Institute of Technology Blanchardstown

Predictions in Financial Time Series Data

Author:
Dr. Allan Steel

Supervisor:
Dr. Geraldine Gray

A thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Computing

from the

School of Informatics and Engineering

August 2014

Declaration of Authorship

I, Dr. Allan Steel, declare that this thesis titled, 'Predictions in Financial Time Series Data' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

iate

INSTITUTE OF TECHNOLOGY BLANCHARDSTOWN

Abstract

School of Informatics and Engineering

Master of Science in Computing

Predictions in Financial Time Series Data

by Dr. Allan Steel

For hundreds of years speculators have tried to make a profit from the financial markets by predicting their future movements. To this end, many methods and techniques have been developed that purport to assist the market participant in generating profits. This study reports on the effectiveness of several such techniques to help predict the one step ahead forecasts of a range of national stock market indices (DAX, FTSE, CAC, Dow Jones, Nikkei and AORD) . A selection of technical analysis indicators as well as more traditional time series prediction models such as exponential smoothing, ARIMA and hybrid ARIMA techniques are explored. Results from the predictive models are passed to trading algorithms where purchasing decisions are directed by the forecasts. Profitable systems are developed based on predictions from technical analysis and hybrid ARIMA models using a k-NN learner.

Acknowledgements

I wish to thank my supervisor Dr Geraldine Grey for her guidance, support and patience throughout the completion of this project. I would also like to acknowledge Dr Markus Hofmann and Laura Keyes, who along with Geraldine provided two years of stimulating and enjoyable lectures.

This study made extensive use of open source software and I would like to thank and acknowledge all the people who provided their time, skill and expertise to make world class software freely available.

Finally, I would like to thank my partner Ann for all her patience and help during the writing of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

For hundreds of years speculators have tried to make a monetary profit in financial markets by predicting the future price of commodities, stocks, foreign exchange rates and more recently futures and options. Over the last few decades these efforts have increased markedly, using a variety of techniques (Hsu, 2011), which can be broadly classified into three categories:

- fundamental analysis
- technical analysis
- traditional time series forecasting

1.1.1 Fundamental Analysis

Fundamental analysis makes use of basic market information in order to predict future movements of an asset. If an investor was looking at a particular stock's fundamental data they would consider information such as revenue, profit forecasts, supply, demand and operating margins etc. Speculators looking at commodities might consider weather patterns, political aspects, government legislation and so on. Effectively fundamental analysis is concerned with macro economic and political factors that might affect the future price of a financial asset. Fundamental analysis is not considered further in this study.

1.1.2 Technical Analysis

Technical analysis is the study of historical prices and patterns with the aim of predicting future prices. Practitioners of technical analysis in the past were referred to as chartists, as they believed all that was needed to know about a particular market was contained in its pricing chart. Murphy (1999) defines technical analysis as:

"Technical analysis is the study of market action, primarily through the use of charts for the purpose of forecasting future price trends."

Technical analysis (TA) is interesting as it tends to polarise opinion as to its scientific basis and effectiveness. To many people and particularly scholars in academia it is considered little more than Black Magic. Consider the words of Malkiel (1999):

"Obviously I am biased against the chartist. This is not only a personal predilection, but a professional one as well. Technical Analysis is anathema to the academic world. We love to pick on it. Our bullying tactics are prompted by two considerations: (1) the method is patently false; and (2) it's easy to pick on. And while it may seem a bit unfair to pick on such a sorry target, just remember: it is your money we are trying to save."

However, in the world of finance technical analysis is ubiquitous and widely used (Menkhoff, 2010). In support of TA a plethora of so-called indicators have been developed over the years from simple moving averages to much more exotic offerings. Today every piece of software or on-line analysis tool provides the ability to place a multitude of technical indicators on a graph of a stock, commodity or any financial instrument.

Most technical indicators essentially fall into one of two main categories, ones attempting to detect the start and direction of trends and those trying to identify market reversals generally called oscillators. Trend analysis indicators include Average Direction Index (ADX), Aroon, Moving Averages and Commodity Channel Indexes (CCI). Price oscillator indicators include, Moving Average Convergence Divergence (MACD - (Appel and Dobson, 2007)), Stochastics, Relative Strength Index (RSI) and the Chande Momentum Oscillator (CMO).

1.1.3 Time Series Forecasting

The study of forecasting time series data has been an active area of study for several decades (De Gooijer and Hyndman, 2006). Series data is ordered such that the ordering is an important if not critical aspect of the data, with the requirement to maintain this ordering enforcing certain requirements on any processing. Series data can be ordered by factors such as distance or height but typically time is the ordering encountered. Financial data is an important category of series data and a variety of well known time series forecasting methods have been applied to the problem of predicting price movements in the financial markets. These have included, exponential smoothing, auto-regressive moving average (ARMA) and auto-regressive integrated moving average (ARIMA).

A variety of smoothing algorithms have been applied to series data in general and financial data in particular. Moving averages, including simple, weighted and exponential, are widely employed by participants in financial markets to both predict future movements and quantify current conditions. Classical time series analysis such as so-called Holt-Winters exponential smoothing, the auto-regressive moving average (ARMA or Box-Jenkins model) and auto-regressive integrated moving average (ARIMA) methods have also been widely employed. In more recent years data mining techniques have been applied to the problem of financial time series prediction, for example with the use of artificial neural networks (ANNs) and support vector machines (SVM) as well as an hybrid approach of combining the classic time series techniques with the data mining methods in an attempt to leverage the strengths of each technique.

1.2 Statement of the Problem

The problem under study in this thesis is that of predicting the movement of financial markets. Financial markets include:

- Indices e.g. Dow Jones Index, FTSE 100 etc.
- Commodities e.g. gold, oil etc.
- Foreign exchange rates (also known as Forex or FX) e.g. GBP USD (price of British pounds divided by US dollars).
- Stocks e.g. Google, Apple, Barclays Bank etc.

The goal of financial traders is to detect the movement of the markets and buy instruments expected to rise in price "going long" and sell those predicted to fall in price "going short". The markets are a neutral sum process, for every participant who gains there are those who lose.

1.3 Purpose of Study

The purpose of this study is to investigate and establish the usefulness and accuracy of a selection of technical indicators and time series analysis on the ability to predict future data movements in a group of national indice data sets.

1.3.1 Study Objectives

The objective of this study is three fold:

1. To investigate a group of national indice data sets in terms of their behaviour and characteristics.
2. To determine if a group of popular and widely used technical indicators can be used to predict the direction of movement in a range of financial markets.
3. To investigate if traditional time series models can predict the direction of movement in a range of financial markets.

1.4 Research Question

The research question addressed in this study is:

"Can the use of technical indicators or time series analysis help to predict the future direction and movement of financial markets?"

1.5 Methodology

The following methodology was used to answer the research question:

1. The current research in the field was reviewed.
2. Appropriate data was collected, primarily from freely available sources on the internet such as Yahoo and Google.
3. Initial data investigations and visualisations were carried out on the data.
4. Based on initial analysis, "base line" systems were established that could be used to compare the performance of systems generated by technical analysis and time series modelling.

5. A number of trading algorithms were generated that consumed the output of technical analysis indicators to determine in which direction to trade the financial data at any one time.
6. Times series modelling methods were used to generate forecasts for the financial data, and these were used in trading algorithms to decide in which direction, long or short, to trade.

In summary, the output of technical and times series analysis was consumed in a range trading algorithms. The decisions regarding which direction to trade a particular stock market was based on the predictions or output of the analysis. Success was measured in terms of whether the trading systems developed can profitably predict which way to trade the financial markets.

1.6 Limitations of the Study

Limitations in this study include:

1. Choice of technical indicators - a small selection of the huge number available was selected. The selected group represent widely used examples and are drawn from the various categories available.
2. Use of financial data relating to stock market indices - daily data in the form of open, high, low and close prices (OHLC) from national stock market indices such as the Dow Jones or FTSE 100 is readily and freely available and thus was used in this study. High quality data in time frames other than daily or from alternative financial markets such commodities or foreign exchange is generally only commercially available and beyond the resources of this study.

1.7 Scope of the Study

There is a huge choice of financial data sets from which to choose and likewise many dozens of technical indicators. Given the time frame and resources available, this study employed daily data from major national indices such as the German DAX, US Dow and Japanese Nikkei. Technical indicators selected included examples from the primary categories such as trend detection and market reversal indicators.

1.8 Structure of Project

Chapter 2 is a literature review and introduction to financial market trading, the methods and theory of technical analysis and time series modelling. Financial trading systems in general are discussed along with the use and applicability of technical analysis. The classical time series methods of Holt-Winters exponential smoothing, auto-regressive moving average (ARMA or Box-Jenkins model) and auto-regressive integrated moving average (ARIMA) are introduced and explained along with more recent developments such as hybrid ARIMA models.

Chapter 3 introduces the methodology used in this study. It includes a description of the data sets employed, software and programming languages utilised and the general approach taken. Chapter 4 details the experiments carried out using a variety of technical analysis indicators and lists the results from the trading algorithms generated. Chapter 5 documents the experimental work based on the use of time series modelling to generate forecasts for the financial data sets.

Chapter 6 is an analysis of the results obtained in Chapters 4 and 5 along with conclusions and suggestions for future work. Appendix A lists all the R programming code used in study. Wherever possible this report has the analysis generated by R programming code embedded into it. Thus, all trading algorithms coded in R detailed in Chapters 4 and 5 generate results that are dynamically embedded into this report. An update or alteration of this code followed by recompilation of this manuscript updates the tables and results accordingly.

Appendix B provides additional details and background information on various technical analysis indicators. Finally, Appendix C presents all the results generated in Chapters 4 and 5 collected together by the particular financial market.

Chapter 2

Literature Review

Speculators, stock market traders, market participants or simply traders are all terms used to describe individuals and organisations who attempt to make a living from buying and selling various financial assets in a huge range of markets around the world. Clearly the ability to forecast the direction of market movements, up or down, is vital to these individuals and entities. To this end a wide variety of techniques and methods have been tried and used by the participants in the market. Further, over the last few decades academics have shown an interest in this field and attempted to quantify and justify the wide variety of techniques used.

Two areas where traders and academics have looked for help in predicting future market direction is time series forecasting and the use of technical indicators. This chapter is divided into two these general categories, time series modelling and the use of technical indicators.

2.1 Technical Analysis

2.1.1 Trading Systems

A wide variety of techniques have been employed by financial market traders in their attempts to make profits with the term "trading system" being applied generally to the methodology used. Often trading systems are "mechanical" in nature in that traders use a distinct set of rules in order to guide them as when to enter a trade, when to exit and so on. Faith (2007), one of the original and now famous "Turtle Traders" provides an excellent overview of mechanical trading systems (and how they were to become known as the "Turtles").

Weissman (2005) makes the point that there are several aspects to a trading system. Firstly there are entry and exit signals, which are market events that trigger a speculator to enter into the market and either buy or sell a particular asset. These signals are typically events such as a fast moving average crossing a slower one, the market hitting a certain price or the occurrence of a particular chart pattern (see section 2.1.5). Other elements of a trading system include position sizing rules and money management strategies such that returns are significant, losses are minimised and the entire risk profile is controlled.

Many traders erroneously mistake entry and exit signals as being a full trading system in themselves whereas in actuality they are merely components of a system (Beau and Lucas, 1999). Likewise most, if not all, papers published by academia focus on entry and exit signals alone, which is probably a result of several factors. Firstly, entry and exit signals are important components in trading systems and are a good place to start in system development. Additionally, the other aspects of a system are not as well known and their importance is often ignored (Kaufman, 2013). Finally, testing an "entire" system as defined here is far more difficult and time consuming than considering entry and exit signals alone and often it is not practical to extend a study to include a full system. In summary there is value in considering entry and exit signal in isolation but one has to remember it is not the whole story.

Attempting to forecast stock market prices is a complex and challenging endeavour, yet one that is widely encountered. There is a large body of research published in this area which has been reviewed by Atsalakis and Valavanis (2009). Work usually focuses on either individual stocks or more commonly stock indices. Stock indices are the sum movements of many individual equities and therefore reflect the movement of the market as a whole as opposed to any one stock. Many stock market indices have been investigated including those belonging to well-developed countries such as those in Western Europe, North America etc. as well as developing markets such as those in Eastern Europe.

In trying to predict stock market movements a variety of input variables have been used. Frequently, the so-called OHLC (open, high, low and closing prices) are used as inputs along with a variety of technical indicators (Fiess and MacDonald, 2002). In addition, many authors have used a combination of markets, for example Huang et al. (2005) use both the USD/YEN exchange rate and the S&P 500 to build a prediction model for the Japanese NIKKEI index. A variety of predictive methodologies have been reported in the literature including linear and multi-linear regression, ARMA and ARIMA models, genetic algorithms (GA), artificial neural networks (ANN), random walk (RW) and the so-called buy and hold (B & H) strategy.

A variety of performance measures have been reported including both non-statistical and statistical methods. Non-statistical performance measures encountered include annual return and annual profit of a particular model as well as the hit rate or the number of times a model correctly predicts whether a market will go up or down. Alternatively a variety of statistical measures have also been employed and prominent amongst them are, mean absolute error (MAE), root mean squared (RMSE), mean squared prediction error (MSPE), correlation coefficient and autocorrelation squared correlation and Akaike's minimum final prediction error (FPE).

Two well studied and used methodologies in stock trading are the moving average system and range breakout system as reported by (Brock et al., 1992) in one of the very earliest papers published covering technical analysis. In a moving average system (see section 4.3.1) the speculator buys into a market when its price is above the moving average and sells in the reverse situation. A large number of variations on this theme can be found, with the use of two moving averages being popular. When using two averages there is normally a "fast" one, usually of the order of 10 to 25 days, and a "slow" one in the 50 to 250 day range. In these circumstances a buy is usually triggered when the fast average crosses above the slower average. The theory is that the moving averages follow the trends in the market and thus allow the market participant to trade in the direction of the trend, which is an advantageous situation for the trader.

A second popular idea is that of breaking out of a range. Often financial markets trade between a range of values in a particular time period, essentially markets are either trending (up or down) or not trending at all but moving within a defined range. While moving in a range the lower price boundary is referred to as support and the upper one as resistance. In a breakout system the analyst buys a market when it moves beyond these resistance levels or sells when it breaks below the support. Brock et al. (1992) analysed both these two ideas and found merit in them. Using daily data from the Dow Jones industrial index they found that these strategies provided better results than those generated with random walk, AR and GARCH models.

2.1.2 Technical Analysis Overview

Technical analysis is the technique of looking at the past history of a financial market, identifying patterns and trends and utilising the information in predicting future price movements (Bulkowski, 2011). A technical indicator is a method used to identify a particular pattern, and there have been a large number developed over the years to predict situations such as the start of a trend or a reversal in price movement. A wide range of papers on technical analysis (TA) indicators and methods can be found in the literature. Likewise technical analysis is prominent in many best selling books including Market Wizards (Schwager, 1988), New Market Wizards (Schwager, 1994) and Covel's Trend Following (Covel, 2009). In the following sections various technical indicators are introduced and their use in predicting market movements are explored. Firstly, the question of whether technical analysis even works is addressed, as this has received attention in the literature (Marshall and Cahan, 2005, Reitz, 2006, Schulmeister, 2009, Marshall et al., 2008). Although technical analysis is widely used in the market place there is a question mark over the entire concept behind it and many people, especially academics, are highly sceptical about the validity of the entire approach.

