The first of our extensions introduces non-linearity into $h(v)$, and consequently the task of $U(x)$ minimisation becomes analytically intractable, requiring instead a numerical solution. However, an analytical solution can be found when $h(v)$ has a linear form $h(v) = \eta(k)v$ and $f(v)$ is constant with respect to volume traded, $f(v) = \alpha$. By writing $v_k = x_{k-1} - x_k$ and differentiating $U(x)$ with respect to each $x_k$ we obtain,

$$
\frac{\partial U(x_k)}{\partial x_k} = 2\{\lambda \sigma_k^2 x_k - A_k(x_{k-1} - x_k) + A_{k+1}(x_k - x_{k+1})\} \tag{3.9}
$$

where $k$ is the time interval index, while $A_k$ is given by,

$$
A_k = \eta_k + \lambda \sigma_k^2 \tag{3.10}
$$

The set of equations in $x_k$ are second order difference equations. To solve them we impose two boundary conditions, the initial number of shares $x_0 = X$, and the final number of shares held $x_N = 0$. Given these constraints we derive the following formula for $x_k$,

$$
x_k = \frac{(\prod_{i=1}^{k} A_i) D_{k+1} X}{C_k D_{k+1} - A_{k+1}^2 C_{k-1} D_{k+2}} \tag{3.11}
$$

where $C_k$ and $D_k$ are given by,

$$
C_k = B_k C_{k-1} - A_k^2 C_{k-2} \tag{3.12}
$$

$$
C_0 = 1, \quad C_{-1} = 0
$$

$$
D_k = B_k D_{k+1} - A_{k+1}^2 D_{k+2} \tag{3.13}
$$

$$
D_N = 1, \quad D_{N+1} = 0
$$

$$
B_k = \lambda \sigma_k^2 + A_k + A_{k+1} \tag{3.14}
$$
Detailed derivations of the formulae presented in this section are provided in appendix F. The equation 3.11 is an analytic solution of the optimal order submission problem with the objective function given by 3.6. Our solution does not take into account non-linearities in the price impact functions, but it allows the non-stationarities in the parameters, and should therefore be useful for rapidly calculating approximate strategies for a large number of stocks.

3.3 Dataset - 3 ASX stocks, 78 days

Our dataset covers a period of 78 trading days on the Australian Stock Exchange, between the 11th February, 2002, and the 31st May, 2002. It contains information on all trades, orders, as well as the contents of the limit order book collected during the trading hours, between 10 am and 4 pm, for three large stocks with the codes BHP, NAB, and TLS. To calculate total trading volume, daily and intradaily, we consider buyer-initiated\(^7\) and seller-initiated trades only.

3.4 A dynamic programming approach

We use the first 15 days in the dataset as a training period to estimate the intraday values of price volatility and parameters in functions \(h(v)\) and \(f(v)\). Subsequently we employ deterministic discrete time dynamic programming to generate order submission plans that minimise the objective function 3.6, for the estimated parameter values and two levels of the trader’s aggressiveness. An overview of discrete dynamic programming is provided in appendix A. To compare performance, we also generate submission plans for three other strategies, referred to as one-interval, uniform and VWAP. Subsequently the five trading plans are evaluated by a trading simulator, out-of-sample, on the test period

\(^7\)An overview of trade initiator types is presented in section 2.3.
Chapter 3. Optimisation of order submission

with the 63 remaining days in the dataset. The evaluation results are then com-
pared using two benchmarks, the measured execution shortfall and distance from
market VWAP. We also show the connection between the mean/variance optimi-
misation and the Value at Risk (VaR) approach. All of the above procedures are
carried out separately for each of the three stocks.

The first 15 days, or three weeks, in the dataset are used to estimate intraday
values of parameters employed in the optimisation. The ASX is open between
10 am and 4 pm, so we set the interval length to 30 minutes, which gives us 12
intraday intervals. For each combination of a parameter and a time interval we
calculate 15 values, one for each of the 15 trading days. A parameter average
across the 15 values is then derived. We consider all days as identical, ignoring
possible intra-week seasonalities. To determine single interval parameter values
we collect the following data:

- Total daily trading volume for each of the 15 days.
- The difference in midpoint price between the end and start of each half
  hour interval.
- The average, normalised, time weighted cumulative order volume available
  at each price tick\(^8\) in the order book relative to the midpoint price, for
  each half hour interval. Normalisation is with respect to the daily traded
  volume.

Figure 3.1 shows an example of the transformed normalised cumulative order
volume as a function of price per share, measured in ticks from the midpoint price,
for a particular half hour interval. This is a volume normalised, price invariant
measure of (temporary) price impact as a function of traded volume. Figure 3.2
shows the average intraday price impact parameter \(\eta(k)\), price volatility \(\sigma(k)\),
and price impact risk parameter \(\alpha(k)\). Note that \(\eta\) is derived from the inverse of
a slope for the corresponding data, such as the one illustrated in Figure 3.1.
Chapter 3. Optimisation of order submission

Figure 3.1: Transformed normalised cumulative order volume for the NAB stock averaged over a three week period.\textsuperscript{1,2}

\textsuperscript{1}This plot is derived from the order book by taking the time weighted average of the cumulative order volume, normalising it by the daily traded volume (y-axis), and then calculating the price impact per share (x-axis), in price ticks away from the midpoint price, for all observed cumulative order volumes measured as a percentage of the total daily volume. The origin of the x-axis is referenced to the midpoint price.

\textsuperscript{2}The vertical order of the line markers in the legend corresponds to the vertical order of the plotted curves. The minimal volumes are zero and are not visible.

In order to generate optimal order submission plans for non-linear and non-stationary forms of functions $h(v)$ and $f(v)$ we use the intraday parameters estimated above and employ deterministic discrete time dynamic programming with the following reward function and two dimensional state space model. The definitions are given for a sell plan, but can easily be converted into a buy plan.

The state variable is denoted by $S = (s_1, s_2)$, where $s_1$ represents the number of shares still held, and $s_2$ measures time intervals (half hour lengths). Let $a_k$ denote the action variable, specifying the number of shares to be sold at each time step with the constraint $0 \leq a_k \leq s_1(k)$, implying that we never buy shares and we can never sell more shares than we currently hold. Let $R(S, a)$ denote the reward function which is a function of the current state $S$ and the action taken $a_k$ ($a_k = n_k$). Then,

\textsuperscript{8}Price tick is a minimum difference between two price levels in the order book.
Figure 3.2: Intraday impact parameter $\eta$, price volatility $\sigma$, and price impact risk parameter $\alpha$ for the NAB stock averaged over a three week period.

$$R(S, a) = E(S, a) + \lambda V(S, a)$$  \hspace{1cm} (3.15)

where $E()$ and $V()$ have the general form as in equations (3.4) and (3.5), respectively, and,

\begin{align*}
    s_1(k) &= s_1(k-1) - a(k) \\
    s_2(k) &= s_2(k-1) + 1 \hspace{1cm} (3.16)
\end{align*}

For a sell program, the states are initialised as $s_1(1) = X$ and $s_2(1) = 1$. We would like to point out that the time interval $s_2(k)$ is modelled explicitly as part of the state space due to the reward function $R$ depending on the non-stationary parameters used in our model, such as price impact parameter $\eta(k)$ and price volatility $\sigma(k)$. Given the above state space model $S$ and the reward function $R$ the expected shortfall and variance of the shortfall can be rewritten as:

$$E(S, a) = \tau \sum_{k=1}^{N} a_k h(a_k, \eta_k, \epsilon_k, h_{k=0})$$  \hspace{1cm} (3.17)
\[ V(S, a) = \tau \sum_{k=1}^{N} \sigma_k^2(s_1(k) - a_k)^2 + \]
\[ \tau \sum_{k=1}^{N} a_k^2 f^2(a_k, \beta_k, \alpha_k, f_{kv0}) \]  

The optimal solutions are calculated for 12 intraday intervals, so that the whole stock holding is sold within a single day, and for two levels of the trader’s aggressiveness, as captured by the parameter \( \lambda \) in the reward function 3.15, with \( \lambda_1 = 10^{-6} \) and \( \lambda_2 = 10^{-1} \). In order to compare performance we have also designed three simpler order submission strategies. They are referred to as one-interval, uniform, and VWAP, and defined as:

- **One interval (ONEINT)**: all the stock is sold during the first interval.
- **Uniform (UNIFORM)**: the shares are sold at even speed throughout the day.
- **(VWAP) trader**: the percentage of shares sold during each interval, relative to our total volume, is equal to the corresponding percentage measured for the total daily market volume for a given stock and averaged over the first three weeks (training period) of our data.

Given a trading plan \( x \), we test its performance out-of-sample by simulating a trading process over the remaining 63 days in the dataset. Each of our trading plans has 12 trading targets, corresponding to 12 half hour intervals during the trading day. We simulate trading over a given half hour interval by trading every five minutes a fixed fraction of the trading target for that interval against the orders available in the limit order book. This approach, however, is not without difficulties. It is possible that sometimes we consume the same orders in the book a number of times. We also assume that the orders we consume will be replaced by new orders by the time the next interval starts, and that our trades
will only have a temporary price impact and no permanent price impact. The trading performance is measured by two benchmarks, execution shortfall and distance from market VWAP. The execution shortfall is defined in equation 3.1. The second benchmark, distance from market VWAP, where VWAP stands for volume weighted average price, is given by,

\[
\Delta VWAP = \left\{ \frac{VWAP_{trade}}{VWAP_{market}} - 1 \right\} \times 10^4
\] (3.19)

where \(VWAP_{trade} = \sum_k \hat{S}_k n_k / X\) and the units are basis points (\(\frac{1}{100}\)th of a percent). The volume weighted average price for the whole market, \(VWAP_{market}\), is derived in the same way as \(VWAP_{trade}\), but counts all trades executed on the market on a given day. Execution shortfall measures transaction costs of an individual trader regardless of the performance of the market. \(\Delta VWAP\), on the other hand measures the trading performance against the rest of the market. As far as the trade execution shortfall is concerned, we can also calculate a Value at Risk measure for a given trade execution strategy. We do this as follows. Firstly, we determine,

\[
\lambda_v = -\frac{\partial E}{\partial V^{1/2}}
\] (3.20)

for any trading strategy \(x\). We then calculate,

\[
p = \int_{-\infty}^{\lambda_v} N(z)dz
\] (3.21)

where \(N(z)\) is the standard normal density function and the Value at Risk is given by \(Var(p) = E(x) + \lambda_v V(x)^{1/2}\).

The software for the experiment was implemented using the SMARTS® trading and surveillance system, Matlab® with a dynamic programming toolbox [162], and gcc compiler on Unix.
3.5 Numerical solutions

The parameterisation of the reward function described in the previous section allowed us to generate optimal order submission plans. These plans were produced for two levels of the trader’s aggressiveness, equivalent to risk neutral (aggressive) and risk averse (non-aggressive). We also calculated submission plans for three other strategies, one-interval, uniform, and VWAP. Each of the five plans was generated for three different levels of order volume, set to 1\%, 5\%, and 10\%, respectively, of forecast total market volume on a single day. The above procedure was carried out separately for each of the three stocks. Three typical trading plans are shown in Figure 3.3. The first optimal plan, where $\lambda = 10^{-6}$, corresponds to the case where the trader is not worried about the risk of adverse price movements and about price impact risk. This is a risk neutral, or aggressive, strategy. The second optimal plan, where $\lambda = 0.1$, represents a very risk averse (non-aggressive) strategy. It corresponds to a shortfall Value at Risk $VaR(p) = 66c/share$, with $p = 0.99$, implying that this shortfall is predicted to be exceeded 1\% of the time. We note that for the second optimal plan the first two intervals trade 20\% of total volume, which is due to a rule introduced by us to limit trading in a single interval to a maximum of 2\% of forecast daily volume of the whole market. This is to ensure that our price impact is temporary. The VWAP plan resembles the letter “U”, which resembles the dynamics of intraday trading activity observed on the market.

The results of simulated trading are shown in Tables 3.1, 3.2 and 3.3, enclosed at the end of the chapter. We note that the variance of percentage daily volume traded is not taken into account in forecasting the variance of the shortfall. We will take this into account in later work. We test our strategies for both the buy and sell cases, so as to reveal any bias in our results due to the specific out-of-sample test period used. As far as one interval strategies are concerned, we note that each stock has low or zero values of $p$, with low or negative transaction
Chapter 3. Optimisation of order submission

Execution Strategies for NAB stock at 10% daily volume

<table>
<thead>
<tr>
<th>Interval No.</th>
<th>VWAP</th>
<th>Lambda1</th>
<th>Lambda2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.3: Two optimal trading plans and a VWAP plan for the NAB stock. The plans were computed at a level of 10% of daily volume. The trader’s aggressiveness for the optimal plans was set to $\lambda_1 = 10^{-6}$ and $\lambda_2 = 10^{-1}$, respectively.

This indicates that these strategies, despite being considered risk averse, have rather unpredictable transaction costs, due to large variations in price and liquidity in the book during the opening half hour. We also note that in a small number of instances our Value at Risk calculation appears spurious, which we believe is due to the non-differentiable nature of the price impact function given in (3.7). Such cases are indicated in the tables by a ‘-’. We can see that for each of the three stocks the UNIFORM, VWAP and $\lambda_1$ strategies produce similar shortfall forecasts and measured shortfalls, while the Value at Risk calculation shows that they are often close to risk neutral strategies ($p = 0.5$). The $\lambda_2$ strategies correspond to more risk averse strategies, as indicated by the Value at Risk forecasts and standard deviations of shortfall forecasts. The measured transaction costs do show reduced standard deviations, but not to the extent that our simple model predicts. Comparing the three stocks, we observe that TLS is very liquid, in the sense that at all trading levels, the forecast shortfalls are in most cases equal to 0.5 of a price tick (half a cent), indicating that there is large order volume at the best price quotes in the order book. We find that...
Figure 3.4: Intraday midpoint prices for the three stocks at 5 minute intervals, for 10 trading days of the out-of-sample test period.\footnote{Note the sudden price jumps, some of which are intraday, others overnight. In our approach we do not estimate overnight volatility.}

the resulting measured shortfalls are independent of the volume traded. This is probably caused by price volatility, which seems to dominate our results. It appears therefore, that to improve our order submission strategies we would need an improved price process model which exploits intraday serial correlation in prices. Figure 3.4 gives an indication of the level of volatility for the three stocks over 10 trading days of the test period.

3.6 Conclusions

This chapter has shown that a relatively simple model can produce fairly good forecasts of the variance of trading shortfalls for the static strategies tested by us. It also shows that if more information on future prices is not available, then standard deviations of trading shortfalls are many multiples of the expected shortfalls, up to moderate levels of trading (10% of daily volume traded on the whole market). The same optimisation model can be used to numerically compute optimal
### Chapter 3. Optimisation of order submission

**Table 3.1: Performance of trade execution strategies for the BHP stock.**

1. Results are provided for the two naive strategies ONEINT and UNIFORM, the VWAP heuristic, and the optimal approach with $\lambda = 10^{-6}$ and $\lambda = 10^{-1}$. Each of the strategies are tested at nominally 1%, 5% and 10% of average daily traded volume ($\hat{V}_D$). $V_D$ is actual average daily traded volume for the test period, measured as a percentage of absolute $\hat{V}_D$. VaR(p) is the shortfall upper bound that can be achieved with probability $p$. $sf$ and $sf$ are the forecast shortfall and measured shortfall in cents per share traded respectively. $\Delta \text{VWAP}$ indicates the fractional difference in volume weighted price achieved for the whole trade against the entire market for the day expressed in percentage basis points. Standard deviations over the 63 trading day test period appear in brackets.
Table 3.2: Performance of trade execution strategies for the NAB stock.\(^1\)

\(^1\)Results are provided for the two naïve strategies ONEINT and UNIFORM, the VWAP heuristic and the optimal approach with \(\lambda = 10^{-6}\) and \(\lambda = 10^{-1}\). Each of the strategies are tested at nominally 1\%, 5\% and 10\% of average daily traded volume (\(\hat{V}_D\)). \(\hat{V}_D\) is actual average daily traded volume for the test period, measured as a percentage of absolute \(\hat{V}_D\). \text{VaR}(p)\) is the shortfall upper bound that can be achieved with probability \(p\). \(\hat{s}_f\) and \(s_f\) are the forecast shortfall and measured shortfall in cents per share traded respectively. \(\Delta \text{VWAP}\) indicates the fractional difference in volume weighted price achieved for the whole trade against the entire market for the day expressed in percentage basis points. Standard deviations over the 63 trading day test period appear in brackets.
Chapter 3. Optimisation of order submission

<table>
<thead>
<tr>
<th>Trade Type</th>
<th>( V_D ) (Units)</th>
<th>( V_D ) (%)</th>
<th>( p )</th>
<th>VaR(( p )) (c/share)</th>
<th>( sf ) (c/share)</th>
<th>sf (c/share)</th>
<th>( \Delta VWAP ) (basis pts.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONEINT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buy</td>
<td>1</td>
<td>1.0</td>
<td>0.00</td>
<td>-4.0</td>
<td>0.7 (0.8)</td>
<td>-2.7 (12.2)</td>
<td>8.0 (91.2)</td>
</tr>
<tr>
<td>Sell</td>
<td>1</td>
<td>1.0</td>
<td>0.00</td>
<td>-3.9</td>
<td>0.7 (0.6)</td>
<td>3.1 (11.7)</td>
<td>-1.6 (72.3)</td>
</tr>
<tr>
<td>Buy</td>
<td>5</td>
<td>4.9</td>
<td>0.00</td>
<td>-3.4</td>
<td>3.2 (2.1)</td>
<td>-2.2 (12.0)</td>
<td>16.8 (81.9)</td>
</tr>
<tr>
<td>Sell</td>
<td>5</td>
<td>4.9</td>
<td>0.00</td>
<td>-3.2</td>
<td>3.1 (1.7)</td>
<td>3.5 (11.6)</td>
<td>-9.2 (70.7)</td>
</tr>
<tr>
<td>Buy</td>
<td>10</td>
<td>9.8</td>
<td>0.01</td>
<td>-3.2</td>
<td>6.4 (3.9)</td>
<td>-1.6 (11.9)</td>
<td>27.6 (76.5)</td>
</tr>
<tr>
<td>Sell</td>
<td>10</td>
<td>9.8</td>
<td>0.00</td>
<td>-3.1</td>
<td>6.1 (3.2)</td>
<td>4.0 (11.7)</td>
<td>-17.8 (73.2)</td>
</tr>
<tr>
<td>UNIFORM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buy</td>
<td>1</td>
<td>1.0</td>
<td>0.70</td>
<td>4.5</td>
<td>0.5 (7.7)</td>
<td>-2.5 (11.3)</td>
<td>10.4 (12.5)</td>
</tr>
<tr>
<td>Sell</td>
<td>1</td>
<td>1.0</td>
<td>0.70</td>
<td>4.5</td>
<td>0.5 (7.7)</td>
<td>3.4 (11.3)</td>
<td>-7.8 (12.4)</td>
</tr>
<tr>
<td>Buy</td>
<td>5</td>
<td>4.9</td>
<td>0.70</td>
<td>4.5</td>
<td>0.5 (7.7)</td>
<td>-2.5 (11.3)</td>
<td>10.6 (12.6)</td>
</tr>
<tr>
<td>Sell</td>
<td>5</td>
<td>4.9</td>
<td>0.70</td>
<td>4.5</td>
<td>0.5 (7.7)</td>
<td>3.4 (11.3)</td>
<td>-8.1 (12.4)</td>
</tr>
<tr>
<td>Buy</td>
<td>10</td>
<td>9.8</td>
<td>0.72</td>
<td>5.0</td>
<td>0.5 (7.7)</td>
<td>-2.5 (11.3)</td>
<td>10.8 (12.6)</td>
</tr>
<tr>
<td>Sell</td>
<td>10</td>
<td>9.8</td>
<td>0.72</td>
<td>5.0</td>
<td>0.5 (7.7)</td>
<td>3.4 (11.3)</td>
<td>-8.3 (12.3)</td>
</tr>
<tr>
<td>VWAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buy</td>
<td>1</td>
<td>1.0</td>
<td>0.69</td>
<td>4.3</td>
<td>0.5 (7.6)</td>
<td>-2.5 (11.4)</td>
<td>9.8 (11.8)</td>
</tr>
<tr>
<td>Sell</td>
<td>1</td>
<td>1.0</td>
<td>0.69</td>
<td>4.3</td>
<td>0.5 (7.6)</td>
<td>3.4 (11.4)</td>
<td>-8.2 (11.4)</td>
</tr>
<tr>
<td>Buy</td>
<td>5</td>
<td>4.9</td>
<td>0.69</td>
<td>4.3</td>
<td>0.5 (7.6)</td>
<td>-2.5 (11.4)</td>
<td>10.1 (11.8)</td>
</tr>
<tr>
<td>Sell</td>
<td>5</td>
<td>4.9</td>
<td>0.69</td>
<td>4.3</td>
<td>0.5 (7.6)</td>
<td>3.4 (11.4)</td>
<td>-8.5 (11.4)</td>
</tr>
<tr>
<td>Buy</td>
<td>10</td>
<td>9.8</td>
<td>0.71</td>
<td>4.8</td>
<td>0.5 (7.6)</td>
<td>-2.5 (11.4)</td>
<td>10.3 (11.8)</td>
</tr>
<tr>
<td>Sell</td>
<td>10</td>
<td>9.8</td>
<td>0.71</td>
<td>4.8</td>
<td>0.5 (7.6)</td>
<td>3.4 (11.4)</td>
<td>-8.8 (11.4)</td>
</tr>
<tr>
<td>( \lambda_1 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buy</td>
<td>1</td>
<td>1.0</td>
<td>0.70</td>
<td>4.7</td>
<td>0.5 (8.1)</td>
<td>-2.6 (11.9)</td>
<td>9.2 (75.1)</td>
</tr>
<tr>
<td>Sell</td>
<td>1</td>
<td>1.0</td>
<td>0.70</td>
<td>4.6</td>
<td>0.5 (8.1)</td>
<td>3.2 (11.6)</td>
<td>-2.9 (59.5)</td>
</tr>
<tr>
<td>Buy</td>
<td>5</td>
<td>4.9</td>
<td>0.70</td>
<td>4.7</td>
<td>0.5 (8.1)</td>
<td>-2.5 (11.6)</td>
<td>11.3 (39.3)</td>
</tr>
<tr>
<td>Sell</td>
<td>5</td>
<td>4.9</td>
<td>0.70</td>
<td>4.7</td>
<td>0.5 (8.1)</td>
<td>3.4 (11.5)</td>
<td>-7.9 (36.5)</td>
</tr>
<tr>
<td>Buy</td>
<td>10</td>
<td>9.8</td>
<td>0.70</td>
<td>4.7</td>
<td>0.5 (8.1)</td>
<td>-2.4 (11.4)</td>
<td>11.9 (26.6)</td>
</tr>
<tr>
<td>Sell</td>
<td>10</td>
<td>9.8</td>
<td>0.70</td>
<td>4.6</td>
<td>0.5 (8.1)</td>
<td>3.4 (11.4)</td>
<td>-8.0 (25.2)</td>
</tr>
<tr>
<td>( \lambda_2 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Buy</td>
<td>1</td>
<td>1.0</td>
<td>0.88</td>
<td>1.8</td>
<td>0.6 (1.0)</td>
<td>-2.7 (12.1)</td>
<td>8.3 (87.2)</td>
</tr>
<tr>
<td>Sell</td>
<td>1</td>
<td>1.0</td>
<td>0.85</td>
<td>1.6</td>
<td>0.7 (0.9)</td>
<td>3.2 (11.7)</td>
<td>-1.7 (70.3)</td>
</tr>
<tr>
<td>Buy</td>
<td>5</td>
<td>4.9</td>
<td>0.80</td>
<td>5.0</td>
<td>0.8 (5.0)</td>
<td>-2.5 (11.7)</td>
<td>11.2 (53.6)</td>
</tr>
<tr>
<td>Sell</td>
<td>5</td>
<td>4.9</td>
<td>0.79</td>
<td>4.9</td>
<td>0.8 (5.0)</td>
<td>3.4 (11.6)</td>
<td>-6.5 (46.9)</td>
</tr>
<tr>
<td>Buy</td>
<td>10</td>
<td>9.8</td>
<td>0.80</td>
<td>6.1</td>
<td>0.7 (6.6)</td>
<td>-2.4 (11.5)</td>
<td>11.6 (34.4)</td>
</tr>
<tr>
<td>Sell</td>
<td>10</td>
<td>9.8</td>
<td>0.79</td>
<td>6.1</td>
<td>0.7 (6.6)</td>
<td>3.4 (11.4)</td>
<td>-7.6 (31.5)</td>
</tr>
</tbody>
</table>