2.1.3 Does Technical Analysis Work?

Friesen et al. (2009) have examined various price "patterns" used by traders in their systems such as "head-and-shoulders" and "double-top" patterns. The authors note that although a wide array of patterns have been identified and documented there lacks any convincing explanations for the formation of these patterns and how they can lead to profitable trading systems. The authors report that several studies based on the US equity market have identified distinct behaviours, namely the tendency for short-term momentum over 1 year to 6 months (De Bondt and Thaler, 1985, Chopra et al., 1992, Jegadeesh and Titman, 1993), longer term mean reversion and finally price reversals over the one to four week period (Jegadeesh, 1990, Lehmann, 1990, Jegadeesh and Titman, 1995, Gutierrez Jr and Kelley, 2008). These observations lend support to the success of trading systems that purport to detect and follow trends in the market (Sweeney, 1986, Levich and Thomas, 1993, Neely et al., 1997, Dueker and Neely, 2007).

The authors present a model that can explain the profitability of selected trading rules that utilise past chart patterns. One important aspect of this model is the inclusion of confirmation bias, which shows up in a wide range of decision making processes. Their model displays negative autocorrelations over the very short term, positive ones in the mid term and become negative again over the longer horizon, reflecting the documented empirical properties of US stock prices (Novy-Marx, 2012, Moskowitz et al., 2012, Fama and French, 2012). It is suggested that traders take market positions affected by their original biased view which leads to autocorrelations and price movement patterns resulting in the previously described market behaviour.

Shynkevich (2012) investigated the power of a large selection of technical trading rules to yield profits when applied a selection of small cap and technology portfolios (US stocks) between 1995 and 2010. The author chose technical indicators from four general categories:

1. standard filter rules - for example a buy is generated when prices increase from a previous low. Such a low may be defined as the lowest closing price in a particular period. In more recent years this technique has been replaced by moving averages.
2. moving averages (MA) - signals generated when short MA cross long MA.
3. support and resistance trading strategy (SR) - a buy is initiative when prices rise above a local maximum, and vice versa for a local minimum price.
4. channel breakout - related to SR, a buy/sell is triggered when a price moves outside a channel generated from highs and lows of a certain period.

The author applied a variety of parameters in each model resulting in a total of 12937 models being tested. It was reported that TA produced positive results in the first half of the time period tested, but not in the latter half. In the second half of the time period studied TA provided inferior performance than a buy-and-hold approach, i.e. a trader simply buys a particular asset and waits. The author concludes these differences in performance are due to equity markets having become more efficient in recent years which has reduced the short term predictive powers of TA.

The use of technical analysis in the finance community was studied by Menkhoff (2010) who looked into its use by professional fund managers. This study is note worthy as it used data from experienced and educated market professionals and not a wider cross-section of traders. With the advent of the internet and the explosive growth in on-line financial charting and trading sites, financial trading became accessible to the general public, resulting in huge numbers of amateur traders entering the market. All of the web sites that cater for this segment of traders offer a huge number of technical analysis indicators built into their respective charting packages and even a rudimentary visit to any of the discussion forums will demonstrate the popularity and wide spread use of technical analysis.

The author surveyed 692 fund managers in several countries, with funds of various sizes under management. The vast majority of these fund managers reported using technical analysis to some degree and particular faith was put in TA for predicting price movements in the short term of up to a few weeks, beyond which focus shifts to fundamental analysis. Further, the workers found that smaller asset manager firms make greater use of TA, possibly because deriving the information for fundamental analysis is beyond their resources. Finally, most respondents to the survey believe that human psychology is the reason TA works. In particular they suggest psychological biases in the market participants are the root cause of market trends and that TA is able to identify and follow them.

2.1.4 Moving Average Indicators

A study of moving average convergence divergence (MACD) is reported by Ulku and Prodan (2013). MACD is a technique which attempts to detect the early stage of a trend as it forms, and is widely used by market participants. It is described in more detail in Appendix B section B.1. Ulku and Prodan (2013) apply MACD to a wide range of national stock market indices comprising developed as well as emerging markets. The authors compare the MACD signals against entry signals generated from simple break out systems (described previously). The comparison systems would generate a buy signal if the price moved higher than a moving average (MA), set at either 22, 56 and 200 days. The MACD and the comparison system using 22 day moving averages are classified as short horizon signals, while the break out of the 56 and 200 day MA are considered long horizon signals. The workers reported that the MACD indicators provide for profitable returns on 23 of 30 national indices, but that the 22 day MA performs better being positive in 27 of the 30 markets.

2.1.5 Candlesticks Patterns

Probably the oldest form of technical analysis in use today is the so-called candlestick analysis, so named because daily open and close prices are plotted such that they resemble candlesticks (Morris, 2006). Figure 2.1 is an example of daily prices being plotted as a candlestick, with this plotting methodology being ubiquitous today in trading software. Typically the colour in which the candlestick is plotted indicates whether the price went up or down over the course of the day. Many charts that are plotted in colour use green to represent days that close up and red for days that close down. The main body of the candlestick represents the movement from open to close, and the protruding lines mark the high and low of the day.

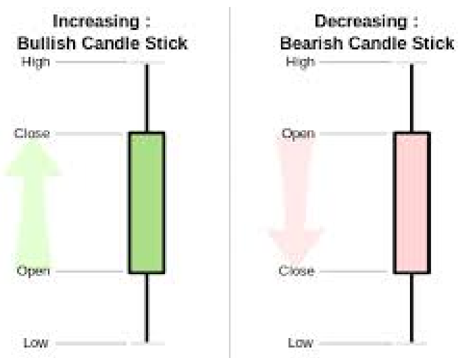

Figure 2.1: Candlestick representation of daily open and close prices. Different colouring is used to distinguish between prices going up or down.

Technical analysis via candlesticks is reputed to have been developed by Munelusa Homma, a legendary trader of rice in Osaka, Japan who made a fortune analysing rice prices with candlesticks in the seventeenth century (Nison, 2001). Candlestick patterns with supposed predictive qualities can be derived from a single day or from considering a few days, usually 2 or 3, together (Bigalow, 2011). There are a huge number of patterns recorded in the literature and usually assigned exotic names such as “White Marubozu”, ”Black Shooting Star” and ”Hanging Man”. Examples of such named patterns can be seen in Figure 2.2.

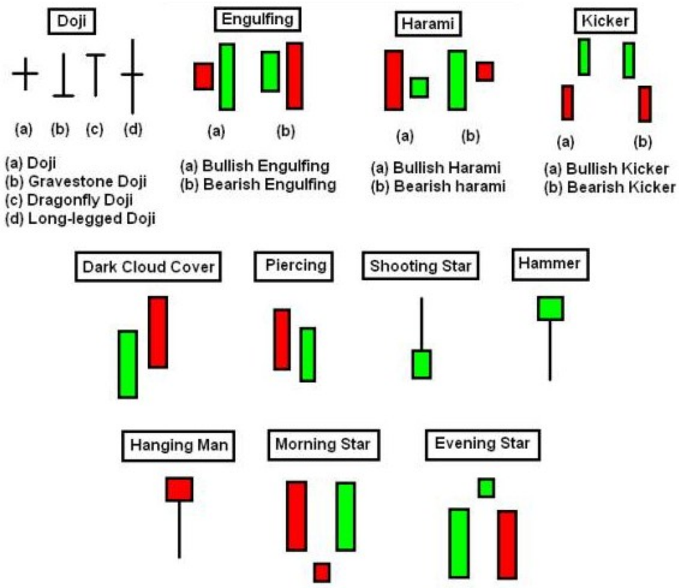

FIGURE 2.2: Examples of well known patterns encountered in candlestick analysis.

Candlestick patterns are essentially visualisation tools providing an easy to comprehend view of the market movements in a particular day. However there is some vital information which is not conveyed in a candlestick. In particular the order of events isn’t displayed. Figure 2.3 shows how two days can produce the same candlestick but in actuality the price movements and volatility in them was very different. Depending upon the type of trading system being employed this could have important effects.

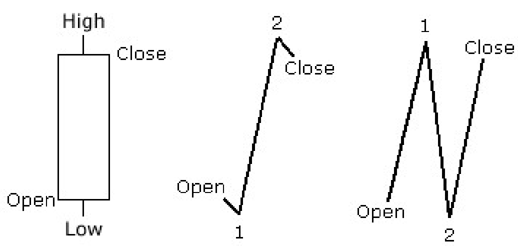

FIGURE 2.3: Candlesticks don’t provide information regarding the order of price movements. Both these daily price movements would be represented with the same candlestick pattern.

As always with technical analysis there is doubt as to the validity of the methods despite its almost universal employment. An in-depth study of the predictive power of a range of candlestick patterns on stock prices between 1992 and 2002 from the Dow Jones Industrial Average (DJIA) was carried out by (Marshall et al., 2006) in which doubt was cast on the validity of candlestick patterns to predict market movements. The workers used a range of bullish (signals that indicate a trader should buy) and bearish (signals that indicate a trader should sell) candlestick patterns to initiate trades on the various stocks. Trades were held for ten days as it was assumed that these patterns reflect short terms trends and thus have a predictive power in a similar time frame. In order to quantify the results generated from the use candlestick patterns they were compared to results observed from four alternative null models. Simulated stock data was generated using a bootstrapping methodology (Efron, 1979) and then four null models were applied to the data, random walk, an autoregressive process of order one (AR(1)), a GARCH in-Mean (GARCH-M) model and an Exponential GARCH (EGARCH) model.

From the comparison of the results generated from the candlestick patterns and the four null models the workers concluded that the variety of candlestick patterns tested had no predictive power on the stocks at all. The returns from making buying and selling decisions based on candlestick patterns didn't outperform the null models on the simulated data. As always one has to be slightly careful with results of this nature as the trading period was fixed at ten days, in other words the candlestick patterns were used as an entry signal for the trade but there wasn't an exit signal. Further in reality use of candlesticks analysis would be incorporated into a trading system, which typically consists entry and exit signal, position sizing rules and money management strategies (Faith, 2007).

2.1.6 Trend Reversal Oscillators

Tanaka-Yamawaki and Tokuoka (2007) reported the use of several technical analysis techniques in the successful prediction of price movements in eight stocks found on the New York Stock Exchange (NYSE) by analysing tick data. The predictions were in the very short term as tick data is the most granular level reported in financial data. The workers used ten technical analysis indicators from three broad classes, namely trend indicators, oscillators to find market reversals and momentum indicators to measure the strength of the market. Combinations of indicators, typically from the different categories are usually combined by market participants into a variety of systems. In this study the ten indicators can form a possible 1023 combinations. A genetic algorithm was used to determine the best combination of indicators for each stock, resulting in a customised combination for each. Using each stock's indicators, the next ten ticks of data were modelled with very high accuracy, with predictions for IBM's stock being the best at a very impressive 82%.

2.2 Time Series Analysis

The study of forecasting time series data has been an active area of study for several decades and an overview of work over 25 years has been documented by De Gooijer and Hyndman (2006). Series data is ordered such that the ordering is an important if not critical aspect of the data and the requirement to maintain this ordering enforces certain constraints on its processing. Series data can be ordered by factors such as distance or height but typically time is the ordering encountered, and thus such collections are referred to as time series. Analysis of time series data is found in a wide range of areas including, Sales Forecasting, Speech Recognition, Economic Forecasting, Stock Market Analysis, Process and Quality Control and Seismic Recordings.

In general with non-series data we are interested in the relationships between the attributes of any particular row of data and perhaps how they affect the parameter we are interested in. Frequently some kind of regression technique is used in this kind of analysis in order to answer questions such as how is rainfall in an area affected by altitude or how does fuel consumption vary with car engine size (Han et al., 2011).

However with time series data there is an additional consideration, the relationship between the attribute's current value to that of its previous or later values. This is known as auto-correlation (Mills, 2011) and more details can be seen in section 2.2.2.1. Typically with financial data we are interested in previous values, in other words how is today's price of a security affected by the price one, two or three days ago?

As illustrated in Figure 2.4 a time series can contain some or all of the following components:

1. Trend - the overall direction of the series, is it increasing or decreasing over time?
2. Seasonality - regular variations in the time series that is caused by re-occurring events, for example a spike in sales during the Christmas period (So and Chung, 2014).
3. Random component - additional fluctuations in the series that may be attributed to noise or other random events.

There are three primary types of time series, stationary, additive and multiplicative. Stationary series have constant amplitude without a trend element and an example can

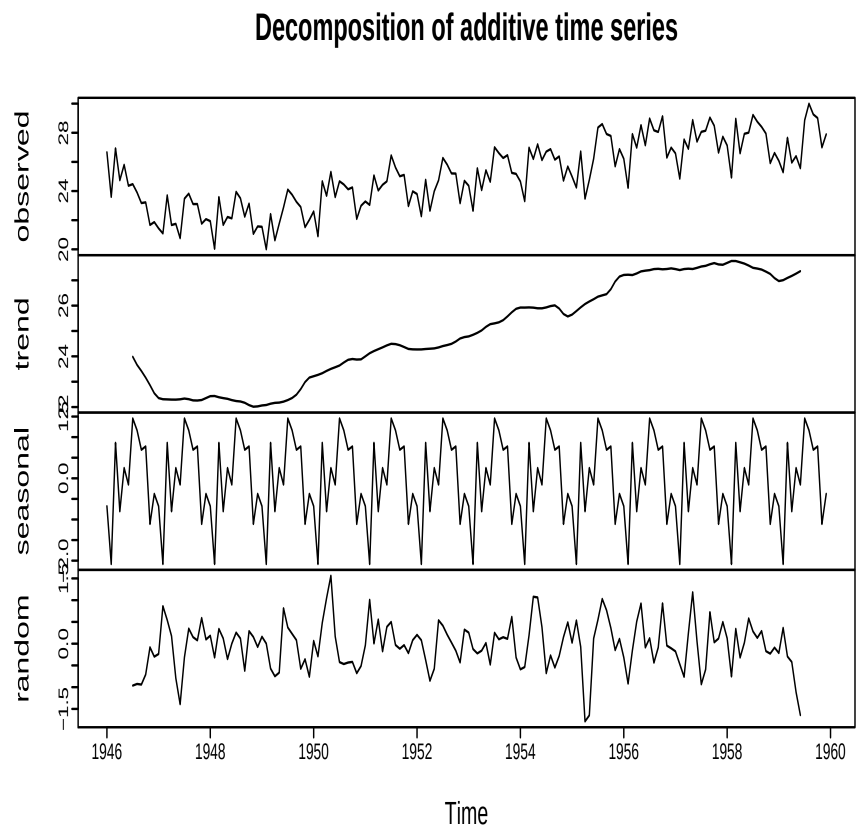

FIGURE 2.4: A time series decomposed into its three primary components.