Table 3.3: Performance of trade execution strategies for the TLS stock.\(^1\)

\(^1\)Results are provided for the two naive strategies ONEINT and UNIFORM, the VWAP heuristic and the optimal approach with \( \lambda = 10^{-6} \) and \( \lambda = 10^{-1} \). Each of the strategies are tested at nominally 1%, 5% and 10% of average daily traded volume (\( V_D \)). \( V_D \) is actual average daily traded volume for the test period, measured as a percentage of absolute \( V_D \). VaR(\( p \)) is the shortfall upper bound that can be achieved with probability \( p \). \( sf \) and \( sf \) are the forecast shortfall and measured shortfall in cents per share traded respectively. \( \Delta VWAP \) indicates the fractional difference in volume weighted price achieved for the whole trade against the entire market for the day expressed in percentage basis points. Standard deviations over the 63 trading day test period appear in brackets.
strategies for non-stationary volatility and order book liquidity effects, and with non-linearities in the price impact function. An analytical optimal solution for the case with non-stationary price impact has also been derived. We note that high price volatility makes transaction costs difficult to forecast without some idea of future price direction. The models presented in this chapter are static models. Future investigations could consider state space models which also include state variables for price and liquidity in the order book. Investigation of how often the models should be recomputed is another important issue. Other extensions to the model could include improved forecasts of daily trading volume, improved price process models with non-gaussian innovations, price jumps and serial correlation in price returns, medium term and permanent trading impacts, and effects of undisclosed orders. The next chapter is devoted to an exploratory analysis of trade level data. The analysis uses histograms and unsupervised clustering by the self-organising map. The uncovered clusters are shown to correspond to the states of market non-equilibrium as manifested by the volume imbalance in the limit order book.
Chapter 4

Visualisation and clustering of trade data

The previous chapter presented an order submission optimisation framework. An analytical solution was derived for non-stationary price volatility and price impact functions. More general cases, with non-linear and non-stationary price impact functions, were solved numerically via deterministic discrete time dynamic programming. This chapter presents an exploratory analysis of single trade data for the ten stocks on the ASX with the largest market capitalisation. We employ the self-organising map (SOM) technique to perform unsupervised clustering of four dimensional data so as to enable their visualisation. The input space data include trade size, the best bid and ask volumes, and a variable capturing lagged trade signs. The visualisation of the SOM-processed data reveals two clusters dominated by buyer-initiated and seller-initiated traders, respectively. Furthermore, the uncovered clusters correspond to respective order (volume) imbalances in the limit order book.

Starting from section 4.1 we conduct an exploratory analysis of trade level data. Our research hypotheses are formulated in section 4.1. The trade dataset is described in section 4.2. Section 4.3 gives some details on the self-organising map algorithm applied to the trade clustering task. More information on the SOM
technique can be found in the background chapter, in section 2.6.2. Results and their discussion are presented in section 4.4, followed by the conclusions in section 4.5.

4.1 Hypotheses

In this chapter we provide evidence of various order submission strategies by analysing a historical record of executed trades. We study individual trades and the states of the limit order book immediately preceding them. Our qualitative approach employs histograms and unsupervised clustering as the main techniques. We have divided the data exploration process into four tasks by formulating the following four hypotheses (labels in brackets):

1. \(H_1\) Market orders do not request more volume than there is available at the best relevant price in the order book.

2. \(H_2\) Seller-initiated trades are more frequent than buyer-initiated trades for high values of an order (volume) imbalance in the order book.

3. \(H_3\) Seller-initiated trades are more frequent than buyer-initiated trades when a recent market pressure was a selling pressure.

4. \(H_4\) The above relationships have stock-specific and time-varying characteristics.

The order (volume) imbalance in the limit order book is defined as a ratio of the total volume at the best ask price \(askvol\) and the total volume at the best bid price \(bidvol\). We normalise these volumes as well as the volume ratio via a natural logarithm transformation to reduce the skewness of the data. More formally, the order (volume) imbalance \(imb\) is defined as \(imb = \ln(askvol/bidvol)\). According to this definition, the order imbalance is positive if \(askvol > bidvol\). The market pressure, on the other hand, captures a recent history of the trade sign.
If more than half of a selected number of recent trades have been seller(buyer)-initiated then the market pressure is considered to be a selling (buying) pressure.

As far as single trade data are concerned to date, researchers have focused mainly on the clustering of trade prices, where some prices occur more frequently than others within a given price range. A number of papers, for example Ref. [100], have studied the US markets. Ref. [5] analyses price clustering on the ASX, and reports an increased clustering effect for larger trades. Positive serial correlation in trade sign has been found by Ref. [29] for the Paris Bourse and by Ref. [104] on the NYSE. We are not aware of any previous research on cluster analysis of joint trade and limit order book data on a single trade level. Refs. [35–37] use supervised learning techniques and are based on the work described in chapters 5 and 6 of this thesis.

4.2 Dataset - 10 ASX stocks, 103 days

The dataset consists of 1,059,714 trades, covering a period of 103 trading days, from the 2nd January, 2002, to the 31st May, 2002. It includes information on all trades as well as bids and asks in the limit order book in a time range from 10:15am to 4pm for the ten stocks with the highest market capitalisation

\[1\]

during the investigated period, on the ASX. The complete record of all trades makes it the highest frequency dataset possible, in a temporal sense. For each trade there is a trade sign attribute, which can assume one out of four possible values: buyer-initiated, seller-initiated, exchange-initiated, and a crossing. We are interested in buyer-initiated and seller-initiated trades only. Before further analysis we perform an aggregation of trades triggered by the same order. A single market order may result in a sequence of trades (illustrated in Figure 2.1). To obtain a one-to-one mapping between market orders and trades we need to add up share volumes of all trades for each trade sequence, as described in

\[1\]The codes of the stocks, ordered by market capitalisation in a decreasing order, are as follows: NAB, BHP, CBA, TLS, WBC, NCP, ANZ, AMP, RIO, WOW.
the trade aggregation procedure in section 5.2. The aggregation of the original dataset reduces the number of trades from 1,059,714 to 767,612. We also remove all aggregated trades which took place during single price auctions. After pre-processing, our filtered dataset contains 689,076 trades, with 49.51% being buyer-initiated and 50.49% seller-initiated.

4.3 The self-organising map

We perform an exploratory, mainly qualitative analysis of the trade dataset. The key techniques used are histograms and unsupervised clustering, with results presented as two dimensional charts. We employ a self-organising map (SOM) as the unsupervised clustering procedure. A good description of the SOM algorithm can be found in Ref. [132], while its applications in exploratory data analysis are presented in Refs. [67, 121]. A brief overview of the SOM algorithm is also provided in appendix B. Specifically, we use SOM’s data quantisation for clustering of a four dimensional dataset (input space). The clustered data are then projected back onto two selected dimensions of the input space to enable their visualisation. The SOM algorithm was chosen because of its popularity, robustness, minimal assumptions, and performance in unsupervised density mapping of an input space distribution. The number of nodes for the SOM is selected to be proportional to the square root of the number of trades [212]. In our dataset of ten stocks, the minimum number of trades for the whole period for a single stock was 37,679, while the corresponding maximum number was 102,130. Consequently we set the number of nodes to 900, arranged in a square grid of 30x30, with rectangular connections between nodes. We use batch learning with a Gaussian neighbourhood function. Euclidean distance ($L_2$) serves as a measure of similarity, with scaling factors enabling significance tuning (additional normalisation) for individual variables. Other data projection and quantisation methods, such as Generative Topographic Mapping [32] or a combination of multidimensional
scaling and k-means, and other distance metrics, for example $L_1$ (Manhattan distance), could also be used, but are outside the scope of this experiment.

We chose the following four variables for the input space:

- $askvol$ - total volume at the best ask price in the limit order book just before a trade.
- $bidvol$ - total volume at the best bid price in the limit order book just before a trade.
- $size$ - size of a trade.
- $\epsilon_{cnt}$ - market pressure indicator.

The last variable, $\epsilon_{cnt}$, is calculated as a number of seller-initiated trades during the previous five trades. This is a discrete variable, with six possible values, from 0 to 5. Before employing the clustering procedure we perform data normalisation. The raw values of $askvol$, $bidvol$, and $size$ variables are transformed via the natural logarithm. This is done to correct for their heavily skewed distributions. To allow for data visualisation the output space has two variables, selected depending on the type of a chart and marked on the chart’s axes. There are two types of charts. The count charts show normalised trade counts, scaled in the range $[0, 1]$. The trade sign ratio charts present a ratio of seller-initiated and buyer-initiated trade counts, scaled in the range $[-3, 3]$. This last range has been empirically found to provide the best contrast for the visualisation of the trade sign ratio. The more seller-initiated trades there are, relative to the buyer-initiated ones, the higher the ratio and the closer to red the presentation colour is. The two types of charts use grey and colour scales to indicate counts and ratios, respectively. To show on the charts the results of the SOM clustering we use a two dimensional projection of the SOM codebook vectors back onto the input space. This approach has been chosen instead of presenting the two
Chapter 4. Visualisation and clustering of trade data

Figure 4.1: Normalised trade count (grey scale axis) for all stocks as a function of order book imbalance and trade size (60x70 bins).

dimensional SOM grid because the dimensions of the input space are easier to interpret.

The analysis is performed on a stock by stock basis. This approach has been dictated by the fact that even though there can be some correlations between price returns of different stocks, the stocks are traded through their own separate limit order books and they trade at different prices. The last point means that the same dollar value of shares could be represented by different volumes of shares for different stocks. As a result we would observe different trade sizes and different volumes in the order book. Figure 4.1 shows trade counts for all ten stocks in the dataset, with trade size and order imbalance on the axes. It can be seen that trade sizes form five major (marked with arrows) and several minor clusters (horizontal lines). When we compare this against Figure 4.2, which shows daily trade counts for the BHP stock, we can see that some of the BHP clusters are much more pronounced. Also, the trade counts for the clusters seem to vary on a daily basis. We find that the data, within the limits of our exploration, seem to
support hypothesis $H_4$. On the other hand, regardless of the trade count values, the existence of similar major clusters for all stocks and for the BHP stock in a size (ln) range between 6 and 9.3 seems to indicate a behavioural regularity, with corresponding trade sizes of 500, 1000, 2000, 5000, and 10,000 shares. As far as the time-varying aspects of the problem are concerned, to properly account for them we would need to apply a local modelling approach, which is beyond the scope of this experiment. We compute results for each of the ten stocks separately, for the whole period in the dataset. We find that, despite individual characteristics, the BHP stock is representative of the majority (but not all) of our stocks on a qualitative level. In the subsequent section therefore, unless indicated otherwise, we present results for the BHP stock only. Corresponding results for the other 9 stocks in the dataset are provided in appendix G.

The data processing and modelling software was implemented using the SMARTS® suite of applications, Matlab® computing platform, and the SOM toolbox developed at the Helsinki University of Technology, Finland.
4.4 Results

To address the first hypothesis ($H_1$) we calculate a percentage of trades, relative to the total number of trades, which do not trade more volume than available at the best relevant price in the limit order book. Table 4.1 presents results for each of the 10 stocks and across the whole dataset. As can be seen, trades which do not meet the condition represent less than 2\% of all trades for the whole dataset, and also less than 2\% for 7 individual stocks. We find that the data support hypothesis $H_1$. Next, we are going to analyse a number of charts produced to obtain a qualitative answer to the question posed by hypothesis $H_2$.

<table>
<thead>
<tr>
<th>Code</th>
<th>Si%</th>
<th>Bi%</th>
<th>Total%</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMP</td>
<td>60.16</td>
<td>38.05</td>
<td>98.21</td>
</tr>
<tr>
<td>ANZ</td>
<td>45.31</td>
<td>53.20</td>
<td>98.51</td>
</tr>
<tr>
<td>BHP</td>
<td>46.60</td>
<td>52.70</td>
<td>99.30</td>
</tr>
<tr>
<td>CBA</td>
<td>54.48</td>
<td>43.04</td>
<td>97.53</td>
</tr>
<tr>
<td>NAB</td>
<td>52.87</td>
<td>44.58</td>
<td>97.45</td>
</tr>
<tr>
<td>NCP</td>
<td>45.74</td>
<td>52.42</td>
<td>98.16</td>
</tr>
<tr>
<td>RIO</td>
<td>45.08</td>
<td>51.40</td>
<td>96.48</td>
</tr>
<tr>
<td>TLS</td>
<td>52.04</td>
<td>47.86</td>
<td>99.89</td>
</tr>
<tr>
<td>WBC</td>
<td>43.98</td>
<td>54.59</td>
<td>98.57</td>
</tr>
<tr>
<td>WOW</td>
<td>49.80</td>
<td>48.44</td>
<td>98.24</td>
</tr>
<tr>
<td>All</td>
<td>49.64</td>
<td>48.75</td>
<td>98.39</td>
</tr>
</tbody>
</table>

Table 4.1: Percentage of trades with the executed volume no greater than the volume at the best price in the limit order book (Si-Seller-initiated, Bi-Buyer-initiated).

Figure 4.3 depicts a trade count for the variables $askvol$ and $bidvol$. It seems to have a single core with a centre around the point (10, 10), and two weak (low count) projections to the sides. Looking at Figure 4.4 we can see that it is these two weak projections in Figure 4.3 which correspond to buyer-initiated (green to blue) and seller-initiated (green to red) trades, located symmetrically around the line for which $askvol = bidvol$. The seller-initiated trades, however, are above this line, while the buyer-initiated ones are below it. The separating line has a green colour, which stands for a (ln) ratio of 0. It appears to represent...
Chapter 4. Visualisation and clustering of trade data

Figure 4.3: Normalised trade count (grey scale axis) for the BHP stock as a function of \( \ln(bidvol) \) and \( \ln(askvol) \) (70x70 bins).

Figure 4.3 shows an equilibrium, where equal volumes at the best ask and the best bid coincide with equal trade counts for each sign, meaning equal probabilities of observing seller-initiated and buyer-initiated trades. However, there is also an elongated cloud of data between points (10, 6) and (6, 10), which is perpendicular to the \( \ln(bidvol) = \ln(askvol) \) line. This formation is a mixture of buyer-initiated and seller-initiated trades and could possibly be separated by accounting for trade size.

To include in our analysis additional variables we perform unsupervised clustering by applying the SOM algorithm to the four dimensional input space, with \( \ln(bidvol), \ln(askvol), \text{size}, \) and \( \epsilon_{\text{cnt}} \) as input variables. The codebook of the SOM is subsequently projected back onto a plane in the input space, by selecting only two of input dimensions. The results are presented in Figures 4.5 and 4.6. The trade count chart in Figure 4.5 has the appearance of a skeleton version of the chart in Figure 4.3\(^2\). The area with the highest values, the core, is very small, while the sideways projections manage to capture the direction and length of their

\(^2\)In Figures 4.5 and 4.6 the ranges of \( \ln(bidvol) \) and \( \ln(askvol) \) are smaller than the corresponding ranges in Figures 4.3 and 4.4 in order to magnify shrunken (quantised) features.
Figure 4.4: Trade sign ratio (colour scale axis) for the BHP stock as a function of askvol and bidvol (70x70 bins).

counterparts in Figure 4.3. We have experimented with various scaling factors for the input variables to assess how a change in the distance measure affects size and the separation of clusters on the chart. We have obtained the best results with scaling factors equal to one, that is, no scaling. The trade sign ratio chart in Figure 4.6 has one sideways projection assigned to the buyer-initiated trades (green to blue), and the other one to the seller-initiated trades (green to red). We note that there are a few blue points in the red area, but in general the separation between the buyer-initiated and seller-initiated trades seems to be very good. Statistics for the other 9 stocks in the dataset, provided in appendix G, typically show two large clusters, corresponding to buyer-initiated and seller-initiated trades, respectively. Furthermore, on the outside edges of the large clusters there are often two small clusters, one with the seller-initiated trades and the other with the buyer-initiated trades. We conclude that the foregoing provides support for hypothesis $H_2$.

---

3In terms of visual separation of clusters with buyer-initiated and seller-initiated trades.

4In terms of trade sign ratio.
Figure 4.5: Normalised trade count (grey scale axis) for the BHP stock after the SOM transformation as a function of \( \text{askvol} \), \( \text{bidvol} \), \( \text{size} \), and \( \epsilon_{cnt} \) (70x70 bins).

The third hypothesis, \( H_3 \), can be addressed through an appropriate frequency table. Table 4.2 shows respective percentages of seller-initiated and buyer-initiated trades, relative to the total number of trades, associated with the six possible values of the \( \epsilon_{cnt} \) variable. The extreme values of 0 and 5 correspond to an extreme imbalance between the numbers of seller-initiated and buyer-initiated trades, with a strong positive autocorrelation between the majority sign of the previous five trades and the sign of the next trade. The Ln ratio in the table

<table>
<thead>
<tr>
<th>( \epsilon_{cnt} )</th>
<th>Si%</th>
<th>Bi%</th>
<th>( \text{Ln(}\frac{\text{Si}%}{\text{Bi}%}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.51</td>
<td>5.12</td>
<td>-0.713</td>
</tr>
<tr>
<td>1</td>
<td>7.76</td>
<td>11.60</td>
<td>-0.402</td>
</tr>
<tr>
<td>2</td>
<td>12.51</td>
<td>14.91</td>
<td>-0.176</td>
</tr>
<tr>
<td>3</td>
<td>13.12</td>
<td>12.86</td>
<td>0.020</td>
</tr>
<tr>
<td>4</td>
<td>8.33</td>
<td>6.75</td>
<td>0.211</td>
</tr>
<tr>
<td>5</td>
<td>2.77</td>
<td>1.78</td>
<td>0.440</td>
</tr>
</tbody>
</table>

Table 4.2: Percentage of the seller-initiated and buyer-initiated trades for the BHP stock after observing \( \epsilon_{cnt} \) seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).
Figure 4.6: Trade sign ratio (colour scale axis) for the BHP stock after the SOM transformation as a function of \( \log(\text{askvol}) \), \( \log(\text{bidvol}) \), \( \log(\text{size}) \), and \( \epsilon_{\text{cnt}} \) (70x70 bins).

has been calculated as a natural logarithm of the ratio of the percentage of the seller-initiated and buyer-initiated trades. Although the corresponding data for the other stocks vary\(^5\), they also show a positive autocorrelation in trade sign. We conclude that the data support hypothesis \( H_3 \).

4.5 Conclusions

In this chapter we conducted an exploratory analysis of the trade level data for the ten stocks on the Australian Stock Exchange. We formulated and then qualitatively confirmed four hypotheses. The novel contribution of this chapter is the application of the self-organising map to the unsupervised clustering of trades, based on limit order book data and other parameters. The visualisation of the transformed data has revealed that buyer-initiated and seller-initiated trades form two distinct clusters. Furthermore, the uncovered clusters correspond to

\(^5\)See appendix G.
respective order (volume) imbalances\textsuperscript{6} in the limit order book, i.e. states of market non-equilibrium. In the next chapter (5) we use a supervised learning technique to develop a non-parametric trade sign inference model. The set of predictor variables is enlarged by adding more contemporaneous and lagged order book volume and trade size information. Various subsets of the enlarged variable set are tested to determine the one which maximises trade sign classification accuracy.

\textsuperscript{6}Defined on page 72.
Chapter 5

Trade sign - non-parametric model

The previous chapter was devoted to an exploratory analysis of trade level data. The analysis used histograms and unsupervised clustering by the self-organising map. Two distinct clusters, with buyer-initiated and seller-initiated trades, respectively, were uncovered. The uncovered clusters correspond to the states of market non-equilibrium as manifested by the volume imbalance in the limit order book. In this chapter we investigate a regularity in market order submission strategies for twelve stocks with large market capitalisation on the Australian Stock Exchange. The regularity is evidenced by a predictable relationship between trade sign, size of the trade, and the contents of the limit order book before the trade. We demonstrate this predictability by employing supervised learning to develop an empirical inference model that classifies trades into buyer-initiated and seller-initiated. The model is based on a local non-parametric method, k-nearest-neighbour, which in the past was used successfully for chaotic time series prediction [85, 118, 140]. The k-nearest-neighbour with three predictor variables achieves an average out-of-sample classification accuracy of 71.40% (SD=4.01%), compared to 63.32% (SD=4.27%) for the linear logistic regression with seven predictor variables. The result suggests that a non-linear approach may produce a
more parsimonious trade sign inference model with a higher out-of-sample classification accuracy. Furthermore, for most of our stocks the observed regularity in market order submissions seems to be stationary up to 30 trading days.