be seen in Figure 2.5. Often stationary time series are repetitive, in other words showing constant auto-correlation and are considered the easiest type to model. A stationary time series can be composed of a seasonal element and/or a random component, thus:

$$\textit{stationary time series} = \textit{seasonality} \; +/\textit{or noise}$$

The second type of time series is the additive type. In this type all three components of the series are present, trend, seasonality and noise. The distinguishing feature here is the amplitude of the seasonal component in that it is quite regular being static over time. An example of an additive series can be seen in Figure 2.6. This time series is trending upwards overall but there is a clear repetitive pattern of peaks and troughs caused by the seasonality, with the heights of the peaks all being similar. We can consider an additive time series as:

$$\textit{additive time series} = \textit{trend} + \textit{seasonality} + \textit{noise}$$

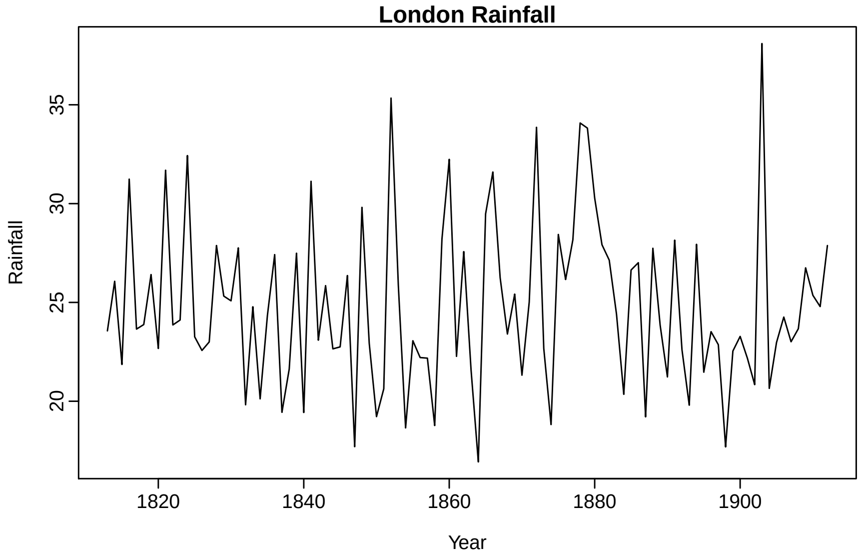

FIGURE 2.5: Example of a stationary time series which can be made up from noise and/or a seasonal component.

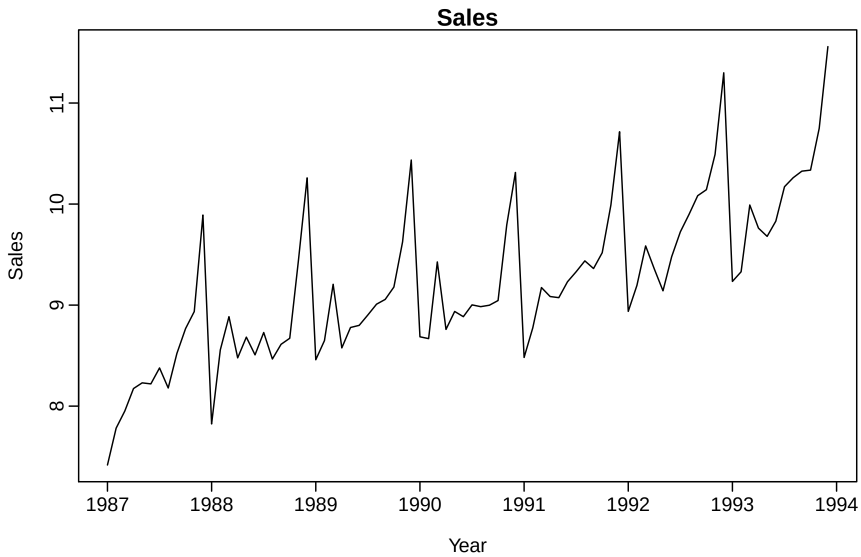

FIGURE 2.6: Example of an additive time series which results from all three components trend, noise and seasonality.

The third type of time series, as seen in Figure 2.7 is multiplicative. This is similar to the additive version except the amplitude of the seasonality increases over time. It can be considered as:

$$multiplicative\ time\ series = trend * seasonality * noise$$

Financial time series can be considered as containing all three elements of a time series. They can show properties of a stationary time series when they are range bound and only move between two values. At other times, markets trend strongly consistently, making

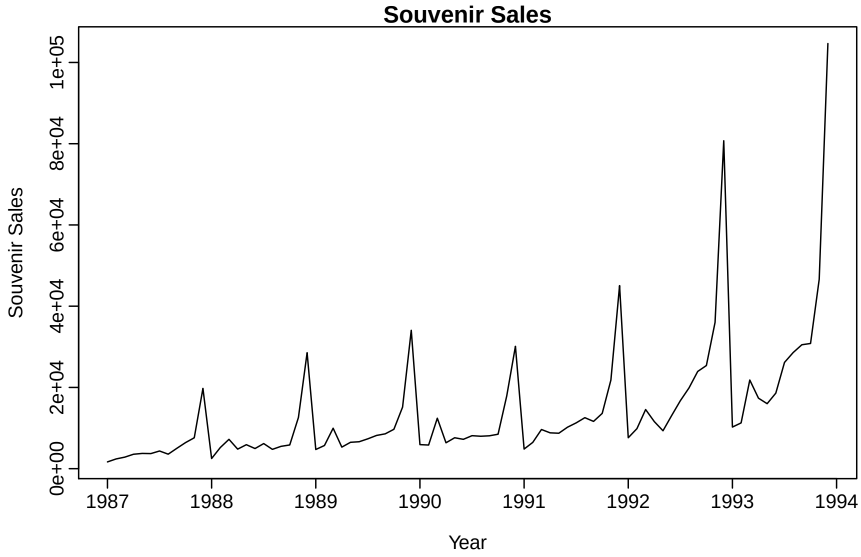

FIGURE 2.7: Example of a multiplicative time series resulting from the effects of trend, noise and seasonality.

new highs or lows and exhibit properties of an additive and occasionally a multiplicative series.

2.2.1 Time Series Smoothing

Smoothing is an important and widely adopted method to predict financial markets. Recent work on smoothing time series data has its origins in Brown (1959), Brown (1963), Holt (2004) and Winters (1960). Typically, the various smoothing techniques encountered are based around the concept of moving averages. This section will introduce a variety of smoothing methods commonly encountered in forecasting financial data.

2.2.1.1 Simple Moving Average (SMA)

A simple moving average is calculated from the value itself and its neighbours, which can be ahead or behind in the series. In this study values behind the current value are considered. The number of previous values included is often referred to as the “window” or “lag”, so if one was to consider the current value and four previous ones this would be considered a simple moving average of lag 5 (SMA5). An example of a simple moving average can be seen in Table 2.1, where a SMA5 of the closing price has been added.

TABLE 2.1: Example of a simple moving average of the closing price with a lag of 5 periods.

Date	Open	High	Low	Close	SMA5
02/01/14	9598	9621	9394	9400	NA
03/01/14	9410	9453	9368	9435	NA
06/01/14	9419	9469	9400	9428	NA
07/01/14	9446	9519	9417	9506	NA
08/01/14	9513	9516	9468	9498	9453
09/01/14	9492	9550	9403	9422	9458
10/01/14	9474	9530	9441	9473	9465
13/01/14	9498	9519	9457	9510	9482

2.2.1.2 Weighted Moving Average (WMA)

A simple moving average assigns equal importance to all data points being averaged, however if this is considered unsuitable a higher weighting can be applied to certain data points elevating their importance in the average and thus generating a weighted moving average (Devcic, 2010). Typically the more recent data points in a time series would be given higher importance. One common version of a WMA is to decrease the weighting by one for each period in the average. The formula for calculating a weighted moving average is:

$$((n * P_n) + (n - 1 * P_{n-1}) + ...(n - (n - 1) * P_{n-(n-1)})) \div (n + (n - 1) + ...n - (n - 1))$$

where:
n = the number of periods used in calculating the moving average
Pn = the price of the most recent period used to calculate the moving average

An extra column has been added to the data in Table 2.1 which contains the WMA for the last five close values. The current value was multiplied by 5, the previous one by 4, the previous one to that by 3 and so on. These five values were added together and divided by 5+4+3+2+1 to generate the WMA as shown in Table 2.2.

2.2.1.3 Exponential Moving Average (EMA)

An exponential moving average (EMA) is an extension of the weighted moving average (Ord, 2004). In comparison to the simple moving average, greater emphasis is given to the most recent data points and the resulting averaged values are closer to the actual

TABLE 2.2: Example of a weighted moving average.

Date	**Open**	**High**	**Low**	**Close**	**SMA5**	**WMA5**
02/01/14	9598	9621	9394	9400	NA	NA
03/01/14	9410	9453	9368	9435	NA	NA
06/01/14	9419	9469	9400	9428	NA	NA
07/01/14	9446	9519	9417	9506	NA	NA
08/01/14	9513	9516	9468	9498	9453	9471
09/01/14	9492	9550	9403	9422	9458	9461
10/01/14	9474	9530	9441	9473	9465	9466
13/01/14	9498	9519	9457	9510	9482	9481

observations of the data set. Weighting factors decay exponentially resulting in the emphasis falling on the recent values though not discarding the older ones totally.

2.2.1.4 Moving Averages in Practical Use

Moving averages are widely used in the financial world to predict the start of trends which is important as trends are considered the best opportunity to make profits from the markets. By their nature moving averages are lagged indicators in that they reflect market action from the past (recent or distant depending on the lag variable) and this can be considered a drawback. The lag period offers a trade off in terms of prediction. If the lag is short and/or weighting is applied the average is affected strongly by recent prices and trends can be detected in the early stages and trading profits can be enhanced. However when the average is close to the current price they have a tendency to generate “false signals” (see section 2.1.1 for an explanation of entry and exit signals), in other words prices may start to rise (or fall) but they are not actually in a trend, it is just the natural wax and wane of the market, and traders are said to be “whipsawed”. When the lag variable is long a different problem is encountered. For example, if a price moves above a long moving average the indicated trend is usually genuine, however by the time this is reflected in the average a lot of the trend has developed and the trader has lost a lot of potential profits. Thus there are pros and cons associated with using the different types of moving average.

2.2.1.5 Holt-Winters Smoothing Models

The exponential smoothing of a time series containing noise, trend and seasonality was developed by Winters (1960) who as a student of Holt, built upon his previous work, and is today called the Holt-Winters method. This method uses three parameters alpha, beta and gamma which define the degree of smoothing to be applied to the three components of the time series. Firstly, a value of alpha is used to dictate the amount of smoothing to apply, with high smoothing factors placing more emphasis on recent data points at the expense of those further away. In a data set with trend this simple exponential moving average doesn't perform well and a second order of smoothing is needed, so called "double exponential smoothing". The parameter beta in Holt-Winters defines this second order smoothing. Finally, if a seasonal component is also present in the data set a third level of smoothing is introduced making the process a triple exponential smoothing. It is this third level of smoothing that the parameter gamma refers to. Depending upon the nature of the time series one, two or all three of the parameters may be defined in the Holt-Winters methodology.

If researching a time series with no seasonality or trend use of the Holt-Winters model with the beta and gamma parameters set to false, in other words not used, is appropriate. Figure 2.8 shows the addition of an exponential smoothing line to the stationary data set introduced in Figure 2.5.

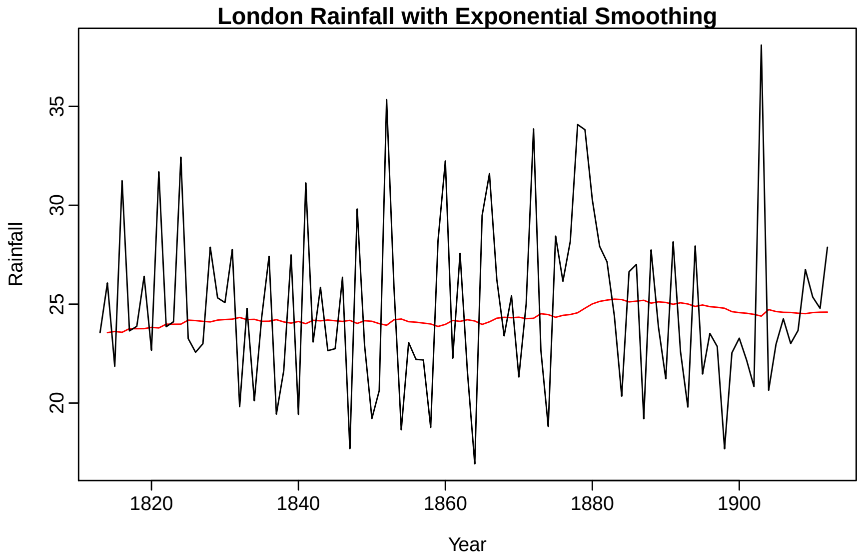

Figure 2.8: A time series with no seasonality or trend, showing the fitted line generated from Holt-Winters exponential smoothing with the beta and gamma parameters set to false.

If the time series is additive with a trend but without seasonality the use of Holt-Winters with values used for alpha and beta but with the gamma parameter set to false is appropriate. Such a time series can be seen in Figure 2.9 with the exponential smoothing. Finally if the time series contains all three components a smoothing line can be fitted using Holt-Winters exponential smoothing in which there are values for all three terms alpha, beta and gamma. Figure 2.10 is an example of a time series with both trend and seasonality and overlaid with Holt-Winters smoothing generated by using values for all three terms in the smoothing algorithm.

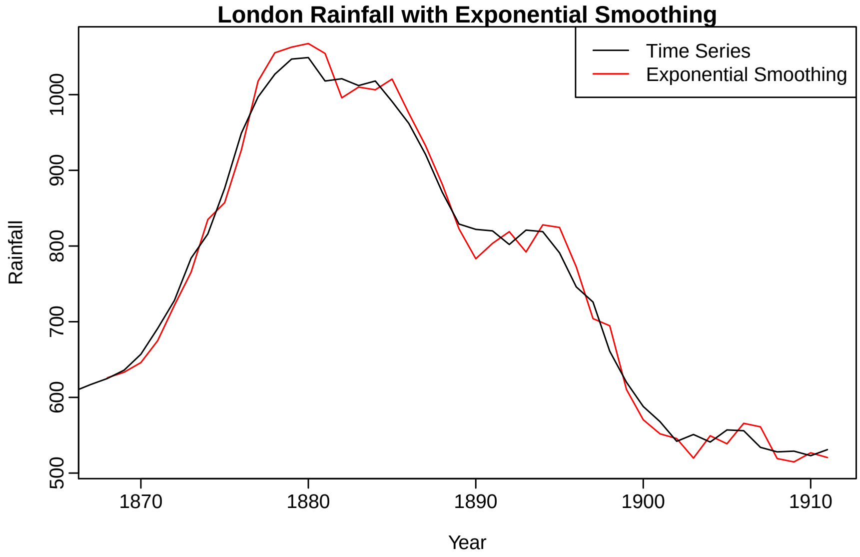

FIGURE 2.9: A time series with trend though no seasonality, showing the fitted Holt-Winters exponential smoothing with the gamma parameter set to false.

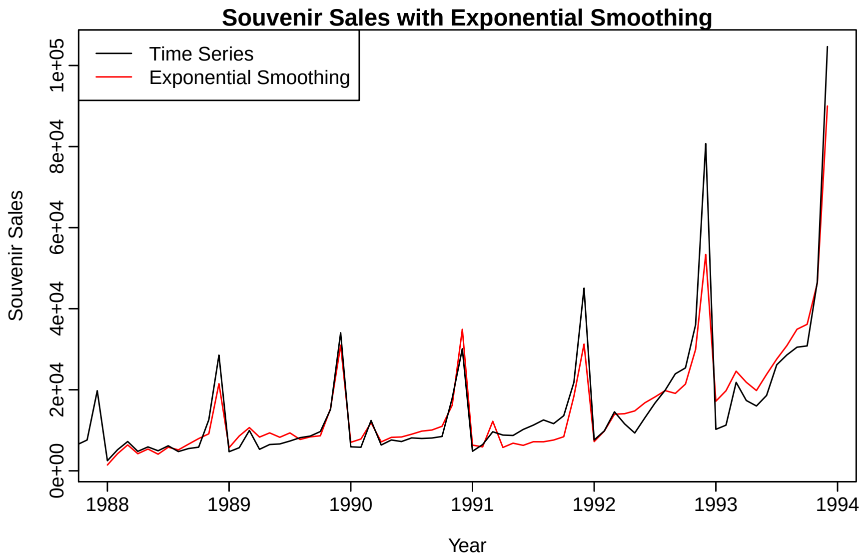

FIGURE 2.10: A time series with trend and seasonality, showing the fitted Holt-Winters exponential smoothing.

2.2.2 Auto-Regression Family of Models

2.2.2.1 Auto-Regression

Regression is the study of the impact of known variables (independent) on an unknown (dependent) variable and addresses questions such as how does a person's income vary with their years of education. The general equation for linear regression is given by:

$$y = a + bx + \varepsilon$$

where:
a is the intercept.
b is the co-efficient.
x is the independent variable.
ε is the error term.

In reality there are often a large number of independent variables that affect the unknown under study and thus multiple regression, shown below, is usually of interest.

$$y_1 = a + b_1 x_{1i} + b_2 x_{2i} + ... + b_n x_{ni} + \varepsilon$$

In a time series the preceding values often have a bearing on the current data point, and this is especially important in financial time series data. Thus auto-regression is the prediction of the current point from the use of previous values of the data point itself, and is given by:

$$t_t = c + b_1 * r_{t-1} + b_2 * r_{t-2}...b_p * r_{t-p} + \varepsilon$$

where:
c is the intercept, is often zero and the mean of the time series.
$b_1 - b_p$ are the independent variables, the previous values.
ε is random noise.

2.2.3 Auto-Regressive Moving Average (ARMA)

The auto-regressive moving average (ARMA)model, also known as Box-Jenkins (Box and Jenkins, 1970), combines moving averages with auto-regression. A model that uses moving averages to predict current values is given by:

$$-r_t = c + a_1 * ma_{t-1} + a_2 * ma_{t-2}...a_q * ma_{t-q} + err_t$$

ARMA combines the moving average model with auto-regressive terms to generate:

$$\begin{aligned} r(t) = {} & c+ \\ & b_1 * r_{t-1} + b_2 * r_{t-2}...b_p * r_{t-p}+ \\ & a_1 * ma_{t-1} + a_2 * ma_{t-2}...a_q * ma_{t-q} \\ & + err \end{aligned}$$

where:
c is the intercept, which is often zero and the mean of the time series.
$b_1 - b_p$ are the independent variables, the previous values in the auto-regression term.
$a_1 - a_p$ are parameters of the moving average model.
ε is random noise.

An ARMA(1,1) model uses the previous value in the auto-regression term and the previous value's moving average. Thus in general terms an ARMA(p,q) model uses the previous p values in the auto-regression term and the moving averages derived from the last q values. There are therefore three steps in developing an ARMA model:

1. identification step in which the order of AR and MA components is determined
2. parameter estimation
3. forecasting

ARMA models have certain intrinsic properties that may be considered drawbacks, namely the requirement for the time series to be stationary with no trend and also linear and the difficulty in deriving the correct parameters to use in the model. In order to overcome these restrictions researchers have tried a number of approaches to enhance the effectiveness of ARMA models.

The problem of model and parameter selection in ARMA models has also been addressed by Rojas et al. (2008). The authors make the point that in traditional research choosing the correct model is time consuming and requires a large degree of expertise. In order to circumvent these issues they propose an automatic model selection method to speed up the process, remove the need for expert intervention and allow the processing of a large number of time series. In a similar study Qian and Zhao (2007) investigate how to determine model selection where there are potentially millions of candidate ARMA models available for the time series. Again, the authors propose an automatic selection algorithm centred on the Gibbs sampler. The proposed method allows for various problems typically encountered in selecting ARMA models and the resulting choice was used to generate a prediction of China's Consumer Price Index (CPI).

2.2.4 Auto-Regressive Integrated Moving Average (ARIMA)

One limitation with the ARMA model and indeed other approaches is that it is assumed that the time series is stationary, it doesn't have trend and has constant variance and mean (Shumway and Stoffer, 2010). In reality of course many time series data sets have trend, and in the world of financial data this is also true. In order to account for trend in a time series it is often transformed into a stationary data set, modelling is then performed on this adapted data after which it is returned to its original state. In effect the trend aspect is removed, modelling is done, then the trend component is added back into the data.

One such method for removing trend is differencing (Mills, 2011). Differencing is the technique of replacing the actual values of the observations with the values of the differences between them. This is represented as:

$$Diff1_t = r_t - r_{t-1}$$

Differencing is the same as calculating the derivative of the series, thus a time series that has under gone differencing is considered "integrated". If taking this so-called first difference doesn't remove the trend one can go further and use the second difference:

$$Diff2_t = (r_t - r_{t-1}) - (r_{t-1} - r_{t-2})$$

Addition of an integration step to the ARMA model results in an auto-regressive integrated moving average (ARIMA) model, with the general formula:

$$\begin{aligned} r(t) = c+ \\ & b_1 * r_{t-1} + b_2 * r_{t-2} ... b_p * r_{t-p} + \\ & a_1 * ma_{t-1} + a_2 * ma_{t-2} ... a_q * ma_{t-q} \\ & d_1 * diff_{t-1} + d_2 * diff_{t-2} ... d_d * diff_{t-d} \\ & + err \end{aligned}$$

where:

c is the intercept, which is often zero and the mean of the time series.

$b_1 - b_p$ are the independent variables, the previous values in the auto-regression term.

$a_1 - a_p$ are parameters of the moving average model.

$d_1 {-}_p$ are the parameters of the differencing term. ε is random noise.

ARIMA models are usually referenced as ARIMA(p,d,q) with p the number of terms used in the auto-regression, d the number of differencing terms and q the number of terms used in the moving average. A summary of which model (Holt-Winters, ARMA or ARIMA) to use with which type of time series can be seen in Table 2.3.

TABLE 2.3: Appropriate models for use with time series data.

Model	**Time Series Required**	**Assumes Correlation**	**Trend**	**Seasonality**
Holt-Winters	Short Term	N	Y	Y
ARMA	Stationary	Y	N	Y
ARIMA	Non-stationary: Additive or Multiplicative	Y	Y	Y

2.2.5 ARIMA Parameter Selection

An important aspect of building time series models with ARIMA techniques is the choice of parameters to use. Auto-correlation (AC) and partial auto-correlation (PAC) are important measures in the selection process of these parameters (Mills, 2011).

Correlation is the measure of how one variable changes with a second one. For example if variable A increases while variable B increases they are positively correlated and conversely they are negatively correlated when one decreases as the other increases. Further, correlations are measured by degree on a scale of 1 to -1, with 1 being perfectly correlated. A value of 1 indicates that the two variables increase together perfectly in sync, whereas a value of -1 suggests that as one variable increases the other decreases by the same amount. Finally a value of 0 is indicative of no correlation at all between the two variables.

Auto-correlation is the correlation between an attributes value now and the same attribute's value in the past or future (Shumway and Stoffer, 2010). Typically with financial data we are interested in the correlation with values in the past. The interval between the value of interest and the previous observation used in determining the correlation is known as the lag. Thus the correlation between the current observation and the previous one may be of interest, and this is a lag of +1, while a value five time intervals previous is +5. Non-intuitively positive values for lags refer to the past while negative values are in the future.

A correlogram is a matrix plot of auto-correlations over a series of time lags. Correlograms are used in checking data for randomness and in the model identification stage of the ARMA methodology (see section 2.2.3). Data is considered random if the auto-correlation value is close to zero. In general a data set's randomness needs to be checked in order to confirm the validity of many statistical tests. Thus a correlogram helps to determine if data is random or if an observation is related to an earlier one, thereby helping in the determination of an appropriate ARMA model. Figure 2.11 is the correlogram of auto-correlation and Figure 2.12 the correlogram of partial auto-correlation for a data set of rain fall figures.

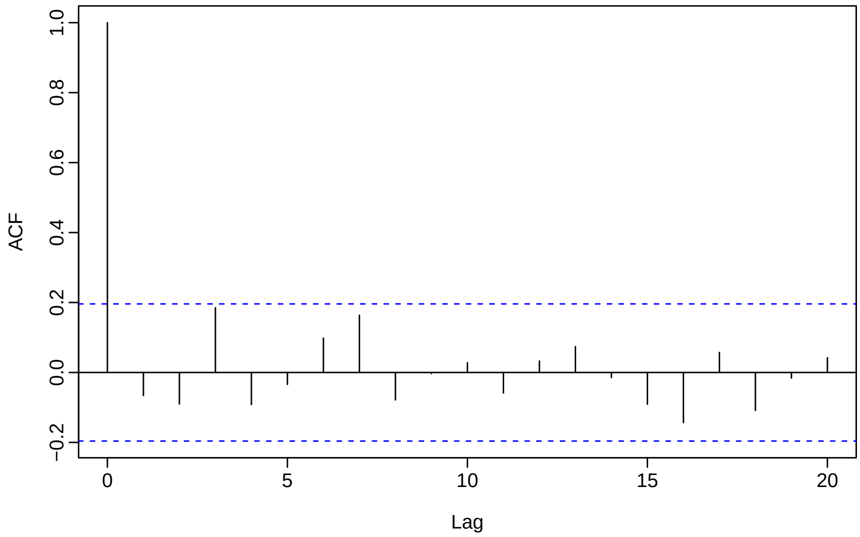

FIGURE 2.11: Correlogram of auto-correlations.

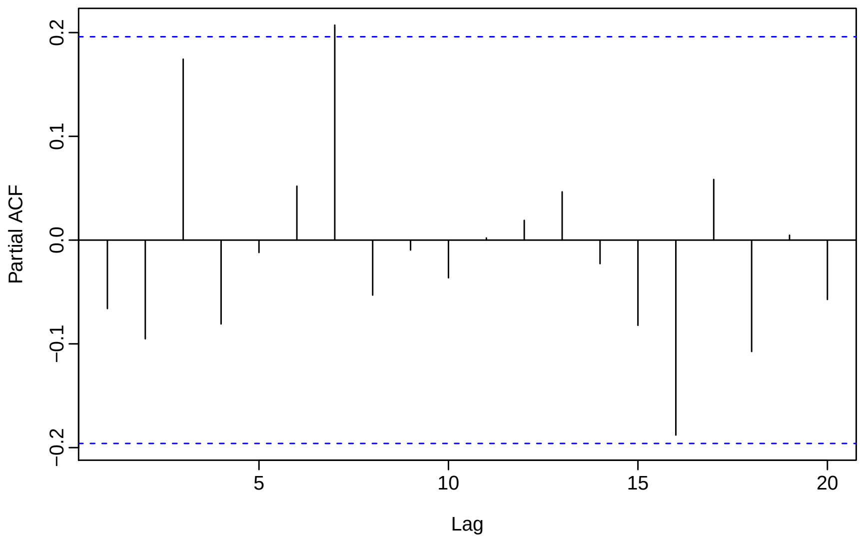

FIGURE 2.12: Correlogram of partial auto-correlations.

The partial correlation is defined as the degree of correlation not already explained by the correlations previously measured. If the regression of variable A on variables B1, B2 and B3 is considered the partial correlation between variables A and B3 is the degree of correlation not accounted for by their common correlations with variables B1 and B2. In a similar manner the partial autocorrelation is the unexplained correlation after considering the variable and itself at an earlier time period. In a series, if a variable

A at time t is correlated with an earlier lag at time t-1 it follows that the variable at t-1 itself is correlated with the previous variable at lag t-2. By extension the variable at time t should also be correlated with the variable at lag t-2, as the correlation will propagate through the series. The partial autocorrelation is the difference the expected correlations due the propagating factors and the actual correlation measured.

If the ARIMA model is ARIMA(p,d,0) or ARIMA(0,d,q) then the ACF and PACF plots are helpful in deciding the values for p or q. If both p and q are positive, the ACF and PACF are not useful in estimating the values for p and q. An ARIMA(p,d,0) model may be appropriate if the ACF and PACF plots of the stationary data exhibit an exponentially decaying pattern in the ACF and a large spike at lag p in PACF plot. Conversely an ARIMA(0,d,q) model may be appropriate if the PACF is decaying exponentially and there is there is a significant spike in the ACF plot at lag q.

2.2.6 Hybrid Models

Auto-regressive (integrated) moving average models have shown themselves to be important modelling methods for time series data, including financial time series data. However the techniques have limitations that have detracted from their popularity, namely their assumption of a linear relationship and the need for a lot of data to produce accurate results. In order to address these limitations a variety of hybrid solutions have been proposed in which ARIMA models are combined with other techniques, often non-linear prediction algorithms (Wang et al., 2012, Khashei and Bijari, 2012, Aladag et al., 2009).

One combination that has found a lot of attention in the literature is the combination of Artificial Neural Networks (ANNs) with ARIMA. Khashei et al. (2009) report on the use of this combination in a attempt to predict the future price movement in gold and US dollar/Iran rials financial markets. The workers report favourable results in comparison to the techniques alone and suggest the method as having potential for accurate predictions of non-linear time series data. In a similar study Zhang (2003) applied a combination of ARIMA and ANN to various data sets including the British pound/US dollar exchange rate. They observe that in the literature in general these two popular techniques are frequently compared in terms of predictive power with the reported results non-conclusive. Results from the three data sets modelled show that the combination of the two methods outperform the individual ones when the mean squared error (MSE) and mean absolute deviation (MAD) are used as the measure of forecasting accuracy.