To date the trade sign variable has been employed in such areas of market microstructure research as price formation, order and trade imbalance, order flow and order submission strategies, price impact, and trade classification. Our study primarily belongs to the research on order submission strategies and trade classification. Refs. [3, 4, 29, 96, 101, 105, 180, 185, 219] represent a small sample of studies on order flow and order submission strategies in various markets with a limit order book. Ref. [29] finds a positive autocorrelation in the order flow in the Paris Bourse, where a probability of a buy (sell) market order conditional on the previous buy (sell) market order is greater than an unconditional probability of such an order. Ref. [96] reports the same regularity for the Tokyo Stock Exchange. The observed positive autocorrelation in the order flow is considered to be caused by the breaking up of large orders, momentum trading, and similar reactions to news releases [29]. The autocorrelation of order signs is also claimed to be a long-memory process, by Ref. [39] for the Paris Bourse, and by Ref. [147] for the London Stock Exchange. As far as order submission strategies are concerned, traders are found to monitor the state of the order book and choose their actions accordingly [3, 4, 180, 185, 210]. In particular, there is some evidence that a large volume on the same side of the book makes submissions of market orders more frequent [185]. On the other hand, some authors speculate that a large volume on the opposite side may encourage the submission of a large market order [180].

There already exist a number of studies on trade classification [8, 76, 145, 169]. The proposed methods, however, have been primarily designed for quote-driven markets. They aim to recover trade sign with as high an accuracy as possible, relying on quote and trade prices. Perhaps the closest to our work are studies by Porter [179] and Aitken et al. [6]. Porter [179] uses logistic regression to
classify trades and finds systematic temporal patterns in interday and intraday probabilities of trading at the asking price on the US and Canadian exchanges. Aitken et al. [6] analyse the intraday probability of trading at the asking price on the Australian Stock Exchange. They use limit order book and other data to build a logistic regression model for a set of over 3 million trades, and manage to correctly classify 53.3% of trades, while 51.58% of all trades in their dataset are at the asking price.

This chapter explores a regularity in market order submission strategies on the Australian Stock Exchange (ASX). The regularity is evidenced by a predictable relationship between trade sign, size of the trade, and the contents of the limit order book before the trade. We demonstrate this predictability by employing supervised learning to develop an empirical trade sign inference model that classifies trades into buyer-initiated and seller-initiated. The model is based on a local non-parametric method, k-nearest-neighbour (k-NN). This method has been successfully applied by other researchers to forecasting chaotic time series [85, 118, 140], as well as various financial time series, for example currency exchange rates [90] and a stock index [218]. We use transaction level data for twelve large stocks on the ASX. The trade sign classification is conditional on contemporaneous and past volumes in the order book, trade sizes, and past trade signs. Quote and trade prices are not used. Classification accuracy is determined through out-of-sample testing. The classification performance of the k-NN classifier is compared against the performance of three other classifiers: linear logistic regression, trade continuation, and majority vote. We show that the k-NN classifier is superior to the other classifiers and can separate buyer-initiated and seller-initiated trades in our dataset with an average accuracy of over 71%. Furthermore, for most of our stocks, the observed regularity in market order submissions seems to be stationary up to 30 trading days.
Chapter 5. Trade sign - non-parametric model

5.1 Dataset - 12 ASX stocks, 199 days

Our dataset consists of single transaction information on all trades and orders, and full limit order book contents for twelve stocks with large market capitalisation\(^1\) on the ASX, for the period from 11 November 2002 to 27 August 2003, comprising 199 trading days. During the investigated period the selected stocks belonged to the top 30 stocks ranked by market capitalisation (8 stocks were in the top 10), were actively traded on each day, and did not undergo any major price revisions or splits. A sample sequence of trade and order book data is provided in appendix H. We analyse buyer-initiated and seller-initiated trades only. Crossing trades and exchange-initiated trades are excluded\(^2\). As far as the limit order book is concerned, the dataset includes complete price and size information for each bid and ask in the book throughout a trading day. There are 2,355,334 trades in the whole dataset. A subset with buyer-initiated and seller-initiated trades represents 92.73% of the dataset and contains 2,184,046 trades, with 50.44% of them being buyer-initiated.

5.2 The k-nearest-neighbour

We develop an empirical inference model of the trade sign variable for a single trade. The first 30 days in the dataset are used to select, through an exhaustive search, the best predictor variable sets out of a collection of 71 sets. Variable sets are ranked by classification accuracy across all stocks, and the best sets are selected for the logistic regression and the k-nearest-neighbour. The remaining 169 days in the dataset serve as a test set to evaluate the classification accuracy of the models with the best predictor variable sets. Two simple classifiers, a trade continuation and a majority vote, based on lagged values of the trade sign

\(^1\)The codes of the stocks, ordered by market capitalisation in a decreasing order, are as follows: NAB, BHP, CBA, ANZ, WBC, NCP, RIO, WOW, FGL, SUN, SGB, MIG.

\(^2\)An overview of trade initiator types is presented in section 2.3.
only, are used for performance comparison. The models are estimated and tested with a moving window method.

Before the analysis, we construct a market order sign proxy by aggregating together trades resulting from the same order. We apply two simple rules to aggregate trade sequences\(^3\). Firstly, a change of the spread in the limit order book from positive to non-positive signals a beginning of a new trade sequence. Secondly, the time when the spread becomes positive again marks the end of that trade sequence. The trade sequence found is then aggregated into a single trade, with its size being equal to the sum of all constituent trade sizes. The process is repeated for all trades in the dataset. This approach works even during periods of concentrated trading, where there are orders and trades with the same timestamp (accurate to 1 second) and the duration between transactions seems to be zero, because no new or amended orders are accepted until the market is cleared. The process of aggregation reduces the total number of buyer-initiated and seller-initiated trades in the dataset to 1,542,205, out of which 51.78% are buyer-initiated.

The complete history of trade and order flow could potentially allow an exhaustive search approach to find the best predictor variables for the specific target variable. By the exhaustive search we mean estimating models and testing their classification accuracy for all possible combinations of predictor variables. Unfortunately, the large amount of data and dimensions in our dataset would make this approach prohibitively expensive (in terms of computational time). On the other hand, variable selection methods, for example those based on the Akaike information criterion [12], are not applicable for the k-NN classifier. The k-nearest-neighbour belongs to a class of memory-based classifiers and requires out-of-sample testing to assess its generalisation performance. Our variable selection procedure is a constrained version of the exhaustive search. The number of possible predictor variable combinations, further referred to as variable sets,

\(^3\)An example of a trade sequence is illustrated in Figure 2.1.
is restricted to 71 by introducing a set of candidate variables and a set of rules for combining these variables into variable sets. The set of candidate predictor variables, \( \mathbf{V} \), consists of the following variables:

- \( a_{n-g}^p \) - lag \( g \) of the total volume in the limit order book at the ask price level \( p \), captured just before an order which triggered the \((n-g)\)-th trade; \( g \in \mathbb{Z}, g = 0 \ldots 3; p \in \mathbb{Z}, p = 1 \ldots 3; n \) indexes over the aggregated daily trades.

- \( b_{n-g}^p \) - lag \( g \) of the total volume in the limit order book at the bid price level \( p \), captured just before an order which triggered the \((n-g)\)-th trade; \( g \in \mathbb{Z}, g = 0 \ldots 3; p \in \mathbb{Z}, p = 1 \ldots 3. \)

- \( s_{n-g} \) - lag \( g \) of the trade size; \( g \in \mathbb{Z}, g = 0 \ldots 5. \)

The symbol \( \mathbb{Z} \) denotes the set of integers. The lagged trade sign is denoted as \( \epsilon_{n-g} \) and does not belong to \( \mathbf{V} \). The sign of the current trade, \( \epsilon_n \) (lag 0), is the target variable for the inference. Throughout the rest of the chapter the index \( n \) of the current trade will be omitted, simplifying variable symbols to \( a_g^p, b_g^p, s_g \), and \( \epsilon_g \), respectively. Consequently, \( \epsilon_0 \) will denote the target variable. The trade sign variable \( \epsilon_g \) can assume two values only, +1 for buyer-initiated trades, and -1 for seller-initiated trades. Total volumes and trade size are measured in units of shares. The first ask (bid) price level \( (p = 1) \) corresponds to the price of the best ask (bid) in the limit order book, while subsequent price levels correspond to prices \( p - 1 \) price ticks above (below) the first ask (bid) price. A price tick represents a minimum allowable distance between two price levels, as determined by the stock exchange. For all the stocks in the dataset one price tick is equal to one cent\textsuperscript{4}. The largest value of \( g \) was set to 5 (for the trade size variable). Consequently, the first five trades on each day are used only as lagged trades.

To further reduce the number of predictor variables and their combinations, an

\textsuperscript{4}Prices on the ASX are quoted in Australian dollars and cents.
additional set of constraints, $C$, is imposed. It specifies rules which must be satisfied by any variable set $X$, where $X \subseteq V$:

1. Number of elements: $\#X = n_x$, and $n_x = 2 \ldots 7$.

2. Bid-Ask symmetry: if $a^p_g \in X$ then $b^p_g \in X$.

3. Mandatory variables: $\{a^1_0, b^1_0\} \subseteq X$ or $\{a^1_1, b^1_1\} \subseteq X$.

4. Price priority: if $x^p \in X$ then $\forall i \in Z$: $x^{p-i} \in X$, $p - i \geq 1$.

5. Lag priority: if $x^g \in X$ then $\forall i \in Z$: $x^{g-i} \in X$, $g - i \geq 1$.

The unary operator “#” determines the number of elements in a set. The maximum number of variables in a set is limited to seven due to our preference for parsimonious models and a need for a sufficient ratio of cases (trades) to predictor variables [206]. The introduction of the two sets, $V$ and $C$, reduces the total number of predictor variable sets to 71. Before model estimation, all predictor variables are pre-processed by calculating their natural logarithms. After this transformation, lagged values of the trade size $s_g$ are signed with the corresponding values of the lagged trade sign $\epsilon_g (g > 0)$. The contemporaneous value of the trade size $s_0$ is not signed because the contemporaneous trade sign $\epsilon_0$ is the target variable. The signing procedure incorporates trade sign into trade size. This avoids a potential problem of how to include binary variables in the distance metric of the k-NN classifier, if lagged trade signs were included as predictor variables.

An instance of a set of values for a given variable set, including the current trade sign, corresponds to a single trade and is called a data point. The terms data point and trade will be used synonymously, but for clarity one may sometimes be preferred over the other. The only target variable is the sign of the current trade, $\epsilon_0$. The process of model estimation (training) and testing is iterative, and employs a moving window method. The models are estimated
using a given training interval $\mathbf{T}$, which consists of all trades on $N_t$ days in $\mathbf{T}$. The models estimated on a given training interval are tested on a test interval $\mathbf{S}$ comprising all trades on a single day (test day) immediately after the training interval $\mathbf{T}$. Selection of more recent data for the test interval than for the training interval is dictated by the time series nature of the data [144]. The result of the testing procedure is a single classification accuracy value for the test day. The estimation and testing are then repeated for a new pair of training and test intervals, obtained by shifting the previous pair of intervals one day forward. The process continues iteratively for $N_s$ test days, producing a set of $N_s$ daily classification accuracy values for each model.

The models are built for the four classifiers: logistic regression, k-nearest-neighbour (k-NN), trade continuation, and majority vote. The logistic regression classifier is constructed as follows:

$$\begin{equation}
\epsilon_0 = \begin{cases} 
-1 & \text{if } \gamma \leq 0 \\
+1 & \text{if } \gamma > 0 
\end{cases}
\end{equation}$$

$$\gamma = \ln \left( \frac{P(\epsilon_0)}{1 - P(\epsilon_0)} \right)$$

$$\gamma = f(x), \quad f(x) = Ax + c, \quad x = (x_1, \ldots, x_{n_x})$$

Function $f(x)$ is a linear regression with $n_x$ predictor variables $x_i$, where $x_1 \in \mathbf{X}$. The value $\gamma$ of the logit function is calculated as a natural logarithm of the ratio of the estimated class membership probabilities $P(\epsilon_0)$ and $(1 - P(\epsilon_0))$ [206]. Input data $x$ are assigned to the buyer-initiated class ($\epsilon_0 = +1$) if their corresponding logit value $\gamma$ is above 0, and to the seller-initiated class ($\epsilon_0 = -1$) otherwise.

The k-nearest-neighbour classifier belongs to a class of non-parametric, memory-based classifiers. During the training phase, a set $\mathbf{D}_t$ of all data points in a given training interval $\mathbf{T}$ is stored in the classifier’s memory. Testing is conducted for a set $\mathbf{D}_s$ of all data points in a test interval $\mathbf{S}$. During the evaluation

---

The coefficients of $f(x)$ are estimated via the maximum likelihood method to find the best match between the estimated and observed class membership probabilities.
of a test data point \( \mathbf{d}_s \) the classifier computes squared Euclidean distances between \( \mathbf{d}_s \) and all the data points \( \mathbf{D}_t \) in its memory. Calculation of the Euclidean distance involves all \( n_x \) dimensions (predictor variables) in a given set \( \mathbf{X} \). Subsequently a set \( \mathbf{K} \) of \( k \) data points from the classifier’s memory with the shortest distances to \( \mathbf{d}_s \) is selected. The trade sign \( \epsilon_0 \) for the test data point \( \mathbf{d}_s \) is inferred to be equal to the trade sign of the majority of the data points in \( \mathbf{K} \), as long as \( k \) is an odd positive integer. In our experiment three values of \( k \) are used: 1, 5, and 9. For each value of \( k \), a separate k-NN model is estimated and tested. The following is a more formal description of the classifier’s operation:

\[
\epsilon_0 = \begin{cases} 
-1 & \text{if } \beta_\epsilon \leq 0 \\
+1 & \text{if } \beta_\epsilon > 0 
\end{cases} 
\] (5.2)

\[
\beta_\epsilon = \sum_{\mathbf{d}_i \in \mathbf{K}} \epsilon_0 \\
\forall (\mathbf{d}_i \in \mathbf{K}) \forall (\mathbf{d}_j \in \mathbf{D}_t') : \|\mathbf{x}_i - \mathbf{x}_s\| \leq \|\mathbf{x}_j - \mathbf{x}_s\|, \mathbf{d}_s \in \mathbf{D}_s
\]

\[
\mathbf{D}_t' = \mathbf{D}_t \setminus \mathbf{K}, \; \#\mathbf{K} = k, \; \mathbf{d} = (x_1, \ldots, x_{n_x}, \epsilon_0) \\
\|\mathbf{x} - \mathbf{x}'\| = \left(\sum_{j=1}^{n_x} (x_j - x_j')^2\right)^{\frac{1}{2}}
\]

The brackets “\(\|\cdot\|\)” denote an Euclidean distance operator, while the binary operator “\(\setminus\)” calculates a set difference. A brief overview of the k-nearest-neighbour classifier is presented in appendix C. More information on this technique is provided in section 2.6.3, while a comprehensive treatment can be found in Ref. [62].

The trade continuation classifier exploits the observed autocorrelation in market order sign. It assumes that the sign of the current trade will be the same as the sign of the previous trade:

\[
\epsilon_0 = \epsilon_1 
\] (5.3)

The majority vote classifier does not use any information from the test interval. It detects an imbalance between buyer-initiated and seller-initiated trades in the training interval. The classifier determines the sign of the majority and then
assigns it to all trades in the test interval:

$$
\epsilon_0 = \begin{cases}
-1 & \text{if } \beta_\epsilon \leq 0 \\
+1 & \text{if } \beta_\epsilon > 0
\end{cases}
$$

$$
\beta_\epsilon = \sum_{d_i \in D_t} \epsilon_0
$$

The last two classifiers, trade continuation and majority vote, do not use any predictor variables from the list $\mathbf{V}$, and consequently their performance does not depend on a choice of $\mathbf{X}$.

As mentioned earlier, the length of a training interval is $N_t$ days, while there are $N_s$ one day test intervals. The choice of $N_t$ is not obvious and may depend on a particular stock. We try various values between 1 and 30 days, starting from one day, and then every even number of days until 30 days. Each of the selected 16 values of $N_t$ defines a separate training timescale. One of the classifiers, the trade continuation, does not depend on a training timescale. We initially intended to estimate and test the four classifiers\(^6\) described above on the whole dataset of 199 trading days. The k-nearest-neighbour classifier has a high computational cost and was expected to consume most of the computer time. Preliminary computations revealed that building models for the four classifiers, 16 training timescales, and 71 variable sets, over the whole period in the dataset, would take several months on a four node computer cluster\(^7\). More importantly, however, it became apparent that the size of the dataset would be too small, relatively to the total number of 4,561 potential models for each stock\(^8\), to find statistically significant differences in the classifiers’ performance. These two factors, the computational cost and the statistical significance of multiple comparisons, led us to impose additional constraints on the experiment.

We divide the whole dataset of 199 trading days into two parts. The first 30

---

\(^6\)Separate k-NN models are built for each of the three values of $k$.

\(^7\)Each node is approximately twice as fast as the Intel® Celeron® 2.00 GHz processor with 256 MB of memory.

\(^8\)The total of 4,561 models is calculated in the following way: 3 * 16 * 71 k-NN, 16 * 71 logistic regression, 1 trade continuation, and 16 majority vote models.
days in the dataset are used to select a subset from the collection of 71 predictor variable sets. The subset will include four variable sets, the best two for the logistic regression, and the best two for the k-nearest-neighbour. Out of the two sets for each classifier one set will allow contemporaneous (current) variables, while the other one will not. The trade continuation and the majority vote classifiers are not affected by these decisions because they do not depend on any of the predictor variables on the list $V$. The training interval length is set to one value only, 20 days. This particular value has been selected because we have a preference for predictor variable sets performing well on longer timescales, between 10 and 30 days. There will be 10 validation days which will provide 10 classification accuracy values for each stock. To mitigate against a potential problem of overfitting\(^9\), due to the small number of validation days, results for all stocks are pooled together. The variable sets with the highest mean classification accuracy, across individual mean accuracies for the twelve stocks, are selected as the best predictor variable sets.

The remaining 169 days in the dataset, starting from day 31, serve as a test set to evaluate daily performance of the classifiers with the selected four predictor variable sets. A separate classification accuracy is determined for each of the 16 training timescales. During this phase the first 30 days in the dataset are re-used to construct training intervals for model estimation, but they are never used for testing of classification accuracy. The last day of a first training interval, regardless of its length, always falls on day 30 in the dataset. This ensures that models are estimated with the most recent data available and that they are all tested on the same set of 169 days.

The software for the experiment was implemented using the SMARTS\(^\text{®}\) trading and surveillance system, and Matlab\(^\text{®}\) with the NETLAB toolbox [164].

\(^9\)Overfitting can occur when multiple tests are performed on a relatively small dataset. There are a number of remedies for this problem, including cross-validation, bootstrap, or increasing the size of the dataset.
Table 5.1: Classification accuracy (%) for the best predictor variable sets across the twelve stocks - 10 validation days.\footnote{Abbreviated headings: CV - contemporaneous variables; SetNo - variable set number; SD - standard deviation.}

<table>
<thead>
<tr>
<th>CV</th>
<th>SetNo</th>
<th>Predictor variables</th>
<th>Mean</th>
<th>SD</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic regression:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>53</td>
<td>$a_0^1$, $a_1^1$, $b_0^1$, $b_1^1$, $s_1$, $s_2$, $s_3$</td>
<td>60.95</td>
<td>3.28</td>
<td>55.70</td>
<td>66.23</td>
</tr>
<tr>
<td>No</td>
<td>45</td>
<td>$a_1^1$, $b_1^1$, $s_1$, $s_2$, $s_3$, $s_4$</td>
<td>55.78</td>
<td>2.82</td>
<td>51.32</td>
<td>60.05</td>
</tr>
<tr>
<td>k-NN ($k = 1$):</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>3</td>
<td>$a_0^1$, $b_0^1$, $s_0$</td>
<td>69.55</td>
<td>2.89</td>
<td>64.97</td>
<td>73.49</td>
</tr>
<tr>
<td>No</td>
<td>6</td>
<td>$a_1^1$, $b_1^1$, $s_1$</td>
<td>54.78</td>
<td>2.43</td>
<td>50.83</td>
<td>58.68</td>
</tr>
<tr>
<td>k-NN ($k = 5$):</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>3</td>
<td>$a_0^1$, $b_0^1$, $s_0$</td>
<td>71.63</td>
<td>2.99</td>
<td>66.61</td>
<td>76.23</td>
</tr>
<tr>
<td>No</td>
<td>6</td>
<td>$a_1^1$, $b_1^1$, $s_1$</td>
<td>57.10</td>
<td>2.71</td>
<td>52.82</td>
<td>61.53</td>
</tr>
<tr>
<td>k-NN ($k = 9$):</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>3</td>
<td>$a_0^1$, $b_0^1$, $s_0$</td>
<td>71.75</td>
<td>3.00</td>
<td>66.99</td>
<td>76.38</td>
</tr>
<tr>
<td>No</td>
<td>6</td>
<td>$a_1^1$, $b_1^1$, $s_1$</td>
<td>58.20</td>
<td>2.79</td>
<td>53.90</td>
<td>62.61</td>
</tr>
</tbody>
</table>

5.3 Results

The test statistics for the best predictor variable sets within the collection of 71 sets, calculated using the first 30 days in the dataset, are presented in Table 5.1. The statistics represent mean values across individual statistics for the twelve stocks. The means of the four individual statistics are reported: mean, standard deviation, minimum, maximum. The individual statistics were calculated across the 10 validation days. The sets with the highest mean classification accuracy were selected as the best predictor variable sets. Statistical significance was not determined because it is not used by our selection procedure. This approach was adopted to avoid an inconclusive result, if differences between sets were not statistically significant. As can be seen for all classifiers, the sets with contemporaneous variables have higher mean accuracies than the sets without them. The difference is around 5% for the logistic regression, and between 13% and 15% for the k-nearest-neighbour. The most interesting, however, is the difference between the k-NN classifier and the logistic regression for the sets with contemporaneous variables. Depending on the value of $k$, the k-nearest-neighbour has the mean
accuracy approximately 9% to 11% higher than the logistic regression, while its standard deviation varies from 2.89% to 3.28%. Furthermore, the higher the value of $k$ the better the performance of the k-NN classifier\textsuperscript{10}, even though the improvement between $k = 5$ and $k = 9$ is minimal. The results for the sets without contemporaneous variables show the difference of only 2.42% between the best k-nearest-neighbour ($k = 9$) and the logistic regression. Another aspect worth pointing out is the number of predictor variables in the best variable sets. In the case of the k-NN classifier, the sets have only three variables, and they are identical for all three values of $k$ within a group with contemporaneous variables (set 3), and within a group without contemporaneous variables (set 6), respectively. On the other hand, in the case of the logistic regression, the best variable set with contemporaneous variables contains seven variables, the maximum number allowed.