Fatima and Hussain (2008) also investigated the impact of a hybrid approach in modelling short term predictions for the Karachi Stock Exchange index (KSE100). The authors reported comparison results for ANN versus ARIMA and a hybrid of ANN/ARIMA. The hybrid solution out-performed the individual ARIMA and ANN models. It is postulated that a rationale for this is that at any point in time financial markets are subject to linear, non-linear and volatility patterns as the cumulative effects of government fiscal and monetary policies and general rumour and political instabilities feed into the market. Under these complex conditions simple models can only capture one aspect of the underlying factors affecting the price series. A hybrid combination approach is more successful as more of the market variance is modelled.

Kriechbaumer et al. (2014) reports on a further hybrid approach to forecast the prices of aluminium, copper, lead and zinc. Previous research has indicated that these markets exhibit a strong cyclic behaviour. In an attempt to factor this into the predictive model ARIMA was coupled with a wavelet approach. Wavelet analysis decomposes a time series into its frequency and time domains in an attempt to isolate this cyclic behaviour. The performance of the ARIMA modelling was shown to be enhanced substantially by the addition of wavelet based multi-resolution analysis (MRA) before performing the ARIMA analysis.

Tan et al. (2010) have also reported the combination of wavelet analysis and ARIMA in the prediction of electricity prices. The general method employed is to transform the original time series data set into a collection of sub-series through the application of wavelet analysis. Subsequent to the transformation a prediction for each sub-series can be made with ARIMA modelling. The final forecasted result is obtained by reforming the sub-series back into the original time series. The authors report results showing the enhanced predictive power of the ARIMA wavelet hybrid approach compared to ARIMA and GARCH models used in isolation.

Pai and Lin (2005) reported on attempts to overcome the limitation of ARIMA models in that the time series must be linear by use of an hybrid ARIMA/Support vector machine (SVM) combination. SVM have been successfully applied to non-linear regression problems and the authors have harnessed the strengths of both methodologies in order to predict the prices of a selection of fifty stocks. Results from the work show that the hybrid method out-performs the ARIMA and SVM methods individually.

Rout et al. (2014) report the use of ARMA models in the prediction of exchange rates. The workers note the limitations of ARMA in that the time series data must be linear and stationary, a condition often not met in practical situations and the difficulty in deriving steps one and two (listed previously) in developing the ARMA model. In order to overcome these limitations ARMA is combined with differential evolution (DE) in order to determine the models feed-forward and feed-back parameters. The results from the prediction models generated are compared with models resulting from ARMA in conjunction with particle swarm optimisation (PSO), cat swarm optimisation (CSO), bacterial foraging optimization (BFO) and forward backward least mean square (FBLMS). The workers conclude that the ARMA - DE model produces the best short and long-range predictions from the options tested and is a potentially valuable method in predicting exchange rates on the international finance markets.

Chapter 3

Methodology

3.1 Data Collection and Quality

The data used in this study was freely collected from the Yahoo finance web site (www.yahoo.com). It is of high quality with no missing values and represents the opening, high, low and closing prices for each day that the particular market indice was open for trading.

3.2 Data Description

Data from a variety of national stock market indices was employed in this study. The indices were from a variety of geographic locations with FTSE (UK), DAX (Germany) and CAC (France) all being in Europe, the Dow is from the US, the Nikkei from Japan and AORD from Australia. The data is in the form of so-called daily OHLC (daily open, high, low and close prices) for Monday to Friday (excluding appropriate national holidays) for the period 2000 until the end of 2013. A schematic representation of daily OHLC data can be seen in Figure 3.1. The first six observations from the DAX data set (German national indice) can be seen in Table 3.1.

The final six observations from the DAX data set can be seen in Table 3.2. Over the period of the data (2000 until the end of 2013) the DAX started at 6691 and finished at 9552. Summary statistics for the DAX data set can be seen in Table 3.3. The data set contains 3621 observations and the closing price has ranged between 2202 and 9742 over the period. A graph of the closing prices from 2000 to 2013 and be seen in Figure 3.2 and a graph for 2013 can be seen in Figure 3.3.

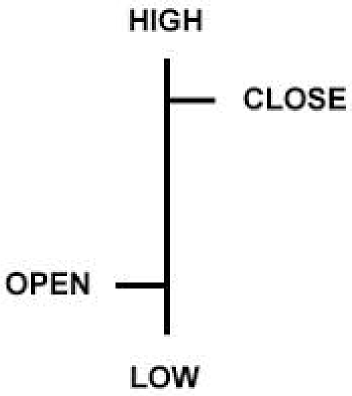

FIGURE 3.1: A schematic representation of open, high, low and closing prices (OHLC).

TABLE 3.1: First 6 rows of the DAX data set

Date	Open	High	Low	Close
03/01/2000	6962	7159	6721	6751
04/01/2000	6747	6755	6510	6587
05/01/2000	6586	6586	6389	6502
06/01/2000	6501	6539	6403	6475
07/01/2000	6490	6792	6470	6781
10/01/2000	6785	6975	6785	6926

TABLE 3.2: Final 6 rows of the DAX data set

Date	Open	High	Low	Close
16/12/2013	9005	9188	8998	9164
17/12/2013	9143	9162	9085	9085
18/12/2013	9145	9191	9122	9182
19/12/2013	9280	9352	9257	9336
20/12/2013	9371	9413	9353	9400
23/12/2013	9400	9400	9400	9400

TABLE 3.3: Summary statistics of the DAX data set.

Statistic	N	Mean	St. Dev	Min	Max
Open	3,621	5,858.36	1,559.40	2,203.97	9,752.11
High	3,621	5,906.70	1,561.17	2,319.65	9,794.05
Low	3,621	5,804.85	1,557.49	2,188.75	9,714.02
Close	3,621	5,857.74	1,559.39	2,202.96	9,742.96

Each data set has a particular set of characteristics and these are important when technical analysis and other analytical techniques are applied to the data set (Chen and Li, 2014, Matheson, 2012). A variety of these are explored in the following sections. The average amount a market moves is investigated and the term Average True Range

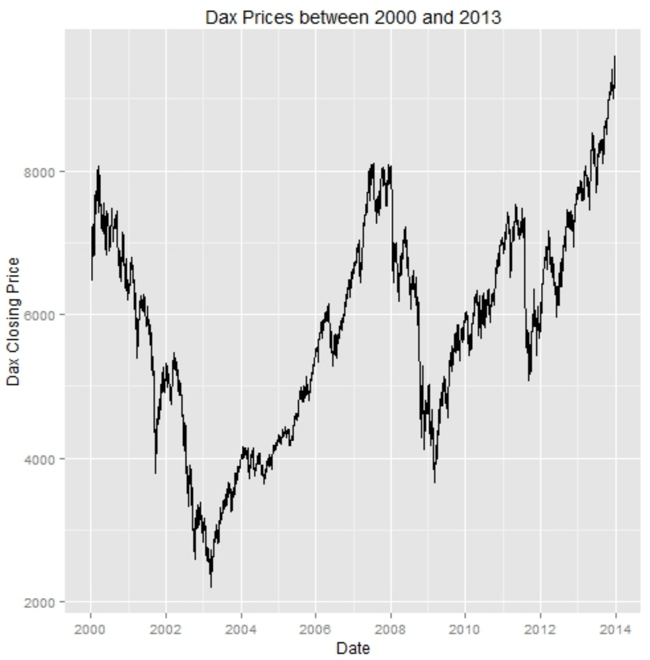

FIGURE 3.2: Graph of German DAX between 2000 and 2013.

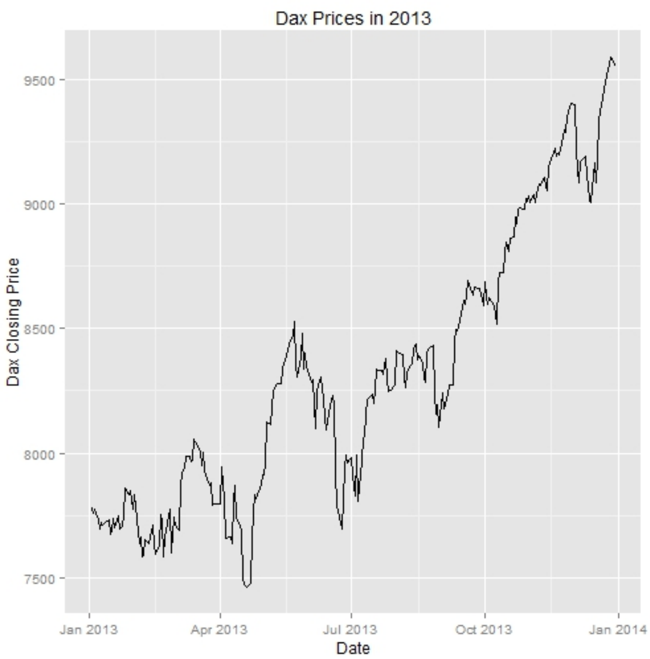

FIGURE 3.3: Graph of German DAX in 2013.

is introduced and defined for the data sets. Where the opening and closing prices are in relation to the previous day's high and low values are also considered. Finally, the distance between the day's opening and high prices and opening to low prices are investigated. The relative ratios of these values are important when considering which technical analysis may be best suited to a particular market.

3.2.1 Average True Range (ATR)

Wilder (1978) introduced the concept of Average True Range (ATR) as a way to measure a market's volatility or the amount the price is likely to move in any one day. Initially the True Range (TR) is calculated as the maximum of:

1. the today's high price minus today's low price.
2. the absolute value of the today's high minus the previous day's closing price.
3. the absolute value of the today's low minus the previous day's closing price.

Having calculated the TR, an average of a previous number of days is used to derive the ATR. Typically the TR values from the previous 14 days are used.

Absolute values are used in the calculation of the ATR as we are not concerned with the market direction but rather the the amount the market is likely to move. ATRs are typically quoted as absolute values and as such markets trading at higher prices will have higher ATRs. For example the Japanese Nikkei with a value of 14000 will move more in a day than the French CAC with a value in the 4000's.

Dividing the ATR by the closing price is a useful way to see how a security's volatility varies over time. Table 3.4 shows summary statistics for the ATR and ATR divided by closing price and Figure 3.4 is a graph of how ATR divided by closing price has varied for the DAX between 2000 and 2013. In absolute terms the ATR varied between 36 and 316, however the value of the indice itself varied a lot. Looking at the ATR value divided by the closing period it can be seen that over the period of 2000 to 2014 the mean value is approximately 2. Thus on average the market can be expected to move 2% of the closing price in any one day. However this value has varied between 0.7% in periods of low volatility to a value of 6.7%.

3.2.2 Opening Price

Where a market opens in relation to the previous day's high and low price varies across the data sets. This is important and can influence the technical analysis indicator or

Table 3.4: ATR and ATR divided by closing price for the DAX between 2000 and 2013

Statistic	N	Mean	St. Dev	Min	Max
ATR	3,556	108.29	45.53	36.07	316.04
ATR/Close	3,556	1.995	1.065	0.700	6.740

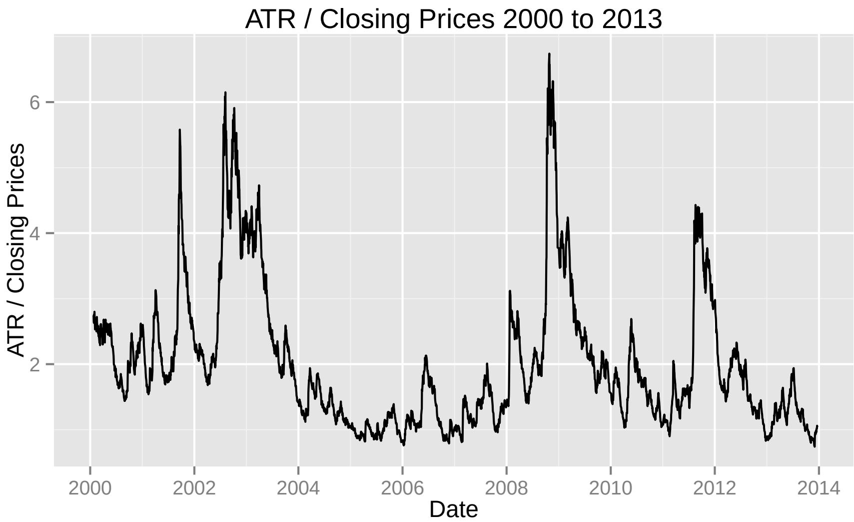

Figure 3.4: ATR of DAX divided by closing price between 2000 and 2013.

trading system to utilise. Table 3.5 lists opening price statistics for a variety of national indices. The table lists the number of times that opening prices are between the previous day's high and low prices. These statistics are useful in characterising a market in terms how they move out of hours and can have an impact when choosing a trading system.

Table 3.5: Opening prices in relation to the previous day's high and low values.

Market	Opening Price between Previous Day's High and Low (%)
DAX	75
FTSE	90
CAC	60
Dow	98
Nikkei	53
AORD	79

3.2.3 Closing Price

In a similar fashion to the opening prices the position of the closing prices in relation to the previous day's high and low price are also of interest. In this case, the percentage of closes outside the previous high/low price may indicate that the market may be a good choice for a breakout type of trading system (see section 4.6.1 for details of breakout systems). Likewise the opposite situation occurs if a market frequently finishes within the previous day's high and low levels and may be a candidate for a reversal type of system. The statistics for various national indices can be seen in Table 3.6. Looking at these figures it would suggest that the Dow with a low ratio of finishing outside the previous period's high low values may be a candidate for a reversal type of system and conversely the Japanese Nikkei has a high percentage value and potentially a candidate for a break-out system.

TABLE 3.6: Number of occasions when closing prices finished outside the previous day's high or low values.

Market	Closing Price outside Previous Day's High and Low (%)
DAX	56
FTSE	56
CAC	58
Dow	39
Nikkei	63
AORD	60

The range from opening price to closing price, either up or down, is of interest. Table 3.7 lists the minimum and maximum values in this range and Table 3.8 shows the quantiles for this price range.

TABLE 3.7: Minimum and maximum values for the open to close price range.

Market	Min Value	Max Value
DAX	0	508
FTSE	0	431
CAC	0	313
Dow	0	950
Nikkei	0	1333
AORD	0	347

TABLE 3.8: Quantile values for the open to close price range.

Market	25%	50%	75%	90%
DAX	16	39	75	508
FTSE	15	33	63	431
CAC	11	26	49	313
Dow	27	61	119	950
Nikkei	32	71	133	1333
AORD	8	19	36	347

3.2.4 High/Low Price

Table 3.9 shows the percentage of times that either today's high price crosses yesterday's high or today's low prices dips below yesterday's low value. The final closing price may be between yesterday's high and low or outside of it. The second column of Table 3.9 is the number of times when today's values crossed both the previous low and the previous high in the same day. This is also known as an Engulfing Candlestick (see section 4.7.2). In all the indices the previous day's high or low value is reached the following day in a large number of instances, in the case of the Nikkei 90% of the time. Conversely, the likelihood of both the previous day's high and low values being touched are low, only 5% of occasions in the Australian AORD.

TABLE 3.9: Number of occasions when today's high or low prices crossed the previous day's high or low values.

Market	Crosses either previous day's High or Low (%)	Crosses both the previous day's High and Low (%)
DAX	89	9
FTSE	87	8
CAC	90	10
Dow	88	9
Nikkei	90	8
AORD	86	5

3.2.5 OH/OL Price Fluctuations

The movements in prices between the open and high (OH) and open to low (OL) are interesting and can have an influence on any trading systems developed. On any given day prices open, move to their lowest point, move to their highest point and then close (not in any particular order). From the OHLC data used in this study the order of these events can not be determined or even the number of times in a day these price points are reached.

In this section we are concerned with the relative sizes of these two price movements, the day's high price minus the opening price (OH) and the opening price minus the low price (OL), one of which is usually greater than the other. We will define the daily "minor" price fluctuation as the smaller of the two price movements. Likewise we will define the larger value as the "major" price fluctuation.

Considering the minor price fluctuation, the range of values encountered in the indice markets under study can be seen in Table 3.10. In all cases the minimum value is zero, in other words the market opening price and either the day's high or low price were the same, the market didn't dip below or above this level. The second column in Table 3.10 is the maximum value. In the case of the German DAX, there was a day when the market moved 189 points away from its opening price but also moved further in the opposite direction away from the opening price. Clearly this was a highly volatile day on the German markets.

TABLE 3.10: Minimum and maximum values for the smaller of the daily OH or OL price movement - the "minor" move.

Market	Min Value	Max Value
DAX	0	189
FTSE	0	186
CAC	0	134
Dow	0	379
Nikkei	0	310
AORD	0	114

The quantiles of the minor price movements can be seen in Table 3.11. The 90% quantile is the level at which 90% of the time the minor move is less than this level. This value may be important to know and understand when considering break-out type of systems (see section 4.6). Looking at the value of the DAX we can see that the 90% quantile level occurs at 46, which indicates that if the market has moved to this level it is unlikely to be the day's minor move (whose level 90% of the time is below this). Perhaps a break-out type of system may be profitable at this point, as once the market has moved this far it is usually a major move and may be expected to continue further in the same direction.

TABLE 3.11: Quantile values for the smaller of the days OH or OL price movement - the "minor" move.

Market	25%	50%	75%	90%
DAX	5	15	29	46
FTSE	0	7	20	33
CAC	4	11	19	31
Dow	12	43	75	113
Nikkei	5	21	43	72
AORD	0	1	7	13

In contrast to the minor daily price fluctuation, the "major" price fluctuation is defined as the largest of the OH or OL values. The range of values encountered in this price fluctuation in the indice markets can be seen in Table 3.12 and the quantiles of the major price movements can be seen in Table 3.13. Considering the DAX once more, it can be seen that the 25% quantile is approximately equal to the 90% quantile of the minor fluctuation. Thus if the DAX moves approximately 50 points away from the opening it is unlikely to be the smaller of the price movements and much more likely to be part of the larger movement. Knowledge of the minor and major price fluctuations may be useful in developing trading systems.

TABLE 3.12: Minimum and maximum values for the larger of the days OH or OL price movement - the “major” daily price fluctuation.

Market	Min Value	Max Value
DAX	0	530
FTSE	0	471
CAC	0	359
Dow	0	992
Nikkei	0	1737
AORD	0	347

TABLE 3.13: Quantile levels for the larger of the day’s OH or OL price movement - the “major” daily price fluctuation.

Market	25%	50%	75%
DAX	43	69	106
FTSE	37	56	86
CAC	30	45	69
Dow	92	131	190
Nikkei	76	118	184
AORD	18	30	48

A final consideration in this section is the range of the open to close prices detailed in section 3.2.3. Again considering the German DAX it can be seen that the 50% quantile value is 39, as shown in Table 3.8, which is below the 90% minor fluctuation level.

3.3 Software Tools

3.3.1 R and R Studio

Experimental results and graphs were produced with the open source programming language R version 3.0.2. For help in the creation and organisation of the R code for this thesis the open-source development environment R Studio version 0.98.490 was used extensively. The following packages were immensely helpful in the preparation of this thesis:

- TTR - provided technical analysis functions
- xts - irregularly spaced time series
- forecast - time series forecasting

- candlestick - Japanese candlestick patterns

3.3.2 Rapid Miner

Rapid Miner version 5.3, a market leading open-source data mining and predictive analytic platform, was used for building hybrid ARIMA models. The base system and time series plug-ins were used.

3.4 Methodology

The aim of this study was to predict market movement, not in terms of absolute values but more in terms of general direction. The trading algorithms developed in this study were based around a daily time period. The data used was daily data containing open, high, low and closing prices. The trading algorithms developed typically opened a trade at the day's opening price and closed them at the day's closing prices. There are some deviations from this, for example trades are closed at some pre-defined point in the event of a stop loss being entered or trades running from close to close, and these are detailed in the text at the appropriate place.

Prior to passing the data sets to the trading algorithms they were subjected to either technical or time series analysis. Typically technical analysis results in the generation of a value (for example a moving average or stochastic value) which is considered meaningful in the future prediction of the financial market. The forecasts produced from the technical analysis and time series models were consumed by the trading algorithms and used to decide in which direction the daily trade would be made, i.e. would the market rise or fall.

The success or failure of any particular method was determined by the number of points gained or lost in these trading algorithms. In turn these points can be considered money and the net number of points can represent a profit or loss (PL). The results presented in Chapters 4 and 5 report either the total PL of a system or the average PL per trade. The latter value is probably most useful when it comes to comparing systems that don't enter trades every day, for example when one uses candlestick patterns which only occur very infrequently. Also the direction of trade is also a consideration. Results are separated into long (a buy is made in the expectation of the market rising) or short (a sell is made in the expectation of the market falling).

Chapter 4

Technical Analysis

4.1 Introduction

This chapter investigates whether technical analysis can provide a positive expectancy for financial traders (Kuang et al., 2014, Hsu et al., 2010). A variety of technical analysis indicators are employed including MACD, Aroon, Stochastics Oscillator and Rate of Change (ROC) indicator. The experimental results from using these indicators are presented in groupings based on the general category of indicator such as trend identification, market reversal and momentum indicators (Taylor, 2014). Some technical indicators have a role to play in more than one area, for example MACD can be considered a trend detection indicator or a market reversal indicator.

The effectiveness of a particular indicator or system is measured in terms of "points" gained, which is also referred to as "PL" (which stands for to profit and loss). The results presented in this chapter are mainly based around systems in which a trade is opened and closed each day, producing a daily PL either positive or negative. The sum of all the individual days produces the total system PL and these values are reported in the results tables. For example, if the market moved from 6000 to 6200 in any one day a PL of either 200 (6200 - 6000) or -200 (6000 - 6200) depending upon which way the trade was placed, would be added to the overall system results.

In addition, the results are presented such that returns from "going long" (expecting the market to rise) are presented separately from the opposite scenario of "going short". This is because market behaviour is often different while it is rising than it is while falling and systems may be more adept at predicting price movements in one of the directions. Further, transactions costs are not taken into account in the results and these would typically be 1 point per trade for the European markets, 2 points for the Dow and 10 for the Nikkei. Thus if a system made a PL of 1000 but it required 2000 trades at 2 points per trade, in reality the system would have lost money.

The results presented in this chapter and the following one are based around trading systems. Essentially the methodology concerned, technical analysis in this chapter and time series analysis in the next, attempt to predict future market direction. The values from the various indicators and forecast techniques are fed into a variety of trading algorithms which use the forecast information to decide whether to make long (expect the market to rise) or short (expect the market to fall) trades. For consistency the algorithms all return the same data object containing the following results:

1. Mkt - the name of the financial market such as DAX, FTSE etc.
2. S Loss - the value of any stop loss applied
3. LongPL - the profit or loss generated from just the “Long” trades.
4. ShortPL - the profit or loss generated from just the “Short” trades.
5. L Win % - the percentage of time the Long trades win.
6. L Trades - the number of Long trades executed.
7. Av L PL - the average profit or loss generated from each Long trade.
8. S Win % - the percentage of time the Short trades win.
9. S Trades - the number of Short trades executed.
10. Av S PL - the average profit or loss generated from each Short trade.
11. misc - miscellaneous information such as the SMA used in the algorithm.

The results from Long and Short trades in particular trading algorithm are considered separately as frequently markets behave differently as they move up as opposed to as they fall. Further, the percentage of times the algorithm results in winning trades, the number of trades and the average profit or loss (PL) for each trade is reported for both Long and Short trades. The average PL is primarily reported in the following results tables because this allows comparisons between systems that generate a lot of trades with those such as the algorithms based on candlestick patterns that results in only a small number of trades.

4.2 Baseline Systems - Naive Methods

Initially two very simple ideas were explored in order for the results to be used as baselines against which the technical indicators explored in the rest of the chapter and the time series models of Chapter 5 can be compared. There is an expectation that the use of technical indicators will produce systems that provide much better results than these two so-called naive systems.

The first system simply uses the idea that markets tend to increase in value over time. The algorithm applies a naive approach and simply enters a trade each day expecting the market to rise. The well-known method of ”Buy and Hold” applies the same principles. The total PL of the resulting system is the the sum of all the daily close minus open prices. This approach has been named a “Naive Long System”.

The second approach is equally simplistic, and again is based around opening and closing a trade each day. A notable difference from the first naive system is that the algorithm can result in either a buy or a sell (expecting the market to decline in value) occurring. If a market increased in price the previous day the algorithm “reverses” it and expects the market to fall today. Likewise if the market had fallen the previous day the system buys the market today. This idea has been named the “Naive Reversing System”.

4.2.1 Naive Long System

The results of the naive long system can be seen in Table 4.1. The R code for the algorithm which generates the results shown in Table 4.1 can be seen in Appendix A section A.1.2.1. For comparison purposes, the opening prices of the indices in January 2000 along with the closing prices in 2013 can be seen in Table 4.2. In this period three of the indices increased in value (DAX, Dow and AORD) and three decreased (FTSE, CAC and Nikkei).

Interestingly, the PL produced from the Naive Long System doesn’t match the price differentials seen in Table 4.2. The German DAX indice produced a marked loss in the naive system even though it actually increased 37% during this period. The Japanese Nikkei declined by over 2600 points in this period, whereas the system reported a loss of over 18000 points in the same period. On the other hand the US Dow increased by around 5000 points during the period of the study but the trading algorithm produced a positive result of almost 10000. These discrepancies can be explained by the fact that the system was using prices from the market’s opening to closing times, which represents approximately eight hours of trading between 8am and 4pm local time. These price movements don’t account for the rest of the hours, the so-called out of market hours, when the market prices also change. Clearly the markets show different characteristics in the amount they move during market hours compared to out of market hours. The Nikkei, DAX and CAC have a tendency to fall during market hours and rise during out of market hours. The opposite situation occurs for the Dow.

Table 4.1: Naive Long System. A very simple system in which the algorithm assumes the market will rise and enters a long trade each day.

Mkt	LongPL	L Win %	Av L PL
DAX	-1714	52	0
CAC	-6725	50	-2
FTSE	149	51	0
Dow	9816	53	3
Nikkei	-18125	49	-5
AORD	972	52	0

Table 4.2: Prices of six national indices in January 2000 and December 2013.

Date	Start 2000	End 2013	Difference	% Change
DAX	6961	9552	+2591	+37
CAC	6024	4250	-1774	-29
FTSE	6930	6749	-181	+-3
Dow	11501	16576	+5075	+44
Nikkei	18937	16291	-2646	-14
AORD	3152	5353	+2201	+70

Altering the algorithm slightly so that a trade represents the difference between the previous closing price and today's closing price affects the results markedly. A full 24 hour period is now accounted for and the system reflects the overall market movement during this period. These results can be seen in Table 4.3 and the amended R code can be seen in Appendix A section A.1.2.2.

Table 4.3: Naive Long System changed such that the trading period is the previous close price minus today's close.

Mkt	LongPL	L Win %	Av L PL
DAX	2649	53	1
CAC	-1667	51	0
FTSE	86	51	0
Dow	5219	53	1
Nikkei	-2712	51	-1
AORD	2229	53	1

4.2.2 Naive Reversing System

The second naive method is to reverse the previous day's movement. For example, if the market closed up the previous day the algorithm follows this by trading short for the current day (the R code for this algorithm can be see in Appendix A section A.1.2.3) . The results from this system can be seen in Table 4.4.

TABLE 4.4: Results from a naive trading system which simply trades in the opposite direction to the previous day's movement.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	947	3131	53	1	49	2
CAC	940	7810	53	1	53	4
FTSE	4284	4115	53	3	50	2
Dow	15799	6047	56	10	49	3
Nikkei	2324	20486	51	1	54	12
AORD	1264	237	53	1	48	0

For all the markets tested, this second naive system produces positive results especially for the Nikkei and CAC trading short and the Dow trading long. These results demonstrate that markets have a tendency to reverse direction each day, they move up one day then down the next. This behaviour is also observed in trending markets, and market "pull-backs" are a well-known phenomena.

4.2.3 Summary of Naive Baseline Systems

Of the two naive systems tested, the "reversing" methodology produces the best results in terms of profit and loss by quite a margin. Thus the results from the "Naive Reversing System" will be used to compare the performance of technical indicators being tested in the following sections.

4.3 Trend Detection Indicators

One of the most widely used phrases in financial trading is "the trend is your friend". Thus, most market participants are interested in identifying the start of trends, their direction and strength. In this section a variety of technical indicators that purport to assist in this important task are tested.

4.3.1 Simple Moving Average (SMA) System

One of the most popular and widely utilised technical indicators is the simple moving average (as detailed in Chapter 2 section 2.2.1.1). The effectiveness of SMA as an aid to predicting future market movements has been widely debated, with views mixed. A system based on simple moving averages is presented here, and the R code used to generate the results can be seen in Appendix A section A.1.3.1. The algorithm trades daily, opening and closing a trade each day. If the market opens above the SMA the algorithm trades long and trades short when the market opens below the SMA.

Table 4.5 lists the results from passing a variety of national index data sets (see Chapter 3 for details) to the algorithm. For each indice the algorithm is run with values of 5, 25, 50, 100 and 200 for the SMA period. In general the results are poor, especially after consideration is given to any transaction costs. The CAC and Nikkei produce negative results for long trades, the FTSE negative results across the board, and the Dow negative returns on the short side.

One aspect of a trading system of this nature worth considering is the risk/reward profile. As written in its current form the SMA algorithm has an unlimited profit potential (trades are left to run until the end of the day) and an unlimited potential loss for the same reason. Often traders employ what is known as a “stop loss”. This is a level in the market that if reached during a trade will cause the trade to close. The risk is now therefore reduced to this value while the profit is still potentially uncapped. Table 4.6 lists the results of using a stop loss with the SMA system.

The logic of the stop loss was coded as follows. Considering a long trade (the opposite holds true for trading short), where there is an expectation that the market will rise, a the stop loss would be triggered if the market fell to a certain level. Thus in the algorithm for a long trade the distance from the opening price to the low is calculated and this is compared to the stop loss value. If the open to low value exceeds the stop loss value the PL for this particular trade is set at the stop loss value, for example a loss of 100 points. One point of note is the fact that after hitting this low level the market may well recover and move upwards as originally expected. In many cases a trade that ultimately would have been profitable may be “stopped out” by the natural wax and wane of the markets. Therefore the impact of a stop loss is the balance between lost good trades and the reduction in the lost PL from losing trades. The size of the stop loss determines the impact of the two competing situations.