The selected best predictor variable sets were subsequently used to determine the classifiers’ performance for the twelve stocks across the 169 test days. Separate classification accuracy statistics were calculated for each of the 16 training timescales. Figure 5.1 shows mean accuracy curves for the WBC stock with the variable sets including contemporaneous variables. Each curve represents a single classifier and depicts mean classification accuracies for all training timescales. The chart for the WBC stock is a typical one. Eight other stocks in the dataset have charts which qualitatively agree with it. Figure 5.2 shows mean accuracy curves of the k-NN ($k = 9$) classifier with contemporaneous variables, for the twelve stocks. The whole set of twelve charts as in Figure 5.1 and the chart in Figure 5.2 have a number of qualitative characteristics enumerated below, with numbers in brackets specifying how many stocks exhibit a given characteristic. The main characteristics are as follows:

1. Among the k-NN classifiers, the higher the value of $k$, the greater the mean

\textsuperscript{10}In general, the performance of k-nearest-neighbour does not always increase for higher values of $k$, and taking into account too many neighbours will often lead to underfitting and an accuracy decrease.
The mean accuracy of the k-NN classifier, where $k = 9$, is a monotonically increasing function of the training interval length. The rate of the increase, however, rapidly declines. Small, negligible fluctuations are sometimes present (10).

3. The mean accuracy of the k-NN classifier, where $k = 9$, is greater than the mean accuracy of the logistic regression classifier for all training timescales (8).

4. The mean accuracy of the k-NN classifier, where $k = 9$, is greater than the
mean accuracies of the trade continuation and the majority vote classifiers, for all training timescales (12).

The best test statistics for individual stocks, calculated across the 169 test days, are presented in Table 5.2. For each classifier, the best statistics were determined by finding a training timescale (best interval) corresponding to the highest mean accuracy. The best training interval length is reported for the k-nearest-neighbour and the logistic regression. Statistics for the best k-NN classifier are included for $k = 9$ only. The trade continuation and the majority vote classifiers are not sensitive to a choice of predictor variables and have their duplicate statistics indicated by hyphens. Paired one-tailed t-tests with the Bonferroni adjustment for multiple comparisons ([184]) were applied to establish the statistical significance of differences in the means for each individual stock. The

---

11A brief overview of the Bonferroni adjustment is presented in appendix I.
### Table 5.2: Classification accuracy (%) for individual stocks - 169 test days.\(^1\)

<table>
<thead>
<tr>
<th>Stock</th>
<th>CV</th>
<th>k-NN ((k = 9))</th>
<th>Logistic reg.</th>
<th>Trade cont.</th>
<th>Major. vote</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>SD</td>
<td>(N_t)</td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>NAB</td>
<td>Yes</td>
<td>74.87*</td>
<td>3.08</td>
<td>30</td>
<td>59.57</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>57.95</td>
<td>2.08</td>
<td>30</td>
<td>55.95</td>
</tr>
<tr>
<td>BHP</td>
<td>Yes</td>
<td>70.08</td>
<td>4.25</td>
<td>6</td>
<td>70.34</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>64.99</td>
<td>3.24</td>
<td>30</td>
<td>59.77</td>
</tr>
<tr>
<td>CBA</td>
<td>Yes</td>
<td>73.48*</td>
<td>2.25</td>
<td>28</td>
<td>58.74</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>56.70</td>
<td>2.27</td>
<td>30</td>
<td>55.28</td>
</tr>
<tr>
<td>ANZ</td>
<td>Yes</td>
<td>73.72*</td>
<td>3.27</td>
<td>30</td>
<td>62.14</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>59.01</td>
<td>2.72</td>
<td>30</td>
<td>56.12</td>
</tr>
<tr>
<td>WBC</td>
<td>Yes</td>
<td>73.41*</td>
<td>3.58</td>
<td>24</td>
<td>63.13</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>59.11</td>
<td>2.46</td>
<td>28</td>
<td>56.38</td>
</tr>
<tr>
<td>NCP</td>
<td>Yes</td>
<td>70.97*</td>
<td>3.66</td>
<td>30</td>
<td>67.95</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>63.34</td>
<td>3.03</td>
<td>30</td>
<td>58.82</td>
</tr>
<tr>
<td>RIO</td>
<td>Yes</td>
<td>76.34*</td>
<td>2.92</td>
<td>30</td>
<td>58.98</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>58.27</td>
<td>3.12</td>
<td>22</td>
<td>56.88</td>
</tr>
<tr>
<td>WOW</td>
<td>Yes</td>
<td>72.07*</td>
<td>4.15</td>
<td>30</td>
<td>62.93</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>58.72</td>
<td>3.30</td>
<td>30</td>
<td>57.64</td>
</tr>
<tr>
<td>FGL</td>
<td>Yes</td>
<td>62.69</td>
<td>6.08</td>
<td>30</td>
<td>66.72</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>61.15</td>
<td>5.12</td>
<td>30</td>
<td>62.30</td>
</tr>
<tr>
<td>SGB</td>
<td>Yes</td>
<td>72.58*</td>
<td>4.91</td>
<td>30</td>
<td>60.78</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>56.59</td>
<td>4.46</td>
<td>10</td>
<td>58.74</td>
</tr>
<tr>
<td>SUN</td>
<td>Yes</td>
<td>69.24*</td>
<td>3.87</td>
<td>30</td>
<td>60.24</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>56.51</td>
<td>3.49</td>
<td>30</td>
<td>57.53</td>
</tr>
<tr>
<td>MIG</td>
<td>Yes</td>
<td>67.39</td>
<td>6.08</td>
<td>30</td>
<td>68.30</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>62.06</td>
<td>5.52</td>
<td>28</td>
<td>61.49</td>
</tr>
<tr>
<td>Av.</td>
<td>Yes</td>
<td>71.40</td>
<td>4.01</td>
<td>27</td>
<td>63.32</td>
</tr>
<tr>
<td>value</td>
<td>No</td>
<td>59.53</td>
<td>3.40</td>
<td>27</td>
<td>58.08</td>
</tr>
</tbody>
</table>

* - significant at \(\alpha = 0.01\). Statistical significance was tested only for the k-NN (\(k = 9\)) with contemporaneous variables and the indicated \(N_t\).

The mean accuracy of the k-NN classifier with \(k = 9\), contemporaneous variables and a respective best training interval length, was compared against 144 mean accuracies for other combinations of a classifier, variable set, and training timescale.\(^1\)

For most of the stocks the mean accuracy of the specified k-nearest-neighbour model was found to be greater than the mean accuracies of all other models except for some k-NN classifiers with contemporaneous variables. This is true for 9 out of 12 stocks (marked with * in Table 5.2), at the significance level of...\(^1\)The total number of models constructed for each stock was 145: \(2 \times 3 \times 16\) k-NN, \(2 \times 16\) logistic regression, 1 trade continuation, and 16 majority vote models.
The lack of significance for comparisons between various k-NN classifiers with contemporaneous variables was most probably due to small differences in their classification accuracy, the large number of statistical tests performed, and the conservative assumptions of the Bonferroni adjustment.

The average best mean for the k-NN \((k = 9)\) classifier with contemporaneous variables and the best training timescale, across all stocks, is 71.40%. It is 8.08% higher than the corresponding average best mean for the logistic regression. It is also 14.08% and 17.87% higher than the corresponding average for the trade continuation and the majority vote classifiers, respectively. The average statistics for the sets without contemporaneous variables are much less impressive. In particular, the average best means for the k-nearest-neighbour \((k = 9)\) and the logistic regression are both below 60%, just above the average mean for the trade continuation classifier (57.32%). The statistical significance of the individual results for the k-nearest-neighbour without contemporaneous variables was not determined in order to limit the total number of multiple comparisons.

As far as the best training interval length for the k-nearest-neighbour \((k = 9)\) is concerned, the average length is 27 days for both types of variable sets, with a value of 30 days for most of the individual stocks. As mentioned above, in the case with contemporaneous variables these values are not statistically significant, while in the case without contemporaneous variables the significance tests were not performed. The classification accuracy for most of the stocks, however, appears to be a monotonically increasing function of the training interval length, as depicted in Figure 5.2. These two characteristics, 30 days being the best training interval length\(^{13}\) and the monotonic increase of the classification accuracy, suggest that for most of our stocks the observed regularity in trade sign may be stationary up to 30 trading days. This value corresponds to the longest training

\(^{13}\)Training intervals longer than 30 days were not used, but the slopes of the curves in Figure 5.2 seem to suggest that any further increase in the classification accuracy would most probably be marginal.
timescale used in our experiment. A dataset covering a longer period would be required to determine an upper bound on the stationarity’s duration.

## 5.4 Conclusions

In this chapter we investigated a regularity in market order submission strategies on the Australian Stock Exchange. An empirical model for trade sign inference was developed using supervised learning and transaction level data for twelve large stocks on the ASX. We proposed the k-nearest-neighbour classifier as an alternative to the linear logistic regression. The average classification accuracy of the k-NN \((k = 9)\) classifier, across all stocks and allowing contemporaneous predictor variables, has been found to be 71.40\% (SD=4.01\%), or 8.08\% higher than the corresponding accuracy of 63.32\% (SD=4.27\%) for the logistic regression. When compared with the trade continuation and the majority vote classifiers, the k-nearest-neighbour is 14.08\% and 17.87\% better, respectively. The results for individual stocks show that the proposed k-NN classifier is better than the other three classifiers for most of the stocks, at the significance level of 0.01. The best k-NN model requires only three predictor variables: total volumes at the best bid and ask in the order book just before a trade, and the contemporaneous trade size. In contrast, the best logistic regression requires seven variables, the maximum allowed. These results suggest that a non-linear approach may produce a more parsimonious trade sign inference model with a higher out-of-sample classification accuracy. Furthermore, for most of our stocks the classification accuracy of the k-nearest-neighbour \((k = 9)\) with contemporaneous predictor variables is a monotonically increasing function of the training interval length, with 30 days being the best interval. It appears that for these stocks the investigated regularity in market order submissions may be stationary up to 30 trading days.

As far as commercial applications are concerned, it is not clear at this stage if
the observed regularity in trade sign can be profitably exploited. Some answers could perhaps be obtained by incorporating our model into the existing models of limit order execution and trading costs. In the next chapter (6) we conduct a detailed analysis of the k-NN classifier’s performance for the best variable set found. This analysis is followed by the development of a parametric non-linear model for trade sign inference. Such a parametric model will provide a more quantitative insight into market order submission strategies employed by market participants.
Chapter 6

Trade sign - parametric model

The previous chapter (5) was devoted to the development of a non-parametric model for trade sign inference. We have shown that a k-nearest-neighbour classifier with just three predictor variables can determine the trade sign with an accuracy of over 71%, across the 12 stocks with large market capitalisation on the ASX. In this chapter, we construct a piecewise linear model by parameterising the non-parametric model proposed in chapter 5 ([35]). The space of the predictor variables is partitioned into six regions. Signs of individual trades within the regions are inferred according to simple and interpretable rules. Across the 12 stocks, the new model achieves an average out-of-sample classification accuracy of 74.38% (SD=4.25%), which is 2.98% above the corresponding accuracy for the k-nearest-neighbour classifier developed in chapter 5. Two of the model’s regions, together accounting for 16.79% of the total number of daily trades, each have an average classification accuracy exceeding 91.50%. The results indicate a strong dependence between the predictor variables and trade sign, and provide evidence for an endogenous component in the order flow. An interpretation of the trade sign classification accuracy within the model’s regions offers new insights into the relationship between two regularities, that is competition for order execution and transaction cost minimisation, observed in the markets with a limit order book.
Recent empirical studies have shown that the order flow depends on the state of the limit order book [14, 29, 45, 68, 84, 93, 171, 180, 185, 210]. Most of the observed dependence seems to be caused by two behavioural regularities in order submission strategies. The first regularity concerns competition for order execution. If one side of the book is dominant, where the dominant side is the one with more depth\(^1\), then there is an imbalance between supply and demand, and limit orders on the dominant side face a longer time to execution [14] and a higher risk of an adverse price movement leading to non-execution. Consequently, traders on the same side of the market as the dominant side of the book are more likely to submit market orders to achieve an immediate execution [45, 93, 171, 185, 210]. The behaviour of buyers and sellers, however, may not be perfectly symmetrical [105, 185]. The second regularity we refer to as transaction cost minimisation. It is observed that a majority of individual market orders consume only a part of, or the whole, volume available at the best price in the order book [14, 29, 34, 68, 84, 93, 180, 185, 210]. Traders try to minimise their transaction costs, and by following this regularity ensure that the price per share of their trades will differ from the pre-trade midpoint price by the value of the half-spread only. Apart from the two regularities, other empirical studies found an autocorrelation in the unconditional and conditional order flow, where similar events tend to follow one another [14, 29, 35, 39, 65, 68, 93, 96]. In particular, Refs. [39, 65, 96, 147] reported an autocorrelation in trade sign\(^2\), Ref. [39] presented evidence for the long memory in trade sign, while Ref. [147] found the long memory in the order flow. The works of Porter [179] and Aitken et al. [6], on the other hand, detected temporal patterns in the probability of

\(^1\)Depth is measured as the volume (total number of shares) on a given side of the limit order book, usually at a single price (best price) or at a number of prices closest to the midpoint price. The midpoint price is an average of the best bid price (best bid) and the best ask price (best ask).

\(^2\)Trade sign has also been referred to by various authors as trade initiator, trade indicator, trade direction, or buy/sell indicator. Similar synonyms exist for market order sign.
trading at the best ask, and represent the closest work prior to the study in chapter 5 ([35]), which is the starting point for the research in this chapter.

We develop a piecewise linear parameterisation of the trade sign inference model proposed in chapter 5 ([35]). We showed that a k-nearest-neighbour classifier can infer the trade sign with an average accuracy of over 71%, for a set of 12 stocks on the ASX. The classifier used three predictor variables, the volume at the best bid and at the best ask just before a trade, and trade size. Across the 12 stocks the highest classification accuracy was achieved for a training interval of 30 days. Our new model is piecewise linear and employs the same set of the three predictor variables. We do not use trade and quote (bid and ask) prices, and, as mentioned in chapter 5, our purpose is different from that of the trade classification algorithms [8, 76, 145, 169]. Those algorithms were designed for markets where full, correctly timestamped limit order book data, and trade sign in particular, are not available. Our empirical model is constructed to demonstrate a strong dependence between the three predictor variables and trade sign as evidence for an endogenous component in the order flow.

The space of the three predictor variables is partitioned into six regions. The trades within each region are signed according to rules derived from the two regularities in the order flow discussed earlier. The boundaries between the regions form a set of three partitioning planes. The coefficients of these planes are estimated over the first 30 days in the dataset. The estimation procedure employs three different methods to produce three corresponding coefficient vectors. The mean in-sample daily classification accuracy is then calculated over the first 30 days. The out-of-sample estimate of this accuracy is determined over the remaining 169 days. The calculations are performed separately for each stock and each coefficient vector. We show that the new model with an intuitively interpretable set of coefficients outperforms the k-NN classifier and achieves an average out-of-sample accuracy of 74.38% (SD=4.25%). We also find that two of
the six regions which represent, on average, 16.79% of the total number of daily trades, have each an average classification accuracy exceeding 91.50%.

### 6.1 Dataset - 12 ASX stocks, 199 days

We use the same dataset as in chapter 5 ([35]). To summarise, the dataset consists of trade, order, and order book data, on a single transaction level, for 12 large stocks on the Australian Stock Exchange (ASX). The data were collected by the ASX over 199 trading days, between 11 November 2002 and 27 August 2003. The dataset contains 2,355,334 trades. Our analysis is restricted to buyer-initiated and seller-initiated trades only. There are 2,184,046 such trades in the dataset (92.73%), of which 50.44% are buyer-initiated. Trades resulting from the same market order are aggregated together (as in chapter 5 [35]), whereby an aggregated trade becomes a proxy for that market order\(^3\). After aggregation we have 1,542,205 buyer-initiated and seller-initiated trades, with 51.78% of them being buyer-initiated. The first five aggregated trades on each day are omitted in order to use the same set of trades as in chapter 5 ([35])\(^4\). More information on the dataset can be found in section 5.1.

### 6.2 The piecewise linear model

The trade sign inference model proposed in chapter 5 ([35]) is based on the k-nearest-neighbour, which is a local, non-parametric, memory-based classifier. That model is a starting point for the development of our parametric model. In particular, we use the same set of predictor variables as employed for the k-NN classifier with the highest predictive accuracy. This set contains the following three variables:

\(^3\)We do not make a distinction between market orders and marketable limit orders. The latter are limit orders priced for immediate execution.

\(^4\)Chapter 5 and Ref. [35] used the omitted trades to obtain lagged values of model variables.
Trade sign - parametric model

There is one target variable (inference target), the trade sign $\epsilon_n$. To simplify the notation, the trade index $n$ will be omitted in the remainder of the chapter. The predictor variables will be denoted as $a$, $b$, and $s$, respectively, while $\epsilon$ will stand for the target variable. All three predictor variables are measured in the same units, that is number of shares. The trade sign is a binary variable which represents buyer-initiated and seller-initiated trades as $+1$ and $-1$, respectively.

To obtain some insight into the relationship between the three predictor variables and the target variable we construct two types of histograms using the first 30 trading days of the NCP stock. Six classification regions $r_i$, $i = 1 \ldots 6$, are numbered from 1 to 6. Bin size is 0.2x0.2.

- $a_n$ - total volume at the best ask price in the limit order book, recorded just before an order which caused the $n$-th trade; $n$ is an index of aggregated trades over a single day.

- $b_n$ - total volume at the best bid price in the limit order book, recorded just before an order which caused the $n$-th trade.

- $s_n$ - size of the $n$-th trade.

Figure 6.1: Ratio of trade counts for buyer-initiated (Bi) and seller-initiated (Si) trades, for the first 30 trading days of the NCP stock. Six classification regions $r_i$, $i = 1 \ldots 6$, are numbered from 1 to 6. Bin size is 0.2x0.2.
Figure 6.2: Trade count density of combined buyer-initiated (Bi) and seller-initiated (Si) trades, for the first 30 trading days of the NCP stock. Six classification regions $r_i$, $i = 1 \ldots 6$, are numbered from 1 to 6. Bin size is 0.2x0.2.

The two histograms for the NCP stock in Figures 6.1 and 6.2 qualitatively agree with the corresponding histograms for most of our stocks. The first type of histogram, shown in Figure 6.1, depicts a trade count ratio. The trade count ratio is defined as a ratio of a buyer-initiated bin trade count to a seller-initiated bin trade count. A bin trade count is the total number of trades in a given histogram bin. A trade count for buyer(seller)-initiated trades counts the trades with the specified sign only. The trade count ratio is shown as a function of the imbalance in the order book $imb = \ln(a / b)$ and the ratio $sbr$ of the trade size $s$ and the total volume at the best bid $b$. The presence of a square in a given histogram bin indicates a dominance of buyer-initiated trades, while a diamond stands for a majority of seller-initiated trades. A small dot represents approximately equal ($\pm 10\%$) trade counts. Bins without any trades are marked by a blank space without any symbol. It can be seen that there are several well defined regions, each dominated by a particular trade sign. The boundaries between the regions seem to form three straight lines, horizontal,
diagonal, and vertical. To highlight these features, three thin lines have been added to the figure. The horizontal line represents the condition \( sbr = 0 \), which means that \( s = b \). The diagonal line can be shown to correspond to the condition \( s = a \). The vertical line, on the other hand, represents the situation where \( imb = 0 \), which is equivalent to \( a = b \). The regions delineated by the three lines have been numbered from one to six and will be denoted as \( r_i, i = 1 \ldots 6 \). Regions \( r_2 \) and \( r_4 \) seem to be dominated by buyer-initiated trades, while regions \( r_3 \) and \( r_5 \) both have a majority of seller-initiated trades. The other two regions, \( r_1 \) and \( r_6 \), do not have an obvious dominant trade sign. They are sparsely occupied and have approximately similar numbers of squares and diamonds. We note that the lower right corner of region \( r_3 \) does not show a clear majority either.

The second type of histogram, shown in Figure 6.2, depicts a trade count density for combined buyer-initiated and seller-initiated trades. The density was calculated by dividing a given bin trade count by a total number of trades in all bins. The axes in the figure, as well as the lines separating the regions, and the region numbers are the same as in Figure 6.1. The trade count density values were transformed to a relative scale from 0 to 1, and mapped to shades of grey, from white to black. The white colour indicates very few or no trades, while the black stands for the highest trade count density. It is evident that parts of the horizontal and diagonal separating lines are located over areas of high trade concentration. Those areas seem to closely follow the course of the lines and do not reach further than half of the bin size away from the lines, in either axial direction. A large cluster of trades can be seen in regions \( r_3 \) and \( r_4 \). The bin density in that cluster is not as high as the density over the separating lines described above but the cluster covers a larger area. Regions \( r_1, r_2, r_5, \) and \( r_6 \) appear to have few trades beyond the areas under the horizontal and diagonal separating lines. The lower right corner of region \( r_3 \), mentioned earlier in the context of Figure 6.1, has also a very low trade count density. It is unclear if
trades, within bins cut in half by the separating lines, have a preference for which side of a given line they are on.