Figure 4.1 shows the situation in which a stop loss is beneficial. The potential large loss is reduced to the value of the stop loss value. Figure 4.2 illustrates the alternative scenario of being “Stopped Out” of an ultimately winning trade, an undesirable outcome.

Table 4.5: Results from a system based on SMA.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL	misc
DAX	2113	3278	54	1	50	2	SMA 5
DAX	1367	3427	54	1	50	2	SMA 25
DAX	779	3447	54	0	51	3	SMA 50
DAX	714	2339	54	0	51	2	SMA 100
DAX	3401	4416	55	2	52	4	SMA 200
CAC	-3952	2338	49	-2	49	1	SMA 5
CAC	-5058	1615	49	-2	49	1	SMA 25
CAC	-5323	1029	49	-3	49	1	SMA 50
CAC	-2363	3188	50	-1	50	2	SMA 100
CAC	-1219	3923	50	-1	50	3	SMA 200
FTSE	-4724	-5331	49	-2	46	-3	SMA 5
FTSE	-1013	-1940	51	0	47	-1	SMA 25
FTSE	-2226	-2769	50	-1	47	-2	SMA 50
FTSE	-889	-1692	51	0	48	-1	SMA 100
FTSE	-158	-835	52	0	49	-1	SMA 200
Dow	408	-9630	52	0	46	-6	SMA 5
Dow	1138	-9204	53	1	46	-7	SMA 25
Dow	5478	-5876	53	3	47	-4	SMA 50
Dow	2576	-8220	53	1	47	-6	SMA 100
Dow	6378	-4567	54	3	48	-4	SMA 200
Nikkei	3078	20401	51	2	54	13	SMA 5
Nikkei	-7878	10770	48	-4	52	7	SMA 25
Nikkei	-6054	11408	49	-4	52	7	SMA 50
Nikkei	-6235	8381	49	-4	52	5	SMA 100
Nikkei	-5928	6836	49	-4	52	4	SMA 200
AORD	5009	3929	55	3	51	3	SMA 5
AORD	3701	2674	54	2	50	2	SMA 25
AORD	2804	1864	54	1	50	1	SMA 50
AORD	2688	1521	54	1	50	1	SMA 100
AORD	2574	1616	54	1	51	2	SMA 200

It is the ratio of these scenarios that ultimately determines whether using a stop loss is a sound strategy.

Comparing Tables 4.5 and 4.6 it can be seen that applying the stop loss has been on the whole beneficial to the results obtained, with the exception of those from the Dow which were markedly negatively impacted. Essentially losing trades have been truncated while winning trades have been left to develop. One question that needs to be addressed is what value is appropriate for a stop loss. If the value is large the benefits of cutting losses is lost, whereas if it is too small a large number of trades will be "stopped out". Many traders use a value based on the Average True Range (see Chapter 3 section 3.2.1 for details) as this allows for the volatility of a particular market.

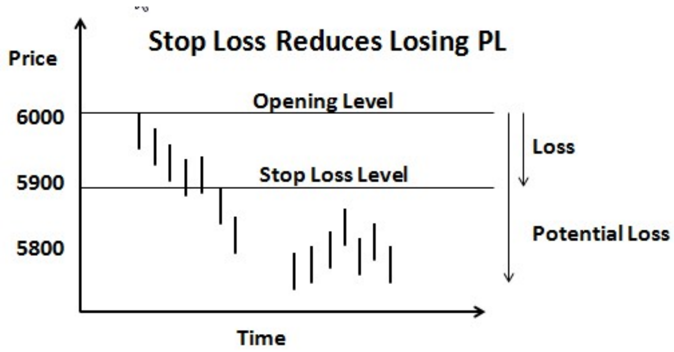

FIGURE 4.1: Situation in which using a stop loss is beneficial, with a losing PL being reduced.

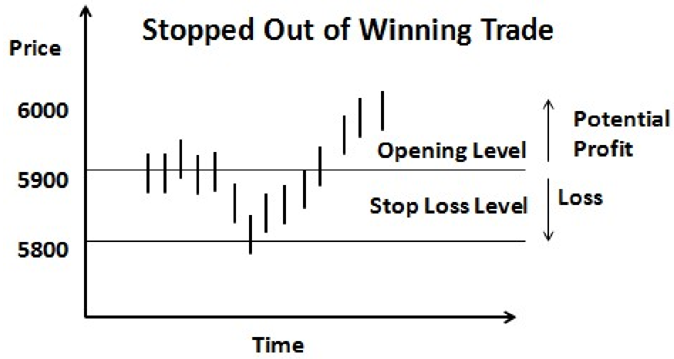

FIGURE 4.2: Situation in which using a stop loss is detrimental, being “stopped out” of an ultimately winning trade.

4.3.2 Moving Average Convergence/Divergence (MACD)

Moving Average Convergence/Divergence (MACD) is a trend following indicator, developed by Appel (2005), that is formed from the relationship of two moving averages, see Appendix B section B.1 for more details. The value of MACD itself is the difference between two exponential moving averages (EMA), a “slower” e.g. 26 day value and a “faster” e.g. 12 day value. In addition an EMA of the MACD value is calculated, which is set to 9 days in the following algorithm, which acts as a “signal” line.

The MACD is generally used two ways. Firstly, it can be used to derive the general trend of the security so that the market participant can trade with the trend. Secondly, it can be employed to identify periods when the market is “over-bought” or “over-sold” and can be expected to reverse direction (Achelis, 2014).

In order to identify the trend of a market using the MACD indicator, the relative values of the MACD itself and the signal line are used. If the value of the MACD exceeds the signal it is considered “bullish” and the market is expected to rise in price. Similarly in

Table 4.6: Results from a system based on SMA with stop loss.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL	misc
DAX	3652	6618	51	2	42	5	SMA 100
DAX	1392	5272	54	1	50	4	SMA 100
CAC	-172	5178	50	0	47	4	SMA 100
CAC	-1822	4658	50	-1	50	3	SMA 100
FTSE	1114	6303	50	1	43	5	SMA 100
FTSE	-885	1892	51	0	47	1	SMA 100
Dow	-18212	-8229	32	-9	22	-6	SMA 100
Dow	-11771	-14696	49	-6	36	-11	SMA 100
Nikkei	8258	33882	38	5	39	20	SMA 100
Nikkei	2550	25582	47	2	48	15	SMA 100
AORD	4008	3730	54	2	49	3	SMA 100
AORD	2881	2149	54	1	50	2	SMA 100

the opposite situation where the value of the signal is greater than the MACD the trend of the market is expected to be down.

Table 4.7 lists the results of using the MACD indicator in just such a way. The MACD value itself is generated using the EMA of the opening prices with values of 26 and 12 for the slow and long averages and a value of 9 days for the indicator line.

The trading algorithm splits the results into two values, days when the system expected the market to rise and days when a market decline were predicted (see Appendix A section A.1.3.2 for details of the R code used). At the start of each day if the MACD value exceeds the signal line the algorithm adds the value of the close price minus the opening price to the "Long PL" running total. Likewise in the opposite situation with the signal line greater than the MACD, the value of the open price minus the close price is added to the "Short PL". Table 4.7 lists the results of the algorithm run against a variety of national indices.

Table 4.7: Results from a system using MACD as a trend indicator.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-791	1424	53	0	48	1
CAC	-4153	2188	49	-2	49	1
FTSE	63	-839	51	0	48	0
Dow	5592	-5190	53	3	46	-3
Nikkei	-4078	14064	49	-2	52	8
AORD	2563	1569	54	1	49	1

4.3.3 Aroon Indicator

Developed by Tushar Chande, the Aroon indicator was designed to identify trending markets (Chande and Kroll, 1994). The word aroon means "dawn's early light" in Sanskrit and this indicator tries to pin point the dawning of a new trend. Essentially it is a measure of the time since the occurrence of a high/low price in a particular period. Further details can be seen in Appendix B section B.2.

TABLE 4.8: Results from a system based on the Aroon indicator.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	5308	5257	56	3	51	4
CAC	-1638	4919	50	-1	52	4
FTSE	3042	5715	52	2	51	5
Dow	12131	3811	55	7	49	3
Nikkei	-4852	12013	49	-3	52	10
AORD	3735	3540	55	2	50	3

Table 4.8 shows the results of applying the Aroon algorithm (shown in Appendix A section A.1.3.3) on the data of the national indices. The results are promising with the indicator making positive predictions in most of the markets and doing particularly well in declining markets.

TABLE 4.9: Results from a system based on the Aroon indicator with stop loss.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	5410	7465	56	3	50	6
CAC	-1224	6086	50	-1	52	5
FTSE	3091	8015	52	2	51	7
Dow	-5922	-9341	49	-3	37	-8
Nikkei	3153	22177	46	2	47	18
AORD	3786	4159	55	2	50	4

The affects of using a stop loss with the Aroon indicator was investigated and the results shown in Table 4.9. The use of a stop loss was beneficial in all cases except the Dow, in which case it had a catastrophic impact turning a winning system into a losing one. The impact of the stop loss is shown in Table 4.10 which lists the difference in PL between the original results without a stop loss and the revised ones with it.

4.4 Market Reversal Indicators

The alternative to trend detection indicators are market reversal indicators, designed to identify when a trend may be ending and the market will start to move in the opposite

Table 4.10: Impact of using stop loss with Aroon trend indicator.

Market	Long Difference	Short Difference
DAX	102	2208
CAC	414	1167
FTSE	49	2300
Dow	-18053	-13152
Nikkei	8005	10164
AORD	51	619

direction. Many traders advocate that this type of trading should be avoided and cite the old phrase “never try to catch a falling knife”. Nevertheless a variety of market reversal technical indicators are explored and their effectiveness noted.

4.4.1 Parabolic Stop-and-Reverse (SAR)

The parabolic stop-and-reverse (SAR) is a method to calculate a trailing stop. This technical indicator was developed by J. Welles Wilder and is detailed in his book New Concepts in Technical Trading Systems (Wilder, 1978). A trailing stop is related to the stop loss explored previously but differs in that it is adjusted as the market moves. The level of this of kind of stop loss is amended periodically such that it is a certain amount away from the high or low value of a market. As the the market makes new highs it is adjusted up or down if the market makes new lows. The parabolic SAR calculates the point at which a long trade would be closed and a short position entered, the assumption being that the market participant is always in the market either short or long. More details on the theory and calculations to generate the parabolic SAR can be found in Appendix B section B.3.

Table 4.11 lists the results from passing a variety of national index data sets to an algorithm using the parabolic SAR. The R code used to generate these results can be seen in See Appendix A section A.1.4.1. On the whole the results from these initial tests are very disappointing. Only three of the national indices generated positive results and only the Japanese Nikkei provided reasonable returns.

4.4.2 MACD as reversal Indicator

MACD can also be used as a reversal indicator. Recalling that the MACD is formed from the relationship of two moving averages, when the faster one moves sharply away from the slower one (i.e.the value of MACD rises) this could be an indication of an “over-bought” market and that a reversal is approaching. In this situation the trader would

TABLE 4.11: Results from a system based on the SAR indicator.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-3856	-2353	53	-2	48	-2
CAC	-5584	1034	49	-3	49	1
FTSE	-1141	-1663	51	-1	48	-1
Dow	-1301	-11112	52	-1	46	-7
Nikkei	-5767	12424	49	-3	52	8
AORD	2071	1097	53	1	49	1

place a sell trade. The opposite is true for a large negative MACD, and it is postulated that the market may well reverse upwards.

Table 4.12 shows the results of applying the algorithm shown in Appendix A section A.1.4.2 on the data of the national indices. In the algorithm the 15% and 85% quantile of the MACD value is calculated and this is used to decide on the reversal point. Once the 85% value is exceeded the algorithm predicts a reversal will occur and trades short, the opposite is true for the 15% level which triggers a long trade. Overall the results are very modest, with small positive gains being seen in 5 of the 6 national indices.

TABLE 4.12: Results from a trading system based on MACD being used as a trend reveral indicator.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	391	407	49	1	48	1
CAC	-545	2657	51	-1	55	5
FTSE	2080	1649	53	4	53	3
Dow	3882	-807	52	7	48	-2
Nikkei	199	2828	51	0	52	6
AORD	-319	-584	50	-1	49	-1

4.5 Momentum Indicators

Momentum indicators are closely related to the trend indicators introduced in section 4.3. They are concerned with trending markets but differ in that the strength of the trend is also included in the information the indicator attempts to portray.

4.5.1 Stochastic Oscillator

The stochastic indicator is one of the oldest in widespread use today having been developed by George Lane in the 1950s (Lane, 1986). It measures the relative position of a market's closing price in the range between the low and high of the period of interest. This is of interest as some market participants believe that financial markets essentially swing between price boundaries marked by where the market closes in this range (Williams, 2011). Thus markets increase until the close is at the top of this range before changing direction and moving down until it is at the bottom of the high low range.

The stochastic is usually represented by two lines %K which is the position of the price within this high low envelope described above, and %D a moving average of %K (see Appendix B section B.4 for more details). It can be used a number of ways and one popular technique is to go long when the %K crosses above %D and to go short in the opposite situation. Table 4.13 lists the results from passing a variety of national index data sets to an algorithm which uses the relative position of %K and %D to decide which way to trade. The R code used to generate these results can be seen in See Appendix A section A.1.4.3.

Table 4.13: Results from a system based on the Stochastic indicator.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-28	1673	53	0	49	1
CAC	-4540	1817	48	-3	48	1
FTSE	-73	-744	51	0	48	0
Dow	867	-9414	53	0	46	-5
Nikkei	-10591	7802	48	-6	51	5
AORD	2839	1780	54	2	49	1

The results from Table 4.13 for this system are very modest with only the Australian ORD showing positive values for both long and short trades. Adding a stop loss of 100 points increases the PL across the board except for the case of the Dow where again the stop loss has had a detrimental affect. The results from using a stochastic based system with a stop loss can be seen in Table 4.14.

Table 4.14: Results from a system based on the Stochastic indicator with a stop loss.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	1173	3889	52	1	48	2
CAC	-3493	2730	48	-2	48	2
FTSE	1640	1424	51	1	48	1
Dow	-13969	-27388	45	-8	37	-16
Nikkei	1647	17977	45	1	46	10
AORD	3028	1974	54	2	49	1

4.5.2 Rate of Change (ROC)

The Rate of Change (ROC) indicator is a simple and widely observed technical indicator. It is the difference between the current price and the price several observations ago. See Appendix B section B.5 for details. If this value is large, either positive or negative it is indicative of a strongly trending market with a lot of momentum either upwards or downwards. The R code for a trading system exploiting these ideas can be seen in Appendix A section A.1.4.4. The results can be seen in Table 4.15 which lists the results from passing a variety of national index data sets to the algorithm.