The analysis of the histograms has revealed the existence of regions dominated by particular trade signs. The boundaries of the regions are clearly delineated and form a set of three separating lines. The only exceptions are regions \( r_1 \) and \( r_6 \). It is not obvious where the boundary line between them is located and what their dominant trade signs are in areas outside the horizontal and diagonal separating lines. However, the trade count density in these areas is very low and, consequently, their contribution to the total trade sign classification accuracy should be minimal. The discovered features and the two regularities discussed in the introduction lead us to propose a parametric trade sign inference model specified by formula 6.1 below. The six functions \( \epsilon_i, \quad i = 1 \ldots 6 \), define six disjoint regions in the space of the three predictor variables. These six regions are generalisations of the regions shown in Figures 6.1 and 6.2, and are denoted in the same way, i.e. \( r_i, \quad i = 1 \ldots 6 \). Each function can assume only two values: 0 and one of the following two, either +1 or −1. For a given combination of the three predictor variables \( a, b, \) and \( s \) only one function \( \epsilon_i \) assumes a non-zero value due to mutually exclusive sets of conditions imposed on the predictor variables. The boundaries of the regions form a set of three boundary planes which are generalisations of the diagonal, horizontal, and vertical separating lines in the histograms. The boundary planes are defined by the following conditions: \( s = f_D(a) \), \( s = f_H(b) \), and \( a = f_V(b) \). The conditions employ three linear boundary functions, \( f_D(a) \), \( f_H(b) \), and \( f_V(b) \). The four coefficients, \( \alpha_a \), \( \beta_a \), \( \alpha_b \), and \( \beta_b \) form a coefficient vector \( \mathbf{c} \) and allow us to search for the optimal locations of the separating planes. The third boundary function \( f_V(b) \) has its own coefficients defined entirely in terms of the coefficients of the other two boundary functions. This constraint ensures that the three separating planes will always intersect along a single line. Consequently, the number of regions will stay fixed at six, irrespective of the values of the four coefficients, as long as \( \alpha_a \neq 0 \) and \( \alpha_b \neq 0 \).
\( \epsilon = p(c) \)  \hspace{1cm} (6.1)

\[
p(c) = \sum_{i=1}^{6} \epsilon_{r_i}, \quad c = (\alpha_a, \beta_a, \alpha_b, \beta_b)
\]

\[
\epsilon_{r_1} = \begin{cases} 
-1 & \text{if } s > f_D(a) \text{ and } a > f_V(b) \\
0 & \text{otherwise}
\end{cases}
\]

\[
\epsilon_{r_2} = \begin{cases} 
+1 & \text{if } s \leq f_D(a) \text{ and } s > f_H(b) \\
0 & \text{otherwise}
\end{cases}
\]

\[
\epsilon_{r_3} = \begin{cases} 
-1 & \text{if } s \leq f_H(b) \text{ and } a > f_V(b) \\
0 & \text{otherwise}
\end{cases}
\]

\[
\epsilon_{r_4} = \begin{cases} 
+1 & \text{if } s \leq f_D(a) \text{ and } a \leq f_V(b) \\
0 & \text{otherwise}
\end{cases}
\]

\[
\epsilon_{r_5} = \begin{cases} 
-1 & \text{if } s > f_D(a) \text{ and } s \leq f_H(b) \\
0 & \text{otherwise}
\end{cases}
\]

\[
\epsilon_{r_6} = \begin{cases} 
+1 & \text{if } s > f_H(b) \text{ and } a \leq f_V(b) \\
0 & \text{otherwise}
\end{cases}
\]

\[
f_D(a) = \alpha_a a + \beta_a, \quad \alpha_a \neq 0
\]

\[
f_H(b) = \alpha_b b + \beta_b, \quad \alpha_b \neq 0
\]

\[
f_V(b) = \frac{\alpha_b b + \beta_b - \beta_a}{\alpha_a}
\]

The proposed parametric model defined by formula 6.1 can be considered a piecewise linear model in respect to the six functions \( \epsilon_{r_i}, \ i = 1 \ldots 6 \), and the three linear boundary functions, \( f_D(a), f_H(b), \) and \( f_V(b) \). To estimate the model we will look for a coefficient vector \( c \) which maximises \( A_p \), where \( A_p \) denotes the classification accuracy of the function \( p(c) \). We employ three methods for this purpose. The simple method is derived from the exact arrangement of the three separating lines shown in Figures 6.1 and 6.2. Its coefficient vector \( c_{\text{smp}} \) is set arbitrarily to \((1, 0, 1, 0)\), resulting in the simplified boundary functions: \( f_D(a) = a, \ f_H(b) = b, \) and \( f_V(b) = b \). The second method uses the Nelder-Mead local optimiser [142, 166, 181] to maximise \( A_p \). This algorithm performs a local search in the space of the four coefficients, starting from \( c_{\text{smp}} \). The
result of the search is referred to as the Nelder-Mead optimised coefficient vector \( c_{nm} \). The third method is a global optimiser based on a recently developed particle swarm optimisation algorithm [129, 174, 217]. The optimiser searches the neighbourhood of \( c_{smp} \) with a swarm of virtual particles. The particles fly through the coefficient space and home in on the maxima of \( A_p \). The number of particles is set to 200. The initial neighbourhood size is set to \([0.5, 1.5]\) for \( \alpha_a \) and \( \alpha_b \), and to \([-1, 1]\) for \( \beta_a \) and \( \beta_b \). As the search progresses, a series of new, smaller neighbourhoods are constructed around the latest locally optimal vector with the highest \( A_p \) among the visited points. The process is continued until the classification accuracy can no longer be improved by a specified increment value. The best solution found is selected as the PSO optimised coefficient vector \( c_{pso} \).

The optimised coefficient vectors \( c_{nm} \) and \( c_{pso} \) are estimated separately by following the same procedure, described below. The estimation is performed over the subset \( E \), which comprises the first 30 days in the dataset. To prevent overfitting, where an estimated solution does not generalise to unseen data, we adopt an approach based on the ten-fold cross-validation [31, 192]. The whole period of 30 days is divided into 10 consecutive subperiods, called folds, each of the same length of 3 days. The estimation is conducted 10 times, each time on a different subset \( E_i \) with 27 days. During the \( i \)-th estimation, where \( i = 1, ..., 10 \), the \( i \)-th fold is omitted. An optimisation algorithm looks for a vector \( c \) which maximises \( A_p \) over a given subset \( E_i \). The classification accuracy is calculated as a single value over all days in \( E_i \), which is equivalent to calculating daily classification accuracy values and weighting them by the number of trades on corresponding days. We have chosen to calculate the accuracy this way because the k-NN classifier constructed in chapter 5 ([35]) achieved the best results with the training interval of 30 days. The whole process produces 10 estimates of an optimised coefficient vector, and the average optimised vector is determined by computing mean coefficient values across those 10 estimates.
Subsequently we use the subset $E$ and the three coefficient vectors, $c_{\text{smp}}$, $c_{\text{nm}}$, and $c_{\text{pso}}$, to calculate mean values of the in-sample daily classification accuracy, separately for each stock and each coefficient vector. The mean daily accuracy is computed as the mean of daily classification accuracy values of $A_p$. The same procedure is performed over the evaluation period, comprising the remaining 169 days in the dataset, in order to determine the out-of-sample daily classification accuracy.

The data processing and statistical calculations in our experiment were implemented in the proprietary market surveillance and trading system called SMARTS®, and on the Matlab® computing platform. Two freely available Matlab toolboxes, NETLAB [164] and PSOt [30], were also used.

### 6.3 Results

The in-sample classification accuracy statistics for the piecewise linear model with the three coefficient vectors $c_{\text{smp}}$, $c_{\text{nm}}$, and $c_{\text{pso}}$ are presented in Table 6.1. The table also shows statistics for the coefficients of the PSO optimised vector $c_{\text{pso}}$. The results were calculated using the first 30 days in the dataset. The table reports the PSO optimised vectors only because all but a few of their mean coefficients are located further away from the corresponding mean coefficients of the simple vector $c_{\text{smp}}$ than the respective mean coefficients of the optimised vectors found by the Nelder-Mead algorithm. In other words, the coefficients produced by the local optimiser are located closer to the coefficients of $c_{\text{smp}}$. The distance between the coefficients was measured with an Euclidean distance metric, for each stock separately. The same relationship exists for the average optimised vectors across the 12 stocks, where the average $c_{\text{pso}}$ is $(0.997379, 0.032122, 0.997895, 0.044311)$, while the average $c_{\text{nm}}$ is $(1.000255, 0.000125, 1.001402, 0.000043)$. The coefficients are reported with six digits after the decimal point to emphasise the differences between them. The optimised vectors $c_{\text{nm}}$ and $c_{\text{pso}}$ do not result
Chapter 6. Trade sign - parametric model

in a substantially improved in-sample classification accuracy when compared to the simple vector $c_{smp}$. This is the case for the stock specific as well as the average (across the 12 stocks) mean accuracy. The average mean accuracy for the piecewise linear model with the vectors $c_{smp}$, $c_{nm}$, and $c_{pso}$ is equal to 75.31%, 75.35%, and 75.34%, respectively. The corresponding standard deviations are 3.65%, 3.65%, and 3.67%. The differences between the mean values, and between the standard deviations, are minimal.

<table>
<thead>
<tr>
<th>Stock</th>
<th>$c_{smp}$</th>
<th>$c_{nm}$</th>
<th>$c_{pso}$</th>
<th>$\alpha_a$</th>
<th>$\beta_a$</th>
<th>$\alpha_b$</th>
<th>$\beta_b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAB</td>
<td>79.65</td>
<td>79.70</td>
<td>79.69</td>
<td>0.999659</td>
<td>0.004462</td>
<td>0.999933</td>
<td>0.001693</td>
</tr>
<tr>
<td></td>
<td>2.28</td>
<td>2.27</td>
<td>2.27</td>
<td>0.000460</td>
<td>0.005209</td>
<td>0.000090</td>
<td>0.001818</td>
</tr>
<tr>
<td>BHP</td>
<td>75.56</td>
<td>75.60</td>
<td>75.55</td>
<td>0.999166</td>
<td>0.011990</td>
<td>0.999938</td>
<td>0.010441</td>
</tr>
<tr>
<td></td>
<td>3.53</td>
<td>3.55</td>
<td>3.57</td>
<td>0.002346</td>
<td>0.027007</td>
<td>0.001185</td>
<td>0.009774</td>
</tr>
<tr>
<td>CBA</td>
<td>72.71</td>
<td>72.68</td>
<td>72.66</td>
<td>1.000042</td>
<td>0.008883</td>
<td>1.000145</td>
<td>0.002786</td>
</tr>
<tr>
<td></td>
<td>3.59</td>
<td>3.60</td>
<td>3.58</td>
<td>0.002038</td>
<td>0.001387</td>
<td>0.007853</td>
<td>0.003218</td>
</tr>
<tr>
<td>ANZ</td>
<td>77.83</td>
<td>77.84</td>
<td>77.86</td>
<td>0.999973</td>
<td>0.000461</td>
<td>1.000050</td>
<td>0.001773</td>
</tr>
<tr>
<td></td>
<td>2.75</td>
<td>2.79</td>
<td>2.79</td>
<td>0.000099</td>
<td>0.001105</td>
<td>0.000451</td>
<td>0.005081</td>
</tr>
<tr>
<td>WBC</td>
<td>76.22</td>
<td>76.22</td>
<td>76.20</td>
<td>0.999855</td>
<td>0.002027</td>
<td>1.000309</td>
<td>0.001833</td>
</tr>
<tr>
<td></td>
<td>3.09</td>
<td>3.11</td>
<td>3.09</td>
<td>0.003048</td>
<td>0.003714</td>
<td>0.000403</td>
<td>0.002095</td>
</tr>
<tr>
<td>NCP</td>
<td>76.31</td>
<td>76.33</td>
<td>76.32</td>
<td>0.999943</td>
<td>0.001015</td>
<td>1.000003</td>
<td>0.000024</td>
</tr>
<tr>
<td></td>
<td>2.91</td>
<td>2.90</td>
<td>2.90</td>
<td>0.002021</td>
<td>0.002350</td>
<td>0.000011</td>
<td>0.000046</td>
</tr>
<tr>
<td>RIO</td>
<td>80.08</td>
<td>80.07</td>
<td>80.02</td>
<td>0.999712</td>
<td>0.006057</td>
<td>0.998399</td>
<td>0.016841</td>
</tr>
<tr>
<td></td>
<td>2.86</td>
<td>2.80</td>
<td>2.92</td>
<td>0.006636</td>
<td>0.007617</td>
<td>0.002133</td>
<td>0.021967</td>
</tr>
<tr>
<td>WOW</td>
<td>75.70</td>
<td>75.74</td>
<td>75.72</td>
<td>0.999627</td>
<td>0.007419</td>
<td>0.998742</td>
<td>0.041676</td>
</tr>
<tr>
<td></td>
<td>3.20</td>
<td>3.14</td>
<td>3.13</td>
<td>0.001004</td>
<td>0.011051</td>
<td>0.001879</td>
<td>0.021138</td>
</tr>
<tr>
<td>FGL</td>
<td>67.67</td>
<td>67.74</td>
<td>67.75</td>
<td>0.979487</td>
<td>0.246303</td>
<td>0.977927</td>
<td>0.416271</td>
</tr>
<tr>
<td></td>
<td>6.28</td>
<td>6.33</td>
<td>6.19</td>
<td>0.033823</td>
<td>0.401725</td>
<td>0.336766</td>
<td>0.392509</td>
</tr>
<tr>
<td>SGB</td>
<td>77.31</td>
<td>77.49</td>
<td>77.52</td>
<td>0.999834</td>
<td>0.003600</td>
<td>0.999668</td>
<td>0.004165</td>
</tr>
<tr>
<td></td>
<td>3.14</td>
<td>3.15</td>
<td>3.15</td>
<td>0.000675</td>
<td>0.006450</td>
<td>0.000542</td>
<td>0.005432</td>
</tr>
<tr>
<td>SUN</td>
<td>73.53</td>
<td>73.48</td>
<td>73.72</td>
<td>0.999999</td>
<td>0.000076</td>
<td>1.000843</td>
<td>0.012721</td>
</tr>
<tr>
<td></td>
<td>3.68</td>
<td>3.71</td>
<td>3.87</td>
<td>0.000001</td>
<td>0.000159</td>
<td>0.001012</td>
<td>0.007791</td>
</tr>
<tr>
<td>MIG</td>
<td>71.14</td>
<td>71.29</td>
<td>71.13</td>
<td>0.991248</td>
<td>0.101173</td>
<td>0.998482</td>
<td>0.049018</td>
</tr>
<tr>
<td></td>
<td>6.53</td>
<td>6.39</td>
<td>6.63</td>
<td>0.027406</td>
<td>0.315834</td>
<td>0.011447</td>
<td>0.119704</td>
</tr>
<tr>
<td>Av.</td>
<td>75.31</td>
<td>75.35</td>
<td>75.34</td>
<td>0.997379</td>
<td>0.032122</td>
<td>0.997895</td>
<td>0.044311</td>
</tr>
<tr>
<td>value</td>
<td>3.65</td>
<td>3.65</td>
<td>3.67</td>
<td>0.005999</td>
<td>0.065300</td>
<td>0.006468</td>
<td>0.049214</td>
</tr>
</tbody>
</table>

Table 6.1: In-sample daily classification accuracy (%) and coefficients of the PSO optimised vector $c_{pso}$ for individual stocks - 30 estimation days.$^1$

$^1$For $c_{smp}$, $c_{nm}$, and $c_{pso}$ the first and second line for each stock show means and standard deviations of the classification accuracy, respectively. For coefficients $\alpha_a$, $\beta_a$, $\alpha_b$, and $\beta_b$ the first and second line for each stock show means and standard deviations of the PSO optimised coefficients, respectively. The optimised coefficients were calculated across the 10 subsets with 27 days, over the 30 estimation days. The first and second line of averages show average values of the corresponding stock specific statistics above them, calculated across the 12 stocks.

Abbreviated headings: $c_{smp}$, $c_{nm}$, $c_{pso}$ - piecewise linear model with the simple, Nelder-Mead optimised, and the PSO optimised coefficients, respectively; $\alpha_a$, $\beta_a$, $\alpha_b$, $\beta_b$ - PSO optimised coefficients.
Table 6.2: Out-of-sample daily classification accuracy (%) and fraction of the daily trade count (%) for individual stocks - 169 evaluation days.\(^\dagger\)

<table>
<thead>
<tr>
<th>Stock</th>
<th>k-NN</th>
<th>(c_{\text{smp}})</th>
<th>(c_{\text{nm}})</th>
<th>(c_{\text{pso}})</th>
<th>(r_1)</th>
<th>(r_2)</th>
<th>(r_3)</th>
<th>(r_4)</th>
<th>(r_5)</th>
<th>(r_6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAB</td>
<td>74.87</td>
<td>*78.54</td>
<td>78.54</td>
<td>78.52</td>
<td>60.79</td>
<td>93.22</td>
<td>74.29</td>
<td>73.73</td>
<td>94.27</td>
<td>56.38</td>
</tr>
<tr>
<td></td>
<td>3.08</td>
<td>3.09</td>
<td>3.09</td>
<td>3.09</td>
<td>0.53</td>
<td>12.16</td>
<td>37.84</td>
<td>37.81</td>
<td>12.02</td>
<td>0.53</td>
</tr>
<tr>
<td>BHP</td>
<td>70.08</td>
<td>*72.27</td>
<td>72.26</td>
<td>72.28</td>
<td>50.30</td>
<td>88.85</td>
<td>69.16</td>
<td>72.71</td>
<td>88.90</td>
<td>59.52</td>
</tr>
<tr>
<td></td>
<td>4.25</td>
<td>3.75</td>
<td>3.75</td>
<td>3.72</td>
<td>0.04</td>
<td>2.95</td>
<td>47.89</td>
<td>46.25</td>
<td>2.81</td>
<td>0.05</td>
</tr>
<tr>
<td>CBA</td>
<td>73.48</td>
<td>*75.59</td>
<td>75.60</td>
<td>75.56</td>
<td>58.05</td>
<td>92.10</td>
<td>71.04</td>
<td>71.93</td>
<td>90.91</td>
<td>52.02</td>
</tr>
<tr>
<td></td>
<td>2.25</td>
<td>2.74</td>
<td>2.75</td>
<td>2.74</td>
<td>0.55</td>
<td>11.49</td>
<td>38.35</td>
<td>38.14</td>
<td>10.96</td>
<td>0.52</td>
</tr>
<tr>
<td>ANZ</td>
<td>73.72</td>
<td>*76.44</td>
<td>76.43</td>
<td>76.42</td>
<td>57.00</td>
<td>94.30</td>
<td>69.47</td>
<td>71.19</td>
<td>92.83</td>
<td>56.95</td>
</tr>
<tr>
<td></td>
<td>3.27</td>
<td>3.42</td>
<td>3.42</td>
<td>3.42</td>
<td>0.28</td>
<td>10.04</td>
<td>41.41</td>
<td>39.09</td>
<td>8.86</td>
<td>0.32</td>
</tr>
<tr>
<td>WBC</td>
<td>73.41</td>
<td>*77.31</td>
<td>77.30</td>
<td>77.28</td>
<td>59.82</td>
<td>93.21</td>
<td>73.21</td>
<td>74.40</td>
<td>94.08</td>
<td>59.63</td>
</tr>
<tr>
<td></td>
<td>3.58</td>
<td>3.22</td>
<td>3.20</td>
<td>3.22</td>
<td>0.28</td>
<td>9.69</td>
<td>40.34</td>
<td>40.16</td>
<td>9.23</td>
<td>0.29</td>
</tr>
<tr>
<td>NCP</td>
<td>70.97</td>
<td>*75.17</td>
<td>75.18</td>
<td>75.18</td>
<td>54.87</td>
<td>92.57</td>
<td>71.87</td>
<td>71.40</td>
<td>91.66</td>
<td>51.87</td>
</tr>
<tr>
<td></td>
<td>3.66</td>
<td>4.03</td>
<td>4.02</td>
<td>4.02</td>
<td>0.15</td>
<td>6.27</td>
<td>43.60</td>
<td>43.79</td>
<td>6.03</td>
<td>0.16</td>
</tr>
<tr>
<td>RIO</td>
<td>76.34</td>
<td>*79.13</td>
<td>79.09</td>
<td>79.06</td>
<td>60.17</td>
<td>93.87</td>
<td>70.78</td>
<td>78.59</td>
<td>92.58</td>
<td>58.98</td>
</tr>
<tr>
<td></td>
<td>2.92</td>
<td>3.63</td>
<td>3.60</td>
<td>3.57</td>
<td>0.74</td>
<td>14.77</td>
<td>37.49</td>
<td>34.00</td>
<td>12.28</td>
<td>0.73</td>
</tr>
<tr>
<td>WOW</td>
<td>72.07</td>
<td>*75.11</td>
<td>75.09</td>
<td>75.04</td>
<td>58.86</td>
<td>92.81</td>
<td>75.26</td>
<td>68.51</td>
<td>93.32</td>
<td>57.41</td>
</tr>
<tr>
<td></td>
<td>4.15</td>
<td>4.04</td>
<td>4.02</td>
<td>4.03</td>
<td>0.25</td>
<td>8.44</td>
<td>39.74</td>
<td>42.19</td>
<td>9.12</td>
<td>0.26</td>
</tr>
<tr>
<td>FGL</td>
<td>62.69</td>
<td>63.84</td>
<td>64.20</td>
<td>64.12</td>
<td>83.33</td>
<td>92.81</td>
<td>66.75</td>
<td>71.68</td>
<td>85.18</td>
<td>42.86</td>
</tr>
<tr>
<td></td>
<td>6.08</td>
<td>6.71</td>
<td>6.67</td>
<td>6.67</td>
<td>0.01</td>
<td>1.57</td>
<td>50.76</td>
<td>46.26</td>
<td>1.38</td>
<td>0.01</td>
</tr>
<tr>
<td>SGB</td>
<td>72.58</td>
<td>*75.63</td>
<td>75.65</td>
<td>75.64</td>
<td>63.82</td>
<td>92.11</td>
<td>75.69</td>
<td>66.63</td>
<td>93.44</td>
<td>48.66</td>
</tr>
<tr>
<td></td>
<td>4.91</td>
<td>5.04</td>
<td>5.07</td>
<td>5.09</td>
<td>0.52</td>
<td>10.83</td>
<td>35.33</td>
<td>40.50</td>
<td>12.25</td>
<td>0.57</td>
</tr>
<tr>
<td>SUN</td>
<td>69.24</td>
<td>*73.70</td>
<td>73.71</td>
<td>73.55</td>
<td>60.93</td>
<td>90.73</td>
<td>69.77</td>
<td>69.66</td>
<td>92.01</td>
<td>51.49</td>
</tr>
<tr>
<td></td>
<td>3.87</td>
<td>4.68</td>
<td>4.63</td>
<td>4.63</td>
<td>0.51</td>
<td>10.98</td>
<td>39.44</td>
<td>38.05</td>
<td>10.44</td>
<td>0.54</td>
</tr>
<tr>
<td>MIG</td>
<td>67.39</td>
<td>*69.79</td>
<td>69.80</td>
<td>69.62</td>
<td>54.55</td>
<td>90.64</td>
<td>63.71</td>
<td>73.69</td>
<td>90.02</td>
<td>50.00</td>
</tr>
<tr>
<td></td>
<td>6.08</td>
<td>6.67</td>
<td>6.71</td>
<td>6.68</td>
<td>0.04</td>
<td>3.66</td>
<td>46.67</td>
<td>46.31</td>
<td>3.26</td>
<td>0.06</td>
</tr>
<tr>
<td>Av.</td>
<td>71.40</td>
<td>74.38</td>
<td>74.40</td>
<td>74.36</td>
<td>60.21</td>
<td>92.27</td>
<td>70.17</td>
<td>72.65</td>
<td>91.60</td>
<td>53.81</td>
</tr>
<tr>
<td>value</td>
<td>4.01</td>
<td>4.25</td>
<td>4.24</td>
<td>4.24</td>
<td>0.33</td>
<td>8.57</td>
<td>41.50</td>
<td>41.05</td>
<td>8.22</td>
<td>0.34</td>
</tr>
<tr>
<td></td>
<td>38.69</td>
<td>6.91</td>
<td>10.00</td>
<td>9.43</td>
<td>8.15</td>
<td>39.46</td>
<td>6.92</td>
<td>2.93</td>
<td>0.30</td>
<td></td>
</tr>
</tbody>
</table>

\(^\dagger\)For k-NN, \(c_{\text{smp}}\), \(c_{\text{nm}}\), and \(c_{\text{pso}}\) the first and second line for each stock show means and standard deviations of the classification accuracy, respectively. For regions \(r_i\), \(i = 1 \ldots 6\), the first and second line for each stock show means of the classification accuracy and means of the fraction of the daily trade count, respectively. The first and second line of averages show average values of the corresponding stock specific statistics above them. The third and fourth line of averages show average standard deviations of the regional classification accuracy and average standard deviations of the regional fraction of the daily trade count, respectively. All average values were calculated across the 12 stocks.

Abbreviated headings: k-NN - k-nearest-neighbour (\(k = 9\)); \(c_{\text{smp}}\), \(c_{\text{nm}}\), \(c_{\text{pso}}\) - piecewise linear model with the simple, Nelder-Mead optimised, and the PSO optimised coefficients, respectively; \(r_i\), \(i = 1 \ldots 6\) - six classification regions of the piecewise linear model with \(c_{\text{smp}}\). * - statistically significant at the level of 0.01.
Consequently, when measured by the in-sample classification accuracy, the three coefficient vectors seem to be equivalent and the simple vector $c_{smp}$ is the preferred one due to its intuitive interpretation. Table 6.2 shows the out-of-sample classification accuracy statistics for the best k-NN ($k = 9$) classifier as found in chapter 5 ([35]), and for the piecewise linear model with the three coefficient vectors $c_{smp}$, $c_{nn}$, and $c_{pso}$. The statistics were calculated over the 169 evaluation days. The differences between the mean classification accuracies for the coefficient vectors $c_{smp}$, $c_{nn}$, and $c_{pso}$ are very small and statistically not significant. Furthermore, for all stocks in our dataset, the mean accuracy for $c_{smp}$ is substantially greater than the mean accuracy for the best k-NN ($k = 9$) classifier reported in chapter 5 ([35]). The differences in their stock specific means, tested with the one tailed paired t-test, are statistically significant for 11 stocks at the level of 0.01, after the Bonferroni adjustment [184] to account for multiple comparisons\footnote{For each stock the total number of new and prior comparisons was 242. The piecewise linear model with the simple coefficient vector was compared against the 96 k-NN models constructed in chapter 5 ([35]). The two other piecewise linear models as well as the 144 prior comparisons in chapter 5 ([35]) were also accounted for.}. The only exception is the FGL stock. The statistical significance indicated in Table 6.2 refers only to tests between the piecewise linear model with the simple coefficient vector $c_{smp}$ and the k-NN ($k = 9$) classifier. The average mean out-of-sample classification accuracy for $c_{smp}$ is 74.38\% (SD=4.25\%). The corresponding average for the k-nearest-neighbour is 71.40\% (SD=4.01\%).

Table 6.2 also reports statistics for the classification accuracy and the fraction of the daily trade count for each of the six regions $r_i$, $i = 1 \ldots 6$, of the piecewise linear model. The statistics were calculated for the simple vector $c_{smp}$ over the 169 evaluation days in the dataset. The presented results quantitatively confirm the observations made during the analysis of Figures 6.1 and 6.2. In particular, regions $r_1$ and $r_6$ have the lowest average mean classification accuracy out of the six regions. The average mean accuracy for the two regions is 60.21\% and 53.81\%, respectively. These low values, however, do not have much influence on
the total classification accuracy \( A_p \) because the two regions together represent, across the 12 stocks, only 0.66\%\(^6\) of the total number of daily trades. The majority of trades occupy regions \( r_3 \) and \( r_4 \). The average mean accuracy for each of these regions is above 70\%, while their combined\(^7\) average mean fraction of the daily trade count is 82.55\%. As far as individual stocks are concerned, the mean classification accuracy can be as low as 56.75\% in \( r_3 \) (FGL) and as high as 78.59\% in \( r_4 \) (RIO). The stock-specific combined mean fraction of the daily trade count varies between 71.49\% (RIO) and 97.02\% (FGL). It needs to be noted that regional accuracies, calculated as a percentage of correctly inferred trade sign and weighted by a corresponding trade volume, are similar or slightly higher than the respective accuracies for the trade sign described above.

Two regions, \( r_3 \) and \( r_4 \), contain all trades for which \( s \leq a \) and \( s \leq b \). This means that the second regularity, transaction cost minimisation, is satisfied in these regions by design. The sign of the trades reflects the imbalance in the order book and is determined by finding the dominant side. In our model the dominant side in the book is the one with more volume at the best price, either bid or ask. The trade sign is set to buyer-initiated if \( a \leq b \), and to seller-initiated otherwise. This signing rule, and the achieved classification accuracy for \( r_3 \) and \( r_4 \), indicate that the first regularity, that is competition for order execution, is also satisfied (statistically). Interestingly, the other two regions, \( r_2 \) and \( r_5 \), sign trades in the opposite direction than the previous signing rule. Contrary to the first regularity, their trade sign reflects the non-dominant side in the order book. The cause of this reversal lies in the constraint imposed on trade size. Trades must satisfy the condition \( b < s \leq a \), to belong to \( r_2 \) and be classified as buyer-initiated; or the condition \( a < s \leq b \), to belong to \( r_5 \) and be classified...
as seller-initiated. The average mean accuracy for each of these two regions is above 91.50%, and together they represent, on average, 16.79% of the daily trade count. On an individual stock basis the mean accuracy has the lowest value of 85.18% in \( r_5 \) (FGL), and the highest value of 94.30% in \( r_2 \) (ANZ). The stock-specific combined mean fraction of the daily trade count varies between 2.96% (FGL) and 27.04% (RIO). The trade size constraints, and the achieved classification accuracy for \( r_2 \) and \( r_5 \), indicate that, if there is a conflict between the two regularities, then traders will follow the second one, that is transaction cost minimisation. Furthermore, due to the high classification accuracy in \( r_2 \) and \( r_5 \), the overall classification accuracy \( A_p \) tends to be higher for stocks with a relatively high fraction of the daily trade count in these two regions.

The above results suggest that \( c_{\text{mp}} \) is the best coefficient vector out of the three vectors tested, in respect to both classification accuracy and intuitive interpretation. The formula 6.1 can therefore be rewritten, and become the simpler formula given in 6.2 below.

\[
\epsilon = \sum_{i=1}^{6} \epsilon_{r_i} \quad (6.2)
\]

\[
\begin{align*}
\epsilon_{r_1} &= \begin{cases} 
-1 & \text{if } s > a \text{ and } a > b \\
0 & \text{otherwise}
\end{cases} \\
\epsilon_{r_2} &= \begin{cases} 
+1 & \text{if } s \leq a \text{ and } s > b \\
0 & \text{otherwise}
\end{cases} \\
\epsilon_{r_3} &= \begin{cases} 
-1 & \text{if } s \leq b \text{ and } a > b \\
0 & \text{otherwise}
\end{cases} \\
\epsilon_{r_4} &= \begin{cases} 
+1 & \text{if } s \leq a \text{ and } a \leq b \\
0 & \text{otherwise}
\end{cases} \\
\epsilon_{r_5} &= \begin{cases} 
-1 & \text{if } s > a \text{ and } s \leq b \\
0 & \text{otherwise}
\end{cases} \\
\epsilon_{r_6} &= \begin{cases} 
+1 & \text{if } s > b \text{ and } a \leq b \\
0 & \text{otherwise}
\end{cases}
\]
The results reported in the tables also suggest that the piecewise linear model with the simple coefficient vector $c_{\text{smp}}$ is superior to the local non-parametric model proposed in chapter 5 ([35]), due to the higher classification accuracy and the intuitive interpretation. We note that the average mean classification accuracy across the 12 stocks, for each of the three coefficients vectors in Table 6.2, is approximately 1% smaller than the corresponding average in Table 6.1. The average standard deviations, on the other hand, are larger in Table 6.2. The differences between mean accuracies for the same individual stocks in the two tables are even more pronounced, varying in value and sign. The observed differences are most probably due to the short estimation period of 30 trading days, relative to the 169 trading days of the evaluation period.

6.4 Conclusions

In this chapter we developed an empirical model for trade sign inference. Our model is a piecewise linear parameterisation of the model proposed in chapter 5 ([35]). The model employs three predictor variables, the volume at the best bid and at the best ask just before a trade, and trade size. There are four parameters which serve as coefficients of the boundary planes that partition the space of the three predictor variables into six regions. Each region has a dominant trade sign associated with it. All trades belonging to a given region are classified as having the dominant sign of that region. The best values of the four coefficients, in terms of the classification accuracy and parsimony, were found by constructing and evaluating the piecewise linear model with three different coefficient vectors. The simple coefficient vector $c_{\text{smp}}$, equal to $(1, 0, 1, 0)$, was shown to perform equally well as the locally optimised (Nelder-Mead) vector $c_{\text{nm}}$ and the globally optimised (PSO) vector $c_{\text{pso}}$. The simple vector $c_{\text{smp}}$ was selected as the best vector because of its intuitive interpretation. Our piecewise linear model outperforms the k-NN ($k = 9$) classifier developed in chapter 5 ([35]), on
a stock specific basis (11 out of 12 stocks) as well as across the 12 stocks. The out-of-sample statistics for individual stocks were calculated over the 169 trading days and are significant at the level of 0.01. The average mean classification accuracy for the new model with the simple vector $c_{smp}$ is 74.38% (SD=4.25%). This value is 2.98% above the average mean of 71.40% (SD=4.01%) reported in chapter 5 ([35]). The overall classification performance of our new model indicates a strong dependence between trade sign and the three predictor variables, and provides evidence for an endogenous component in the order flow.

The proposed piecewise linear model with the simple coefficient vector $c_{smp}$ partitions the space of the three predictor variables into six regions. The classification accuracy and the fraction of the daily trade count vary between the regions. Two regions for which $s \leq a$ and $s \leq b$ have a combined average mean fraction of the daily trade count of 82.55%, while each of them has an average mean classification accuracy above 70%. The trade sign within these regions reflects the dominant side in the order book. These results suggest that most of the trades within the two regions satisfy the first regularity, that is competition for order execution. The second regularity, on the other hand, is satisfied by all trades in these regions by design. Two other regions, where either $b < s \leq a$ or $a < s \leq b$, together represent, on average, 16.79% of the total number of daily trades. They both have an average mean classification accuracy exceeding 91.50%, while their trade sign reflects the non-dominant side in the order book. The results for these two regions indicate that when there is a conflict between the two regularities then the second, that is transaction cost minimisation, prevails. The remaining two regions contain a small number of trades whose size is larger than the volume on both sides of the order book. Their combined average of the daily trade count is only 0.66%, which further reinforces the evidence for the second regularity. Consequently, the regions’ influence on the overall trade sign classification accuracy of our model is negligible.

The daily classification accuracy of the piecewise linear model developed in
this chapter could be used as a new order flow metric. The temporal evolution of the metric, and a question of its privileged timescale, are good topics for future research. The new model captures the two regularities discussed in the introduction, that is competition for order execution and transaction cost minimisation. These regularities are reflected in the relationship between the predictor variables and trade sign, and can probably explain the monotonically increasing accuracy of the model proposed in chapter 5 ([35]). That study reported that the classification accuracy of the k-NN classifier increased with the length of the training interval, which indicated a stationarity of a corresponding length. It appears that the increase in the training interval length provided more data points (trades) for the estimation, which in turn allowed for a better approximation of the relationship between the variables concerned. The observed stationarity could therefore be a consequence of the two regularities operating on a single trade level, as discussed in this chapter. We also believe that the same two regularities are involved in the interplay [147] between the long memory in market order sign [39, 147], the long memory in trade size and the long memory in the volume in the limit order book [147]. A rigorous investigation of this idea is another possible direction for further work.
Chapter 7

Conclusions

The motivation for the research described in this thesis arose from a number of factors. The recent availability of high frequency limit order book data was probably the most important one. Without those data the thesis could not have been written. However, the difficulty of obtaining such data has contributed to a relative scarcity of studies employing them. The economic significance of the problem of optimal order submission, and of transaction costs in particular, has also played a part. Finally, it is our view that data mining has a substantial potential for uncovering patterns and relationships not identified by the traditional top-down modelling approach. We believe that the work presented in this thesis, specifically in chapters 4 to 6, has demonstrated the validity of this view.

The aims of this thesis are to analyse and build computational models for order submissions on the Australian Stock Exchange, an order-driven market with a public electronic limit order book. We were able to obtain a detailed record of ASX transactions, comprising all trades and orders with correct timestamps. The focus of the thesis is on the trade implementation problem faced by a trader who wants to transact a buy or sell order of a certain size. We use two approaches to model building, top-down and bottom-up. The traditional, top-down approach is applied to develop an optimal order submission plan for an order which is too large to be traded immediately without a prohibitive price impact. We present
an optimisation framework and some solutions for non-stationary and non-linear price impact and price impact risk. The second, bottom-up, or data mining, approach is employed for trade sign inference, where trade sign is defined as the side which initiates both a trade and the market order that triggered the trade. We are interested in an endogenous component of the order flow, as evidenced by the predictable relationship between trade sign and the variables used to infer it. We want to discover the rules which govern the trade sign, and establish a connection between them and two empirically observed regularities in market order submissions, competition for order execution and transaction cost minimisation. To achieve the above aims we first use an exploratory analysis of trade and limit order book data, and then develop a non-parametric and a parametric trade sign model.

There are four experimental chapters in the thesis. In chapter 3 we construct various order submission plans for three large stocks on the ASX. An analytic framework for minimising transaction costs is developed first, and a closed-form solution is derived for simplified transaction cost functions. To obtain solutions for the cases which are analytically intractable, with non-stationary and non-linear price impact and price impact risk, we apply deterministic discrete time dynamic programming. The optimal trading plans are generated for two levels of a trader’s aggressiveness. The generated optimal plans, as well as plans corresponding to some other, simpler strategies, are then evaluated by a trading simulator and compared. It was found that the proposed transaction costs model produces fairly good forecasts of the variance of the execution shortfall. However, the expected shortfalls are dominated by the standard deviation of the shortfall, which is due to price trends in our dataset. We conclude that high price volatility makes the task of transaction costs forecasting difficult. An improvement could perhaps be made by incorporating into the model some measure of future price direction. The presented optimisation framework is static. In the future we would like to investigate models which actively update their trading plans in
response to changing price and liquidity in the order book. Other extensions to the model could include, for example, permanent price impact, some measure of the variance of the forecasts of daily trading volume, and serial correlation in price returns.

An exploratory analysis of trade and order book data is conducted in chapter 4. The novel contribution of the chapter is an application of the self-organising map to the unsupervised clustering of trades into buyer-initiated and seller-initiated categories. The visualisation of the transformed data revealed that buyer-initiated and seller-initiated trades form two distinct clusters. Furthermore, the uncovered clusters correspond to respective states of market non-equilibrium as manifested by volume imbalances in the limit order book. The exploration conducted in that chapter is a preparation for model building in chapters 5 and 6.

In chapter 5 we investigated a regularity in market order submission strategies. An empirical model for trade sign inference was developed using supervised learning and transaction level data for twelve large stocks on the ASX. We proposed the k-nearest-neighbour classifier as a local non-parametric trade sign inference model. To find a classifier with the highest accuracy we perform a search across various predictor variable sets, training interval lengths, and values of the classifier’s parameter $k$. The performance of the k-nearest-neighbour is compared against that of the linear logistic regression and two other classifiers, trade continuation, and majority vote. The k-nearest-neighbour classifier is shown to be superior, requiring only three predictor variables and achieving an average out-of-sample accuracy of 71.40% (SD=4.01%), across all of the tested stocks. These results suggest that there exists a regularity in market order submissions. Furthermore, a non-linear approach seems to produce a more parsimonious trade sign inference model with a higher out-of-sample classification accuracy than a more complex linear logistic regression model. The best set of predictor variables
found in that chapter is subsequently used in chapter 6 to develop a parameterised trade sign model.

Chapter 6 was the last experimental chapter. In that chapter we proposed a piecewise linear parameterisation of the non-parametric model developed in chapter 5. The model employs three predictor variables, the volume at the best bid and at the best ask just before a trade, and trade size. The space of three predictor variables is partitioned into six regions. Each region has a dominant trade sign associated with it, and all trades belonging to a given region are classified as having that sign. Our piecewise linear model achieves an average out-of-sample classification accuracy of 74.38% (SD=4.25%), and outperforms the k-NN ($k = 9$) classifier developed in chapter 5. The result is statistically significant, after adjusting for multiple comparisons. The overall classification performance of the piecewise linear model indicates a strong dependence between trade sign and the three predictor variables, and provides evidence for an endogenous component in the order flow. Moreover, the rules for trade sign classification, derived from the six regions in the space of predictor variables, reflect two regularities observed in market order submissions, competition for order execution and transaction cost minimisation. An interpretation of the classification accuracy within the model’s regions offers therefore new insights into the relationship between the two regularities observed.

The daily classification accuracy of the piecewise linear model developed in chapter 6 could be used as a new order flow metric. The temporal evolution of the metric and a question of its privileged timescale are good topics for future research. We also believe that the two regularities in the order flow are involved in the interplay [147] between the long memory in market order sign [39, 147], the long memory in trade size and the long memory in the volume in the limit order book [147]. A rigorous investigation of this idea is another possible direction for further work. Furthermore, according to the results presented in chapter 6 for buyer-initiated and seller-initiated trades, almost three quarters of trades result
from market orders submitted according to the two behavioural regularities. This means that about one quarter of trades are caused by market orders which do not follow those regularities. The temporary and permanent price impacts of such market orders remain an open problem, but it is possible that at least some of them are submitted by master traders who break the two rules in order to earn higher returns. Identification and analysis of those superior orders would be an interesting research topic. As far as commercial applications are concerned, it is not clear at this stage if the observed predictability of the trade sign can be profitably exploited. Some answers could perhaps be obtained by incorporating our model into the existing models of limit order execution and trading costs.

An integration of the the optimal order submission model of chapter 3 and the trade sign inference model of chapter 6 could form a foundation of a trading system capable of two-level optimisation of large trades. The dynamic programming method would provide a high-level optimisation, generating a trading plan consisting of trading targets for a selected number of intraday intervals. The trade sign model, on the other hand, could be used to optimise trading at a low-level, within a single intraday interval. By following the two behavioural regularities governing the trade sign, that is competition for order flow and transaction cost minimisation, the trading system would be able to lower its transaction costs and to conceal its activity from other market participants. The latter, concealment of trading, is of crucial importance for controlling permanent price impact. Moreover, it is likely that there exists a relationship between, on one hand, temporary price impact $h(v)$ and temporary price impact risk $f(v)$, introduced in chapter 3, and, on the other hand, the trade sign model developed in chapter 6. Intuitively, large values of $h(v)$ and $f(v)$ on one side of the limit order book correspond to higher transaction costs on that side of the book and higher competition for order flow on the other side. It should therefore be possible to express separating functions employed in formula 6.1 in terms of some ratio of functions $h(v)$ and $f(v)$ for the two sides of the order book, respectively. The resulting model
would be a dynamic version of the static trade sign model represented by formula 6.2. Research into the temporal evolution of such a model and into that model’s responses to intraday price trends would be a good potential direction for the further development of the proposed trading system.
Appendix A

Discrete dynamic programming

This section provides a brief overview of stochastic, discrete time and discrete state dynamic programming. The description and formulae presented here have been derived from the material contained in Refs. [24, 162, 205, 207]. Let $S$ denote a set of states, and $A(s)$ a set of actions (decisions) available to an agent (decision maker) in a state $s(s \in S)$. An action $a$ selected by the agent in a state $s$, where $a \in A(s)$, leads to a transition to a new state $s'$ ($s' \in S$) with a probability distribution $P^1$, and earns (incurs) a reward (cost) $r = r(s'|s,a)$. The next state $s'$ and the reward $r$ are assumed to depend only on the current state $s$ and the action $a$, denoted formally as:

$$P(s_{t+1} = s', r_{t+1} = r|\{s_i\}, \{a_i\}, \{r_i\}, i = 0 \ldots t) = P(s_{t+1} = s', r_{t+1} = r|s_t, a_t),$$

where $t$ is a discrete time step index. This assumption is known as the Markov property, and defines a class of optimisation problems referred to as Markov Decision Processes (MDPs). A policy $\pi$ is a mapping of states $s \in S$ and actions $a \in A(s)$ to probability $\pi(s,a)$ that the agent in a state $s$ will select an action $a$. A value (cost) $V^\pi(s)$ of the state $s$ can then be defined as an expected sum of rewards earned by the agent by following the policy $\pi$ from the state $s$ over an

---

1Deterministic dynamic programming is a special case of stochastic dynamic programming, where probabilities assume values of 0 and 1 only.
assumed time horizon $T$:

$$V^\pi(s) = E_\pi \left\{ \sum_{k=0}^{T} \sigma^k r_{t+k+1} | s_t = s \right\} \quad (A.1)$$

$$r_{t+k+1} = r(s_{t+k+1} | s_{t+k}, a_{t+k} \in A(s_{t+k}))$$

The symbol $\sigma$ is a discounting factor which can take values in the range $[0, 1]$. The time horizon $T$ can be finite or infinite, with $\sigma < 1$ in the latter case. Formula A.1 can be easily transformed to the form known as the Bellman equation for the value function $V$:

$$V^\pi(s) = \sum_{a \in A(s)} \left\{ \pi(s, a) \sum_{s' \in S} p(s' | s, a) \left[ r(s' | s, a) + \sigma V^\pi(s') \right] \right\} \quad (A.2)$$

The agent’s goal is to maximise (minimise) the value (cost) function for all $s \in S$. Specifically, the agent searches for an optimal policy $\pi^*$ with an optimal value (cost) function $V^*$. A value function is considered optimal if it always assumes values greater than or equal to those of all other value functions:

$$V^*(s) = \max_{\pi} V^\pi(s) \quad (A.3)$$

There may exist more than one optimal policy, but all optimal policies have equivalent value functions. Furthermore, the optimal value function satisfies Bellman optimality equation:

$$V^*(s) = \max_{a \in A(s)} \left\{ \sum_{s' \in S} p(s' | s, a) \left[ r(s' | s, a) + \sigma V^*(s') \right] \right\} \quad (A.4)$$

The above equation captures algebraically the essence of discrete dynamic programming, Bellman's Principle of Optimality: “An optimal policy has the property that whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision”. There are number of algorithms that can solve the equation A.4. Three key algorithms are backward recursion, policy iteration, and function iteration. The first algorithm, backward recursion, is suitable for problems with a finite time horizon (finite $T$), while the other two, policy iteration and function
iteration, are used when the time horizon is infinite. Below we provide an outline of the backward recursion algorithm:

1. Initialisation: Define sets $S$ and $A$, probability distribution $P$, reward function $r$, discount factor $\sigma$, time horizon $T$, and final value $V_{T+1}(s), s \in S$. Let time index $t = T$.

2. Recursive formula: For every $s \in S$, calculate its optimal value $V_t$ and the optimal action $a_t$ at the time step $t$:

$$V_t(s) \leftarrow \max_{a \in A(s)} \left\{ \sum_{s' \in S} p(s'|s, a) \left[ r(s'|s, a) + \sigma V_{t+1}(s') \right] \right\}$$

$$a_t \leftarrow \arg \max_{a \in A(s)} \left\{ \sum_{s' \in S} p(s'|s, a) \left[ r(s'|s, a) + \sigma V_{t+1}(s') \right] \right\}$$

3. Stop condition: If $t = 1$ then stop else go back to step 2.

This is a very simple algorithm with a prohibitive computational complexity for larger problems. Many more advanced algorithms have been developed, and Ref. [24] is a good starting point for the study of this subject.
Appendix B

The self-organising map

The formulae and description of the SOM techniques presented below have been derived from Refs. [132, 133]. The incremental SOM algorithm, described in section 2.6.2, can be summarised using a mathematical notation in the following way:

\[
\begin{align*}
  f_{\text{SOM}} : & \quad \mathbf{x} \in \mathbb{R}^n \rightarrow \mathbf{m} \in \mathbb{R}^n \\
  \mathbf{m}_i(t+1) & = \mathbf{m}_i(t) + h_{c(\mathbf{x}),i}[\mathbf{x}(t) - \mathbf{m}_i(t)] \\
\end{align*}
\] (B.1)

The recursive formula B.1 for the prototype vector \( \mathbf{m}_i \) is re-evaluated for each \( \mathbf{x} \) in a sequence of input vectors, using a certain neighbourhood function \( h_{c(\mathbf{x}),i} \):

\[
\begin{align*}
  h_{c(\mathbf{x}),i} & = \alpha(t)h(d,t), \quad 0 < \alpha(t) < 1 \\
  \lim_{t \to \infty} h_{c(\mathbf{x}),i} & = 0 \\
  d & = \|\mathbf{r}_c - \mathbf{r}_i\|, \quad \mathbf{r}_c \in \mathbb{R}^2, \mathbf{r}_i \in \mathbb{R}^2 \\
  c & = c(\mathbf{x}) \\
  \forall i : & \quad \|\mathbf{x}(t) - \mathbf{m}_c(t)\| \leq \|\mathbf{x}(t) - \mathbf{m}_i(t)\| \\
\end{align*}
\] (B.2)

The second version of the SOM technique calculates optimal prototypes vectors \( \mathbf{m}_i^* \) for batches of input values \( \mathbf{x} \). This batch method is best described as a serious of steps in a short algorithm:

1. Initialisation: Set initial values to \( \mathbf{m}_i^* \).
2. Building of lists: For each $\mathbf{m}_i^*$ create a list $L_i$ of input values $\mathbf{x}$ that lie closer to $\mathbf{m}_i^*$ than to any other prototype vector, using the distance measure B.2.

3. Calculation of means: For each $\mathbf{m}_i^*$ find a union $U_i$ of its list $L_i$ and corresponding lists of its neighbours in a certain neighbourhood $N_i$. Update the values of $\mathbf{m}_i^*$ with means of vectors $\mathbf{x}$ over respective $U_i$.

4. Termination: If a steady solution has been found then stop, otherwise go back to step 2.
Appendix C

The k-nearest-neighbour

The $k$-NN classifier, described in section 2.6.3, computes the following function:

$$f_{knn} : x \in \mathbb{R}^n \rightarrow y \in C,$$

where $C$ is a finite set of membership classes. For the basic version of the $k$-nearest-neighbour the value of $y$ is determined simply as the most frequent class $c \in C$ within the $k$ element subset $K$ of the training data set $D$:

$$y = \arg\max_{c \in C} P(c|K)$$

$$\forall (d_i \in K) \forall (d_j \in D') : \|x_i - x\| \leq \|x_j - x\|$$

$$D' = D \setminus K, \ d = (x_1, \ldots, x_{n_x}, c)$$

$$\|x - x'\| = \left\{ \sum_{j=1}^{n_x} (x_j - x'_j)^2 \right\}^{\frac{1}{2}}$$

The above formulation uses the Euclidean distance metric, but other distance metrics can also be used.
Appendix D

Derivation of Formula 3.4

For convenience we first repeat the formulae introduced in chapter 3, retaining their original numbering:

\[ s_f = XS_0 - \sum_{k=1}^{N} n_k \tilde{S}_k \]  
\[ (3.1) \]

\[ S_k = S_{k-1} + \sigma \tau^{1/2} \xi_k \]  
\[ (3.2) \]

\[ \tilde{S}_k = S_{k-1} - h(v_k) + \tau^{-1/2} f(v_k) \tilde{\xi}_k \]  
\[ (3.3) \]

We are going to show that the expected value of the shortfall \( s_f \) is a function given by the equation 3.4:

\[ E(x) = \tau \sum_{k=1}^{N} v_k h(v_k) \]  
\[ (3.4) \]

Two additional variables to be used in the derivation are speed of trading \( v_k = \frac{n_k}{\tau} \), and the number of unsold shares remaining after the \( k \)-th time interval \( x_k = x_{k-1} - n_k \). The total number of shares to be sold is \( X = \sum_{k=1}^{N} n_k \). In the first step of derivation the original formulae 3.2 and 3.3 are transformed to the formulae D.1 and D.2, respectively. Formula 3.3 is a non-recursive version of 3.3, while formula D.2 is obtained by eliminating \( S_{k-1} \) through a substitution based on D.1. A non-recursive formula for \( x_k \) (D.3) is given as well.

\[ S_k = S_0 + \sigma \tau^{1/2} \sum_{i=1}^{k} \xi_i \]  
\[ (D.1) \]
\[ \tilde{S}_k = S_0 + \sigma^{1/2} \left\{ \sum_{i=1}^{k-1} \xi_i \right\} - h(v_k) + \tau^{-1/2} f(v_k) \tilde{\xi}_k \]  \hspace{1cm} (D.2) 

\[ x_k = x_{k-1} - n_k = X - \sum_{j=1}^{k} n_j = \sum_{j=k+1}^{N} n_j \]  \hspace{1cm} (D.3)

Subsequently the formulae D.2 and D.3 are used to derive a new formula (D.4) for the shortfall:

\[ sf = XS_0 - \sum_{k=1}^{N} n_k \left\{ S_0 + \sigma^{1/2} \left\{ \sum_{i=1}^{k-1} \xi_i \right\} - h(v_k) + \tau^{-1/2} f(v_k) \tilde{\xi}_k \right\} \]

\[ = XS_0 - \sum_{k=1}^{N} n_k S_0 - \sum_{k=1}^{N} n_k \sigma^{1/2} \left\{ \sum_{i=1}^{k-1} \xi_i \right\} + \sum_{k=1}^{N} n_k h(v_k) - \sum_{k=1}^{N} n_k \tau^{-1/2} f(v_k) \tilde{\xi}_k \]

\[ = XS_0 - X S_0 - \sigma^{1/2} \sum_{k=1}^{N} n_k \left\{ \sum_{i=1}^{k-1} \xi_i \right\} + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{-1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \]

\[ = -\sigma^{1/2} \{ n_1 0 + n_2 \xi_1 + n_3 (\xi_1 + \xi_2) + \ldots + n_k \sum_{i=1}^{k-1} \xi_i + \ldots + n_N \sum_{i=1}^{N-1} \xi_i \} \]

\[ + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \]

\[ = -\sigma^{1/2} \{ \xi_1 \sum_{j=2}^{N} n_j + \xi_2 \sum_{j=3}^{N} n_j + \ldots + \xi_k \sum_{j=k+1}^{N} n_j + \ldots + \xi_{N-1} n_N \} \]

\[ + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \]

\[ = -\sigma^{1/2} \{ \xi_1 x_1 + \xi_2 x_2 + \ldots + \xi_k x_k + \ldots + \xi_{N-1} x_{N-1} \} \]

\[ + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \]

\[ = -\sigma^{1/2} \sum_{k=1}^{N-1} \xi_k x_k + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \]  \hspace{1cm} (D.4)

Recalling that \( \xi_k \) and \( \tilde{\xi}_k \) are zero mean unit variance random variables which are independent of each other, we can use the equation D.4 to calculate the expected
value of the shortfall:

\[ E(sf) = E(x) \]

\[ = E\left\{ -\sigma \tau^{1/2} \sum_{k=1}^{N-1} \xi_k x_k + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \right\} \]

\[ = \tau \sum_{k=1}^{N} v_k h(v_k) \hspace{1cm} (D.5) \]

The derived formula D.5 is identical to formula 3.4.
Appendix E

Derivation of Formula 3.5

Having derived the formulae for the shortfall (D.4) and its expected value (D.5) in section D we can use them to prove the equation for shortfall variance which was given in chapter 3 as:

\[
V(x) = \sigma^2 \tau \sum_{k=1}^{N} x_k^2 + \tau \sum_{k=1}^{N} v_k^2 f(v_k)^2
\]  

(3.5)

Employing the classic definition of variance we obtain:

\[
V(sf) = E\{ sf - E(sf) \}^2
\]

\[
= E\left\{ -\sigma^{1/2} \sum_{k=1}^{N-1} \xi_k x_k + \tau \sum_{k=1}^{N} v_k h(v_k) - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \right\}^2
\]

\[
= E\left\{ -\sigma^{1/2} \sum_{k=1}^{N-1} \xi_k x_k - \tau^{1/2} \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \right\}^2
\]

\[
= E\left\{ \sigma^2 \tau \left\{ \sum_{k=1}^{N-1} \xi_k x_k \right\}^2 + \tau \left\{ \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \right\}^2 + 2\sigma^2 \tau \sum_{k=1}^{N-1} \xi_k x_k \sum_{k=1}^{N} v_k^2 f(v_k)^2 \right\}
\]

(E.1)
Appendix E. Derivation of Formula 3.5

Recalling that \( \xi_k \) and \( \tilde{\xi}_k \) are zero mean unit variance random variables which are independent of each other, while \( x_N = 0 \), we can further simplify formula E.1:

\[
V(sf) = V(x)
\]

\[
= E\{\sigma^2 \tau \left\{ \sum_{k=1}^{N-1} \xi_k x_k \right\}^2 + \tau \left\{ \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \right\}^2 \}
\]

\[
= E\{\sigma^2 \tau \sum_{k=1}^{N-1} \xi_k x_k \sum_{k=1}^{N-1} \xi_k x_k + \tau \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \sum_{k=1}^{N} v_k f(v_k) \tilde{\xi}_k \}
\]

\[
= E\{\sigma^2 \tau \sum_{k=1}^{N-1} \xi_k^2 x_k^2 + \tau \sum_{k=1}^{N} v_k^2 f(v_k)^2 \tilde{\xi}_k^2 \}
\]

\[
= \sigma^2 \tau \sum_{k=1}^{N-1} x_k^2 + \tau \sum_{k=1}^{N} v_k^2 f(v_k)^2
\]

\[
= \sigma^2 \tau \sum_{k=1}^{N} x_k^2 + \tau \sum_{k=1}^{N} v_k^2 f(v_k)^2 \quad (E.2)
\]

The derived formula E.2 is identical to formula 3.5.
Appendix F

Derivations for analytical solution in section 3.2

The derivations presented in this appendix are due to the author and Dr Richard Coggins. For convenience we first repeat the formulae introduced in chapter 3, retaining their original numbering:

\[ E(x) = \tau \sum_{k=1}^{N} v_k h(v_k) \]  \hspace{1cm} (3.4)

\[ V(x) = \sigma^2 \tau \sum_{k=1}^{N} x_k^2 + \tau \sum_{k=1}^{N} v_k^2 f(v_k)^2 \]  \hspace{1cm} (3.5)

\[ U(x) = E(x) + \lambda V(x) \]  \hspace{1cm} (3.6)

\[ v_k = \frac{x_{k-1} - x_k}{\tau} \]

\[ x_0 = X, \quad x_N = 0 \]

To facilitate an analytical solution the following assumptions are made:

\[ h(v_k) = \eta_k v_k \]

\[ f(v_k) = \alpha_k \]

\[ \tau = 1 \]

where the unit of \( \tau \) is one trading interval. We allow coefficients \( \eta, \alpha, \) and \( \sigma \) to be non-stationary. They are assumed constant within a given interval only, as
reflected by the index \( k \). We can now expand the formula for \( U(x) \) as follows:

\[
U(x) = E(x) + \lambda V(x)
\]

\[
= \{ \sum_{k=1}^{N} v_k h(v_k) \} + \lambda \{ \sigma_k^2 \sum_{k=1}^{N} x_k^2 + \sum_{k=1}^{N} v_k^2 f(v_k)^2 \}
\]

\[
= \sum_{k=1}^{N} \{ v_k h(v_k) + \lambda \sigma_k^2 x_k^2 + \lambda v_k^2 f(v_k)^2 \}
\]

\[
= \sum_{k=1}^{N} \{ \eta_k (x_{k-1} - x_k)^2 + \lambda \sigma_k^2 x_k^2 + \lambda \alpha_k^2 (x_{k-1} - x_k)^2 \}
\]

\[
= \sum_{k=1}^{N} \{ \lambda \sigma_k^2 x_k^2 + (\eta_k + \lambda \alpha_k^2) (x_{k-1} - x_k)^2 \} \tag{F.1}
\]

To minimise \( U(x) \) we need to find a solution where \( \frac{\partial U(x_k)}{\partial x_k} = 0 \) for \( k = 1, \ldots, N-1 \), in order to satisfy Bellman’s Principal of Optimality (appendix A). To derive formula 3.9 we are going to differentiate the utility function F.1 with respect to \( x_k \). The utility function is a sum of \( N \) terms, but it can be noticed that only two differentiated terms, the ones containing \( x_k \), may assume non-zero values:

\[
\frac{\partial U(x_k)}{\partial x_k} = \frac{\partial \{ \lambda \sigma_k^2 x_k^2 + (\eta_k + \lambda \alpha_k^2) (x_{k-1} - x_k)^2 \}}{\partial x_k}
\]

\[
+ \frac{\partial \{ \lambda \sigma_{k+1}^2 x_{k+1}^2 + (\eta_{k+1} + \lambda \alpha_{k+1}^2) (x_k - x_{k+1})^2 \}}{\partial x_k}
\]

\[
= 2 \{ \lambda \sigma_k^2 x_k + (\eta_k + \lambda \alpha_k^2) (x_{k-1} - x_k) + (\eta_{k+1} + \lambda \alpha_{k+1}^2) (x_k - x_{k+1}) \}
\]

\[
= 2 \{ \lambda \sigma_k^2 x_k - A_k (x_{k-1} - x_k) + A_{k+1} (x_k - x_{k+1}) \} \tag{F.2}
\]

where

\[
A_k = \eta_k + \lambda \alpha_k^2
\]

The derived formula F.2 is identical to formula 3.9. We can now look for a solution where the derivative is zero:

\[
\frac{\partial U(x_k)}{\partial x_k} = 0
\]
Appendix F. Derivations for analytical solution in section 3.2

\[ 0 = 2 \left\{ \lambda \sigma_k^2 x_k - A_k(x_{k-1} - x_k) + A_{k+1}(x_k - x_{k+1}) \right\} \]
\[ 0 = \lambda \sigma_k^2 x_k + A_k x_k + A_{k+1}x_k - A_k x_{k-1} - A_{k+1}x_{k+1} \]
\[ 0 = (\lambda \sigma_k^2 + A_k + A_{k+1}) x_k - (A_k x_{k-1} + A_{k+1}x_{k+1}) \]
\[ x_k = \frac{A_k x_{k-1} + A_{k+1} x_{k+1}}{B_k} \]  \( (F.3) \)

where
\[ B_k = \lambda \sigma_k^2 + A_k + A_{k+1} \]  \( (F.4) \)

There are two boundary conditions, \( x_0 = X \) and \( x_N = 0 \). We will use them to eliminate \( x_{k-1} \) and \( x_{k+1} \) from formula \( F.3 \). The first condition allows us to quickly calculate \( x_k \) for low values of \( k \). For \( k = 1 \) we obtain:
\[ x_1 = \frac{A_1 X + A_2 x_2}{B_1} \]  \( (F.5) \)

To calculate \( x_2 \) we substitute formula \( F.5 \) for \( x_1 \):
\[ x_2 = \frac{A_2 x_1 + A_3 x_3}{B_2} \]
\[ x_2 = \frac{A_2 (A_1 X + A_2 x_2)}{B_1 B_2} + \frac{A_3 x_3}{B_2} \]
\[ 0 = (A_2^2 - B_1 B_2) x_2 + (A_1 A_2 X + A_3 B_1 x_3) \]
\[ x_2 = \frac{A_1 A_2 X + A_3 B_1 x_3}{B_1 B_2 - A_2^2} \]  \( (F.6) \)

The derived formula for \( x_2 \) is used to calculate \( x_3 \):
\[ x_3 = \frac{A_3 x_2 + A_4 x_4}{B_3} \]
\[ x_3 = \frac{A_3 (A_1 A_2 X + A_3 B_1 x_3)}{B_3 (B_1 B_2 - A_2^2)} + \frac{A_4 x_4}{B_3} \]
\[ 0 = \left\{ A_2^2 B_1 - B_3 (B_1 B_2 - A_2^2) \right\} x_3 + \left\{ A_1 A_2 A_3 X + A_4 (B_1 B_2 - A_2^2) x_4 \right\} \]
\[ x_3 = \frac{A_1 A_2 A_3 X + A_4 (B_1 B_2 - A_2^2) x_4}{B_3 (B_1 B_2 - A_2^2) - A_3^2 B_1} \]  \( (F.7) \)

Having analysed the formulae for \( x_1 \), \( x_2 \), and \( x_3 \), we propose a more general formula for \( x_k \):
\[ x_k = \frac{\left( \prod_{i=1}^{k} A_i \right) X + A_{k+1} C_{k-1} x_{k+1}}{C_k} \]  \( (F.8) \)
where

$$C_k = B_k C_{k-1} - A_k^2 C_{k-2}$$  \hspace{1cm} (F.9)

$$C_0 = 1, \quad C_{-1} = 0$$

Formula F.8 can be proved through induction. We need to show that $x_{k+1}$ calculated from formula F.3, with formula F.8 substituted for $x_k$, is equal to formula F.8 for $k + 1$. We proceed as follows:

$$x_{k+1} = \frac{A_{k+1} x_k + A_{k+2} x_{k+2}}{B_{k+1}}$$

$$x_{k+1} = \frac{A_{k+1} \left( (\prod_{i=1}^{k} A_i) X + A_{k+1} C_{k-1} x_{k+1} \right)}{B_{k+1} C_k} + \frac{A_{k+2} x_{k+2}}{B_{k+1}}$$

$$0 = \left( A_{k+1} C_{k-1} - B_{k+1} C_k \right) x_{k+1} + \left( (\prod_{i=1}^{k+1} A_i) X + A_{k+2} C_k x_{k+2} \right)$$

$$x_{k+1} = \frac{\left( \prod_{i=1}^{k+1} A_i \right) X + A_{k+2} C_k x_{k+2}}{B_{k+1} C_k - A_{k+1} C_{k-1}}$$

$$x_{k+1} = \frac{\left( \prod_{i=1}^{k+1} A_i \right) X + A_{k+2} C_k x_{k+2}}{C_{k+1}}$$  \hspace{1cm} (F.10)

It can be seen that the derived formula F.10 is identical to formula F.8 calculated for $k + 1$.

The second boundary condition, $x_N = 0$, allows us to quickly calculate $x_k$ for values of $k$ close to $N$. For $k = N - 1$ we obtain:

$$x_{N-1} = \frac{A_{N-1} x_{N-2}}{B_{N-1}}$$  \hspace{1cm} (F.11)

To calculate $x_{N-2}$ we substitute formula F.11 for $x_{N-1}$:

$$x_{N-2} = \frac{A_{N-2} x_{N-3} + A_{N-1} x_{N-1}}{B_{N-2}}$$

$$x_{N-2} = \frac{A_{N-2} x_{N-3}}{B_{N-2}} + \frac{A_{N-1} (A_{N-1} x_{N-2})}{B_{N-2} B_{N-1}}$$

$$0 = A_{N-2} B_{N-1} x_{N-3} + (A_{N-1}^2 - B_{N-2} B_{N-1}) x_{N-2}$$

$$x_{N-2} = \frac{A_{N-2} B_{N-1} x_{N-3}}{B_{N-2} B_{N-1} - A_{N-1}^2}$$  \hspace{1cm} (F.12)
Appendix F. Derivations for analytical solution in section 3.2

The derived formula for \( x_{N-2} \) is used to calculate \( x_{N-3} \):

\[
x_{N-3} = \frac{A_{N-3}x_{N-4} + A_{N-2}x_{N-2}}{B_{N-3}}
\]

\[
x_{N-3} = \frac{A_{N-3}x_{N-4}}{B_{N-3}} + \frac{A_{N-2}(A_{N-2}B_{N-1}x_{N-3})}{B_{N-3}(B_{N-2}B_{N-1} - A_{N-1}^2)}
\]

\[
0 = A_{N-3}(B_{N-2}B_{N-1} - A_{N-1}^2)x_{N-4}
\]

\[
+ \{ A_{N-2}B_{N-1} - B_{N-3}(B_{N-2}B_{N-1} - A_{N-1}^2) \} x_{N-3}
\]

\[
x_{N-3} = \frac{A_{N-3}(B_{N-2}B_{N-1} - A_{N-1}^2)x_{N-4}}{B_{N-3}(B_{N-2}B_{N-1} - A_{N-1}^2) - A_{N-2}^2B_{N-1}}
\]

(F.13)

Having analysed the formulae for \( x_{N-1}, x_{N-2}, \) and \( x_{N-3} \), we propose another more general formula for \( x_k \):

\[
x_k = \frac{A_kD_{k+1}x_{k-1}}{D_k}
\]

(F.14)

where

\[
D_k = B_kD_{k+1} - A_{k+1}^2D_{k+2}
\]

(F.15)

\[
D_N = 1, \quad D_{N+1} = 0
\]

Formula F.14 can be proved through induction. We need to show that \( x_{k-1} \) calculated from formula F.3, with formula F.14 substituted for \( x_k \), is equal to formula F.14 for \( k - 1 \). We proceed as follows:

\[
x_{k-1} = \frac{A_{k-1}x_{k-2} + A_kx_k}{B_{k-1}}
\]

\[
x_{k-1} = \frac{A_{k-1}x_{k-2}}{B_{k-1}} + \frac{A_k(A_kD_{k+1}x_{k-1})}{B_{k-1}D_k}
\]

\[
0 = A_{k-1}D_kx_{k-2} + (A_k^2D_{k+1} - B_{k-1}D_k)x_{k-1}
\]

\[
x_{k-1} = \frac{A_{k-1}D_kx_{k-2}}{B_{k-1}D_k - A_k^2D_{k+1}}
\]

(F.16)

The derived formula F.16 is identical to formula F.14 calculated for \( k - 1 \). The two formulae for \( x_k \), F.8 and F.14, can be combined into the general formula for
Appendix F. Derivations for analytical solution in section 3.2

\( x_k \) given by 3.11. We first use formula F.14 to calculate \( x_{k+1} \) and then substitute the result for \( x_{k+1} \) in formula F.8:

\[
x_{k+1} = \frac{A_{k+1}D_{k+2}x_k}{D_{k+1}} \]

\[
x_k = \frac{(\prod_{i=1}^k A_i)X + A_{k+1}C_{k-1}x_{k+1}}{C_k} \]

\[
x_k = \frac{(\prod_{i=1}^k A_i)X + A_{k+1}C_{k-1}(A_{k+1}D_{k+2}x_k)}{C_kD_{k+1}} \]

\[
0 = (\prod_{i=1}^k A_i)D_{k+1}X + (A_{k+1}^2C_{k-1}D_{k+2} - C_kD_{k+1})x_k \]

\[
x_k = \frac{(\prod_{i=1}^k A_i)D_{k+1}X}{C_kD_{k+1} - A_{k+1}^2C_{k-1}D_{k+2}} \quad (F.17) \]

The derived formula F.17 is identical to formula 3.11.
Appendix G

Statistics for Chapter 4
G.1 Stock code: AMP

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>$\ln(\frac{Si%}{Bi%})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.00</td>
<td>1.19</td>
<td>-0.176</td>
</tr>
<tr>
<td>1</td>
<td>4.71</td>
<td>4.71</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>11.36</td>
<td>8.91</td>
<td>0.243</td>
</tr>
<tr>
<td>3</td>
<td>17.72</td>
<td>11.48</td>
<td>0.434</td>
</tr>
<tr>
<td>4</td>
<td>17.40</td>
<td>8.98</td>
<td>0.661</td>
</tr>
<tr>
<td>5</td>
<td>8.89</td>
<td>3.65</td>
<td>0.889</td>
</tr>
</tbody>
</table>

Table G.1: Percentage of the seller-initiated and buyer-initiated trades for the AMP stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.1: Daily normalised trade count (grey scale axis) for the AMP stock as a function of time and trade size (103x70 bins).
Figure G.2: Normalised trade count (grey scale axis) for the AMP stock as a function of $askvol$ and $bidvol$ (70x70 bins).

Figure G.3: Normalised trade count (grey scale axis) for the AMP stock after the SOM transformation as a function of $askvol$, $bidvol$, size, and $\epsilon_{cnt}$ (70x70 bins).
Appendix G. Statistics for Chapter 4

Figure G.4: Trade sign ratio (colour scale axis) for the AMP stock as a function of askvol and bidvol (70x70 bins).

Figure G.5: Trade sign ratio (colour scale axis) for the AMP stock after the SOM transformation as a function of askvol, bidvol, size, and $\epsilon_{\text{cut}}$ (70x70 bins).
G.2 Stock code: ANZ

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>Ln(Si/Bi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.38</td>
<td>4.11</td>
<td>-0.549</td>
</tr>
<tr>
<td>1</td>
<td>8.39</td>
<td>11.39</td>
<td>-0.306</td>
</tr>
<tr>
<td>2</td>
<td>13.77</td>
<td>16.70</td>
<td>-0.193</td>
</tr>
<tr>
<td>3</td>
<td>12.67</td>
<td>13.77</td>
<td>-0.083</td>
</tr>
<tr>
<td>4</td>
<td>6.99</td>
<td>6.45</td>
<td>0.081</td>
</tr>
<tr>
<td>5</td>
<td>1.86</td>
<td>1.53</td>
<td>0.200</td>
</tr>
</tbody>
</table>

Table G.2: Percentage of the seller-initiated and buyer-initiated trades for the ANZ stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.6: Daily normalised trade count (grey scale axis) for the ANZ stock as a function of time and trade size (103x70 bins).
Figure G.7: Normalised trade count (grey scale axis) for the ANZ stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.8: Normalised trade count (grey scale axis) for the ANZ stock after the SOM transformation as a function of \( \text{askvol} \), \( \text{bidvol} \), \( \text{size} \), and \( \epsilon_{\text{cut}} \) (70x70 bins).
Figure G.9: Trade sign ratio (colour scale axis) for the ANZ stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.10: Trade sign ratio (colour scale axis) for the ANZ stock after the SOM transformation as a function of \( \text{askvol} \), \( \text{bidvol} \), size, and \( \epsilon_{\text{cnt}} \) (70x70 bins).
G.3 Stock code: CBA

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>$\ln(\text{Si% Bi%})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.17</td>
<td>1.26</td>
<td>-0.075</td>
</tr>
<tr>
<td>1</td>
<td>6.05</td>
<td>5.82</td>
<td>0.038</td>
</tr>
<tr>
<td>2</td>
<td>13.63</td>
<td>11.95</td>
<td>0.132</td>
</tr>
<tr>
<td>3</td>
<td>17.69</td>
<td>13.60</td>
<td>0.263</td>
</tr>
<tr>
<td>4</td>
<td>12.96</td>
<td>8.88</td>
<td>0.378</td>
</tr>
<tr>
<td>5</td>
<td>4.35</td>
<td>2.63</td>
<td>0.503</td>
</tr>
</tbody>
</table>

Table G.3: Percentage of the seller-initiated and buyer-initiated trades for the CBA stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.11: Daily normalised trade count (grey scale axis) for the CBA stock as a function of time and trade size (103x70 bins).
Figure G.12: Normalised trade count (grey scale axis) for the CBA stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.13: Normalised trade count (grey scale axis) for the CBA stock after the SOM transformation as a function of \( \text{askvol}, \text{bidvol}, \text{size}, \) and \( \epsilon_{\text{cnt}} \) (70x70 bins).
Figure G.14: Trade sign ratio (colour scale axis) for the CBA stock as a function of askvol and bidvol (70x70 bins).

Figure G.15: Trade sign ratio (colour scale axis) for the CBA stock after the SOM transformation as a function of askvol, bidvol, size, and $\epsilon_{cut}$ (70x70 bins).
Appendix G. Statistics for Chapter 4

G.4 Stock code: NAB

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>$\ln(\frac{Si%}{Bi%})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.30</td>
<td>1.55</td>
<td>-0.177</td>
</tr>
<tr>
<td>1</td>
<td>6.41</td>
<td>6.36</td>
<td>0.007</td>
</tr>
<tr>
<td>2</td>
<td>14.13</td>
<td>13.03</td>
<td>0.081</td>
</tr>
<tr>
<td>3</td>
<td>17.19</td>
<td>14.17</td>
<td>0.193</td>
</tr>
<tr>
<td>4</td>
<td>11.35</td>
<td>8.42</td>
<td>0.299</td>
</tr>
<tr>
<td>5</td>
<td>3.82</td>
<td>2.28</td>
<td>0.516</td>
</tr>
</tbody>
</table>

Table G.4: Percentage of the seller-initiated and buyer-initiated trades for the NAB stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.16: Daily normalised trade count (grey scale axis) for the NAB stock as a function of time and trade size (103x70 bins).
Figure G.17: Normalised trade count (grey scale axis) for the NAB stock as a function of \( askvol \) and \( bidvol \) (70x70 bins).

Figure G.18: Normalised trade count (grey scale axis) for the NAB stock after the SOM transformation as a function of \( askvol \), \( bidvol \), \( size \), and \( \epsilon_{\text{cnt}} \) (70x70 bins).
Figure G.19: Trade sign ratio (colour scale axis) for the NAB stock as a function of \textit{askvol} and \textit{bidvol} (70x70 bins).

Figure G.20: Trade sign ratio (colour scale axis) for the NAB stock after the SOM transformation as a function of \textit{askvol}, \textit{bidvol}, \textit{size}, and $\epsilon_{\text{cnt}}$ (70x70 bins).
Appendix G. Statistics for Chapter 4

G.5 Stock code: NCP

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>$\ln(\frac{Si%}{Bi%})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.50</td>
<td>5.23</td>
<td>-0.738</td>
</tr>
<tr>
<td>1</td>
<td>7.80</td>
<td>12.08</td>
<td>-0.437</td>
</tr>
<tr>
<td>2</td>
<td>12.47</td>
<td>15.26</td>
<td>-0.202</td>
</tr>
<tr>
<td>3</td>
<td>12.61</td>
<td>12.63</td>
<td>-0.001</td>
</tr>
<tr>
<td>4</td>
<td>8.36</td>
<td>6.37</td>
<td>0.271</td>
</tr>
<tr>
<td>5</td>
<td>2.95</td>
<td>1.74</td>
<td>0.528</td>
</tr>
</tbody>
</table>

Table G.5: Percentage of the seller-initiated and buyer-initiated trades for the NCP stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.21: Daily normalised trade count (grey scale axis) for the NCP stock as a function of time and trade size (103x70 bins).
Figure G.22: Normalised trade count (grey scale axis) for the NCP stock as a function of askvol and bidvol (70x70 bins).

Figure G.23: Normalised trade count (grey scale axis) for the NCP stock after the SOM transformation as a function of askvol, bidvol, size, and ϵ_{cut} (70x70 bins).
Figure G.24: Trade sign ratio (colour scale axis) for the NCP stock as a function of $askvol$ and $bidvol$ (70x70 bins).

Figure G.25: Trade sign ratio (colour scale axis) for the NCP stock after the SOM transformation as a function of $askvol$, $bidvol$, $size$, and $\epsilon_{cnt}$ (70x70 bins).
Appendix G. Statistics for Chapter 4

G.6 Stock code: RIO

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>$\ln(\frac{\text{Si}-%}{\text{Bi}%})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.49</td>
<td>4.17</td>
<td>-0.514</td>
</tr>
<tr>
<td>1</td>
<td>8.25</td>
<td>11.71</td>
<td>-0.350</td>
</tr>
<tr>
<td>2</td>
<td>12.87</td>
<td>15.86</td>
<td>-0.208</td>
</tr>
<tr>
<td>3</td>
<td>12.96</td>
<td>13.07</td>
<td>-0.008</td>
</tr>
<tr>
<td>4</td>
<td>7.72</td>
<td>6.64</td>
<td>0.150</td>
</tr>
<tr>
<td>5</td>
<td>2.60</td>
<td>1.66</td>
<td>0.450</td>
</tr>
</tbody>
</table>

Table G.6: Percentage of the seller-initiated and buyer-initiated trades for the RIO stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.26: Daily normalised trade count (grey scale axis) for the RIO stock as a function of time and trade size (103x70 bins).
Figure G.27: Normalised trade count (grey scale axis) for the RIO stock as a function of askvol and bidvol (70x70 bins).

Figure G.28: Normalised trade count (grey scale axis) for the RIO stock after the SOM transformation as a function of askvol, bidvol, size, and $\epsilon_{\text{cut}}$ (70x70 bins).
Figure G.29: Trade sign ratio (colour scale axis) for the RIO stock as a function of askvol and bidvol (70x70 bins).

Figure G.30: Trade sign ratio (colour scale axis) for the RIO stock after the SOM transformation as a function of askvol, bidvol, size, and $\epsilon_{\text{cut}}$ (70x70 bins).
G.7 Stock code: TLS

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>Ln(\text{Si%}/\text{Bi%})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.22</td>
<td>6.90</td>
<td>-1.132</td>
</tr>
<tr>
<td>1</td>
<td>5.71</td>
<td>10.05</td>
<td>-0.564</td>
</tr>
<tr>
<td>2</td>
<td>9.86</td>
<td>10.72</td>
<td>-0.084</td>
</tr>
<tr>
<td>3</td>
<td>13.17</td>
<td>10.10</td>
<td>0.266</td>
</tr>
<tr>
<td>4</td>
<td>13.23</td>
<td>7.08</td>
<td>0.626</td>
</tr>
<tr>
<td>5</td>
<td>7.89</td>
<td>3.07</td>
<td>0.945</td>
</tr>
</tbody>
</table>

Table G.7: Percentage of the seller-initiated and buyer-initiated trades for the TLS stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.31: Daily normalised trade count (grey scale axis) for the TLS stock as a function of time and trade size (103x70 bins).
Appendix G. Statistics for Chapter 4

Figure G.32: Normalised trade count (grey scale axis) for the TLS stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.33: Normalised trade count (grey scale axis) for the TLS stock after the SOM transformation as a function of \( \text{askvol} \), \( \text{bidvol} \), \( \text{size} \), and \( \epsilon_{\text{cut}} \) (70x70 bins).
Figure G.34: Trade sign ratio (colour scale axis) for the TLS stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.35: Trade sign ratio (colour scale axis) for the TLS stock after the SOM transformation as a function of \( \text{askvol}, \text{bidvol}, \text{size}, \) and \( \epsilon_{\text{cut}} \) (70x70 bins).
G.8 Stock code: WBC

<table>
<thead>
<tr>
<th>ε_{cnt}</th>
<th>Si%</th>
<th>Bi%</th>
<th>Ln(Si/Bi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.76</td>
<td>4.96</td>
<td>-0.586</td>
</tr>
<tr>
<td>1</td>
<td>8.33</td>
<td>12.51</td>
<td>-0.407</td>
</tr>
<tr>
<td>2</td>
<td>13.14</td>
<td>17.08</td>
<td>-0.263</td>
</tr>
<tr>
<td>3</td>
<td>12.01</td>
<td>13.11</td>
<td>-0.087</td>
</tr>
<tr>
<td>4</td>
<td>6.62</td>
<td>6.19</td>
<td>0.067</td>
</tr>
<tr>
<td>5</td>
<td>1.89</td>
<td>1.41</td>
<td>0.297</td>
</tr>
</tbody>
</table>

Table G.8: Percentage of the seller-initiated and buyer-initiated trades for the WBC stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.36: Daily normalised trade count (grey scale axis) for the WBC stock as a function of time and trade size (103x70 bins).
Figure G.37: Normalised trade count (grey scale axis) for the WBC stock as a function of \textit{askvol} and \textit{bidvol} (70x70 bins).

Figure G.38: Normalised trade count (grey scale axis) for the WBC stock after the SOM transformation as a function of \textit{askvol}, \textit{bidvol}, \textit{size}, and \textit{\epsilon_{cnt}} (70x70 bins).
Appendix G. Statistics for Chapter 4

Figure G.39: Trade sign ratio (colour scale axis) for the WBC stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.40: Trade sign ratio (colour scale axis) for the WBC stock after the SOM transformation as a function of \( \text{askvol}, \text{bidvol}, \text{size}, \) and \( \epsilon_{\text{cut}} \) (70x70 bins).
G.9 Stock code: WOW

<table>
<thead>
<tr>
<th>$\epsilon_{cnt}$</th>
<th>Si%</th>
<th>Bi%</th>
<th>$\ln(\frac{Si}{Bi})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.91</td>
<td>2.81</td>
<td>-0.384</td>
</tr>
<tr>
<td>1</td>
<td>7.50</td>
<td>8.99</td>
<td>-0.181</td>
</tr>
<tr>
<td>2</td>
<td>13.41</td>
<td>14.37</td>
<td>-0.069</td>
</tr>
<tr>
<td>3</td>
<td>14.52</td>
<td>13.48</td>
<td>0.074</td>
</tr>
<tr>
<td>4</td>
<td>9.96</td>
<td>7.42</td>
<td>0.294</td>
</tr>
<tr>
<td>5</td>
<td>3.40</td>
<td>2.22</td>
<td>0.425</td>
</tr>
</tbody>
</table>

Table G.9: Percentage of the seller-initiated and buyer-initiated trades for the WOW stock after observing $\epsilon_{cnt}$ seller-initiated trades in the previous five trades (Si-Seller-initiated, Bi-Buyer-initiated).

Figure G.41: Daily normalised trade count (grey scale axis) for the WOW stock as a function of time and trade size (103x70 bins).
Figure G.42: Normalised trade count (grey scale axis) for the WOW stock as a function of $askvol$ and $bidvol$ (70x70 bins).

Figure G.43: Normalised trade count (grey scale axis) for the WOW stock after the SOM transformation as a function of $askvol$, $bidvol$, size, and $c_{cnt}$ (70x70 bins).
Figure G.44: Trade sign ratio (colour scale axis) for the WOW stock as a function of \( \text{askvol} \) and \( \text{bidvol} \) (70x70 bins).

Figure G.45: Trade sign ratio (colour scale axis) for the WOW stock after the SOM transformation as a function of \( \text{askvol}, \text{bidvol}, \text{size}, \) and \( \epsilon_{\text{cut}} \) (70x70 bins).
Appendix H

Sample data for Chapter 5

<table>
<thead>
<tr>
<th>No</th>
<th>$s_0$</th>
<th>$\epsilon_0$</th>
<th>$a^1_0$</th>
<th>$b^1_0$</th>
<th>$a^2_0$</th>
<th>$b^2_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2500</td>
<td>1</td>
<td>7027</td>
<td>2507</td>
<td>2584</td>
<td>2289</td>
</tr>
<tr>
<td>2</td>
<td>260</td>
<td>-1</td>
<td>4527</td>
<td>2500</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>3</td>
<td>2500</td>
<td>1</td>
<td>4527</td>
<td>2240</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>4</td>
<td>125</td>
<td>-1</td>
<td>2027</td>
<td>2240</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>5</td>
<td>125</td>
<td>1</td>
<td>3277</td>
<td>2865</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>6</td>
<td>450</td>
<td>1</td>
<td>3152</td>
<td>2865</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>7</td>
<td>2500</td>
<td>1</td>
<td>2702</td>
<td>2865</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>8</td>
<td>1069</td>
<td>1</td>
<td>1723</td>
<td>2865</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>9</td>
<td>654</td>
<td>1</td>
<td>654</td>
<td>2865</td>
<td>2584</td>
<td>2507</td>
</tr>
<tr>
<td>10</td>
<td>635</td>
<td>-1</td>
<td>2584</td>
<td>986</td>
<td>5223</td>
<td>2865</td>
</tr>
<tr>
<td>11</td>
<td>2574</td>
<td>1</td>
<td>2584</td>
<td>1601</td>
<td>5223</td>
<td>2865</td>
</tr>
<tr>
<td>12</td>
<td>178</td>
<td>-1</td>
<td>10</td>
<td>1601</td>
<td>5223</td>
<td>2865</td>
</tr>
<tr>
<td>13</td>
<td>250</td>
<td>-1</td>
<td>10</td>
<td>1424</td>
<td>5223</td>
<td>2865</td>
</tr>
<tr>
<td>14</td>
<td>500</td>
<td>-1</td>
<td>10</td>
<td>6129</td>
<td>5223</td>
<td>865</td>
</tr>
<tr>
<td>15</td>
<td>10</td>
<td>1</td>
<td>10</td>
<td>5629</td>
<td>5223</td>
<td>865</td>
</tr>
<tr>
<td>16</td>
<td>5000</td>
<td>1</td>
<td>5223</td>
<td>165</td>
<td>4645</td>
<td>5629</td>
</tr>
<tr>
<td>17</td>
<td>223</td>
<td>1</td>
<td>223</td>
<td>165</td>
<td>4645</td>
<td>5629</td>
</tr>
<tr>
<td>18</td>
<td>1418</td>
<td>-1</td>
<td>4645</td>
<td>1418</td>
<td>1698</td>
<td>165</td>
</tr>
<tr>
<td>19</td>
<td>165</td>
<td>-1</td>
<td>2333</td>
<td>165</td>
<td>4645</td>
<td>3989</td>
</tr>
<tr>
<td>20</td>
<td>397</td>
<td>1</td>
<td>397</td>
<td>3989</td>
<td>6333</td>
<td>865</td>
</tr>
</tbody>
</table>

Table H.1: A sample sequence of trade and order book data for a selected stock. The sequence represents consecutive aggregated trades observed on a single day. Order book data were captured just before a trade was executed. Except for the trade sign, the data were scrambled to protect the identity of the stock while preserving volume relationships.$^1$

$^1$Headings: $s_0$ - trade size; $\epsilon_0$ - trade sign (1 - buyer-initiated; -1 - seller-initiated); $a^1_0$ - total volume at the best ask; $b^1_0$ - total volume at the best bid; $a^2_0$ - total volume at the second best ask; $b^2_0$ - total volume at the second best bid. All variables are contemporaneous.
Appendix I

The Bonferroni adjustment

This section is a based on the material presented in Ref. [184]. A difference between two means, \( \mu_i \) and \( \mu_j \), can be tested with a standard t-test. Typically, two hypotheses, a null hypothesis (\( H_0 \)) and an alternative hypothesis (\( H_\alpha \)), are made:

\[
H_0 : \mu_i = \mu_j \\
H_\alpha : \mu_i \neq \mu_j
\]

The two means are considered different if the null hypothesis can be rejected at some nominated significance level \( \alpha \), usually taken as 0.05 or 0.01. Rejecting \( H_0 \) when the two means are in fact equal is referred to as a Type I error. The alternative case, rejecting \( H_\alpha \) when the two means are in fact different, is called a Type II error. Rejecting the null hypothesis at the significance level \( \alpha \) implies that the probability of Type I error is \( \alpha \), while the probability of Type II error is \( 1 - \alpha \).

A group of \( n \) means allows up to \( \frac{k(k-1)}{2} \) different comparisons between them. If each individual comparison is conducted at the significance level of \( \alpha \) then the probability of at least one Type I error in the whole group of \( c \) tests is equal to \( P_G = 1 - (1 - \alpha)^c \). The group significance level \( P_G \) is thus greater than the individual significance level \( \alpha \) and can be set arbitrarily close to 1 by
tuning the value of \( c \). This means that, in practice, it is almost always possible to incorrectly reject the null hypothesis for a group of means by conducting a high enough number of comparisons. The Bonferroni adjustment is used to put a hard limit on the growth of \( P_G \). It can be shown than for a small \( \alpha \) and a large enough \( c \) the group significance level \( P_G \approx \alpha \) if individual tests are conducted at the significance level of \( \alpha/c \). This is a conservative approximation and there exist many more accurate multiple comparison methods. The simplicity of the Bonferroni adjustment, however, makes it an attractive technique for exploratory analysis.
Glossary

ask
A sell order

ASX
Australian Stock Exchange

best ask
An ask with the lowest price in the limit order book

best bid
A bid with the highest price in the limit order book

bid
A buy order

buyer-initiated trade
A trade triggered by a buy order

contemporaneous variable
A non-lagged (current, present) variable

k-NN
K-nearest-neighbour
limit order

An order to buy or sell shares with a specified price. It may be traded (executed) immediately if the specified price produces a non-positive spread.

market order

An order to buy or sell shares without a specified price. It may be traded (executed) immediately if there is a matching volume on the opposite side of the market.

NYSE

New York Stock Exchange

seller-initiated trade

A trade triggered by a sell order

SOM

Self-organising map

spread

A price difference between the best ask and the best bid

VWAP

Volume weighted average price
Bibliography