TABLE 4.15: Results from a system based on the ROC indicator.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	1026	180	50	2	50	0
CAC	952	956	53	2	51	2
FTSE	1147	1880	51	2	51	4
Dow	8517	3396	58	16	49	6
Nikkei	2971	2546	50	6	52	5
AORD	271	1325	51	1	52	2

4.6 Break-out systems

This section explores some trading systems that use a particular price as the indicator to place a trade. The first system uses the simple idea of trading when the previous day's high or low is passed. The second idea is related to the results generated in Chapter 3, where the 90% quantile for the day's minor move was calculated. The system tested here is to simply trade long or short when this point is reached in a day.

4.6.1 Daily High/Low Breakout System

Table 4.16 lists the results from a trading system based around the idea of trading after the previous day's high or low price has been breached. The R code used to generate these results can be seen in See Appendix A section A.1.5.1.

Referring to Table 4.16 we can see that this system produces good results, with the exception of the US Dow. This ties in with the data in Chapter 3 Table 3.6 which shows that the Dow only closes outside of the previous low or high price a relatively low number of times. Likewise good results are seen with the Japanese Nikkei from the breakout system and this tallies with the high proportion of the time in which it closes above or below the previous day's high or low.

Table 4.16: Results from the Daily High/Low Breakout System.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	12225	13411	55	7	54	8
CAC	3491	6955	53	2	53	4
FTSE	13189	18481	59	7	59	12
Dow	-19598	-28337	42	-11	38	-17
Nikkei	31988	43554	57	19	58	27
AORD	17225	19184	66	10	65	13

4.6.2 Break Out of 90% Quantile Level

A second system utilising the break-out concept is presented in this section. In Chapter 3 one characteristic of the markets was noted, namely that each day the market moves from its opening price to a low price and then to a high price (not necessarily in any particular order). One of these moves (O-H vs O-L) is greater than the other was termed the major move and the smaller move was called the minor move. The algorithm generating the results in this section (see Appendix A section A.1.5.2) makes a long or short trade after the market has passed the 90% quantile of the minor move. Table 4.17 lists the results from this algorithm.

Table 4.17: Results from a system that breaks out from the 90% quantile level of the day's minor move.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	7841	6371	56	6	53	4
CAC	2647	5085	54	2	52	3
FTSE	10758	15295	56	7	54	10
Dow	-30262	-34854	39	-24	37	-28
Nikkei	23606	31830	58	16	56	20
AORD	16730	19357	63	9	62	12

4.7 Candlestick Patterns

As previously noted in Chapter 2 section 2.1.5 candlestick patterns are visual representations of price movements over the course of a particular time period (often a day) in terms of the market's opening, closing, high and low prices. The pattern generated from these price markets are categorised and named depending upon the visual shape they produce. Thus candlestick patterns represent the counter forces of buyers and sellers throughout the trading period. This section analyses some well known candlestick patterns for predictive power in making trading decisions (Lu, 2014).

4.7.1 Hanging Man, Hammer, Inverted Hanging Man and Shooting Star

Four well-known patterns that are generally considered to indicate the possible end of a trend and the start of a reversal are the so-called Hanging Man, Hammer, Inverted Hanging Man and Shooting Star candlestick patterns.

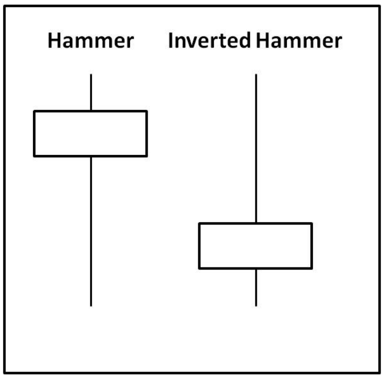

FIGURE 4.3: Hammer and Inverted Hammer candlestick patterns.

Figure 4.3 is a diagram of a Hammer and Inverted Hammer patterns. Both patterns have a small "body" (the distance between the open and close prices) and a long "shadow" (the distance between the high and low prices). In the diagrams presented here a white candlestick means the market price increased over the course of the day while a black one means the market fell. The body of the candlestick is white in this case, indicating that the market moved up (the closing price was above the opening price), although by only a small amount. Hammer and Inverted Hammer differ in that the long shadow in hammer is generated from a low price whereas the shadow of Inverted Hammer goes upwards as it is indicative of the period's high price.

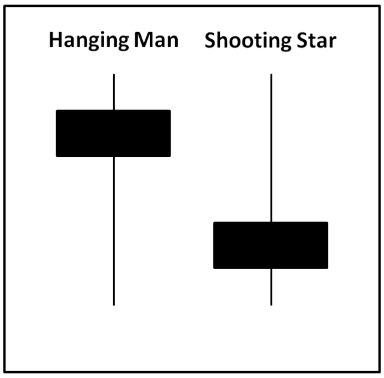

FIGURE 4.4: Hanging Man and Shooting Star candlestick patterns.

Figure 4.4 is a diagram of Hanging Man and Shooting Star, these being the opposite to Hammer and Inverted Hammer. In this case the market direction is down, albeit only by a small amount, and thus the body of the candlestick is a different colour, in this case black. Again both patterns have long shadows, the direction of which determines if the pattern is Hanging Man or Shooting Star.

FIGURE 4.5: Daily candlestick patterns from the German DAX over 22 days in April 2014 with Shooting Star and Hanging Man circled.

Both sets of patterns Hammer/Inverted Hammer and Hanging Man/Shooting Star are considered to indicate that a trend is coming to a close and a reversal could be looming. In the case of Hammer/Inverted Hammer if they are encountered during a down trend they could indicate that the selling pressure is easing and a market move to the upside could happen soon. The opposite is true for Hanging Man/Shooting Star. When these are encountered in an up trend they often indicate that the trend is ending and a reversal may occur. Figure 4.5 shows daily candlestick patterns for the German DAX over 22 days in April 2014. A Shooting Star is circled on the 6th April and a Hanging Man on the 23rd April. In each case they occur while the market is rising and in each case it reverses immediately afterwards.

In order to have a system based on candlestick patterns, the pattern itself must be identified in code. A Hammer and Hanging Man are essentially the same pattern except Hammer has a close higher than the open whereas Hanging Man represents a decline in the price. For these patterns three components are defined, the length of the upper shadow (short), the size of the body (short) and the length of the lower shadow. In the trading system that follows these were defined as:

1. Upper Shadow - the value of the day's high minus the high of the body is less than 10% the total High-Low range.
2. Body - is larger than 10% the total High-Low range.
3. Lower Shadow - the value of the day's low minus the low of the body is greater than 66% of the High-Low range.

Analysing the DAX data set running from 2000 to 2013 with 3570 observations, and using the criteria described above 35 Hammer and 48 Hanging Man patterns can be detected.

Inverted Hammer and Shooting Star are again the same pattern except in Inverted Hammer the price rose. In the later system these are defined as:

1. Upper Shadow - the value of the day's high minus the high of the body is at least 66% the total High-Low range.
2. Body - is larger than 10% the total High-Low range.
3. Lower Shadow - the value of the day's low minus the low of the body is less than 10% of the High-Low range.

Considering the DAX data set again, occurrences of these patterns are quite rare with 30 Inverted Hammers and 17 Shooting Stars in 3570 observations.

Results from a trading system based on the Hammer/Inverted Hammer can be seen in Table 4.18 and the R code in Appendix A section A.1.6.1. The algorithm simply places a buy the day after a Hammer or Inverted Hammer occur, the assumption being that these patterns indicate that the market is about to rise.

TABLE 4.18: Results from a system based on the Hammer and Inverted Hammer candlestick patterns.

Mkt	LongPL	L Win %	L Trades	Av L PL
DAX	594	53	126	5
CAC	-793	44	149	-5
FTSE	834	58	188	4
Dow	2097	59	88	24
Nikkei	-2202	48	147	-15
AORD	-809	46	236	-3

An alternative approach is to look for Hammer and Inverted Hammer patterns occurring in a down trend, in which case it could signal the end of the down trend and the start of a reversal. Table 4.19 shows the results of using the Hammer and Inverted Hammer to predict a price rise during a down trend. An aroon down value of greater than 65 (with a 20 day look back period) is used to define the down trend. The algorithm can be seen in Appendix A section A.1.6.2.

Table 4.19: Results from a system based on the Hammer and Inverted Hammer candlestick patterns occurring in a downtrend as defined by the aroon value.

Mkt	LongPL	L Win %	L Trades	Av L PL
DAX	-187	42	36	-5
CAC	-515	44	55	-9
FTSE	281	55	65	4
Dow	730	55	22	33
Nikkei	-934	48	58	-16
AORD	-614	41	77	-8

4.7.2 Engulfing Candlestick

The "Engulfing" pattern, either Bull or Bear is another widely considered candlestick pattern and is depicted in Figure 4.6. This pattern has a lower low and a higher high than the preceding candlestick and is usually interpreted as indicating a change in direction of the trend. Engulfing candlesticks can be either bullish, where the closing price is above the opening price or bearish when the market moves down.

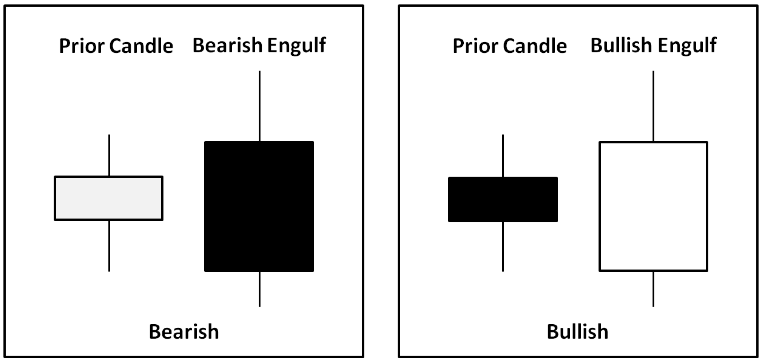

Figure 4.6: Engulfing candlestick patterns.

Table 4.20 lists the results from passing a variety of national index data sets (see Appendix A section A.1.6.3 for details) to an algorithm that buys or sells the market depending on the presence of an Engulfing pattern.

Table 4.20: Results from a system based on the Engulfing candlestick pattern.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-920	-258	44	-7	46	-2
CAC	-319	228	45	-2	50	1
FTSE	-1721	1185	51	-4	50	3
Dow	-770	-3662	48	-4	35	-28
Nikkei	-3823	-1166	37	-39	44	-11
AORD	-6	-600	53	0	46	-3

Table 4.21 lists the results from extending the algorithm such that trades are only taken in either up or down trends, as defined by the aroon indicator. The R code for the amended algorithm can be see Appendix A section A.1.6.4.

TABLE 4.21: Results from a system based on the Engulfing candlestick pattern in a trending market.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-874	-513	38	-20	43	-7
CAC	-118	-666	49	-3	30	-11
FTSE	-1217	-782	47	-8	48	-3
Dow	202	-1154	45	4	44	-11
Nikkei	-1522	-1733	38	-59	37	-32
AORD	-49	-27	53	-1	50	0

4.7.3 Doji

Doji is a well-known candlestick pattern that can appear on its own or as a component of a pattern. A Doji forms when the open and close price are similar and there is an upper and lower shadow, thus they often resemble a cross. Variations within Doji include the Dragonfly and Gravestone Doji, see Figure 4.7. In an up trend Doji (especially Gravestone) can indicate a reversal could occur and likewise in a down trend a Dragonfly could suggest an upward move is about to start.

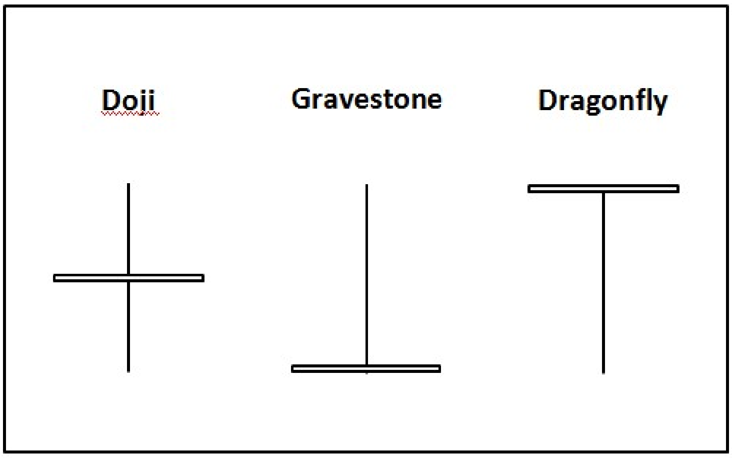

FIGURE 4.7: Doji candlestick patterns.

Table 4.22 lists the results from passing a variety of national index data sets (see Appendix A section A.1.6.5 for details) to an algorithm that buys or sells the market depending on the presence of a Doji. In an up trend, as identified by the aroon indicator, a Doji or Gravestone is used to initiate a sell and conversely in down trend a Doji or Dragonfly is used as a signal to buy.

Table 4.22: Results from a system based on the Doji candlestick pattern in a trending market.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-826	-1132	53	-8	52	-6
CAC	-747	-326	46	-6	49	-2
FTSE	-697	418	53	-8	52	3
Dow	-763	-2869	51	-5	50	-10
Nikkei	1296	-2944	55	12	45	-22
AORD	-115	195	54	-1	54	2

Chapter 5

Time Series

This chapter will explore the use of time series analysis techniques to generate models for forecasting prices in various national stock market indices. Usually, in trying to predict the future behaviour of financial markets the direction they will move, either up or down, is of more interest than the actual value itself. Thus, in this chapter predictions of the future direction as well as the actual value itself are attempted. A variety of time series models are developed using exponential smoothing, ARIMA and hybrid ARIMA methods.

5.1 Exponential Smoothing

Exponential smoothing was applied to the stock market indice data sets in order to generate predictions for the following day's closing price, so-called one step ahead forecasts. Two basic approaches and an exponential smoothing methodology were examined. The two basic methods provide a useful baseline against which to compare later models, and are the mean and drift methodologies. The mean is simply the average of the data points in the sample while the drift is equivalent to drawing a straight line between the first and last point of the sample and then extrapolating this line forward the desired number of observations.

5.1.1 Time Series Base Models

Figure 5.1 shows the two base methods, mean and drift, being applied to a data set derived from the German DAX. The models were trained on the first 3000 observations of the DAX data set and tested on the remaining ones. The actual data points being predicted in Figure 5.1 are added to the plot in Figure 5.2. Five error measures, root mean square error (RMSE), mean absolute error (MAE), mean percentage error (MPE), mean absolute percentage error (MAPE) and mean absolute scaled error (MASE) for the two methods are listed in Table 5.1. From these error measures we can see that the drift model fits the training data the best, having the smallest error values across all the measures, but that the mean model actually performs the best on the test data.

TABLE 5.1: Error measures from mean and drift models.

	RMSE	MAE	MPE	MAPE	MASE
Mean Training Set	1394	1183	-8	25	1
Mean Test Set	208	163	2	3	3
Drift Training Set	84	61	-0	1	0
Drift Test Set	302	262	-5	5	4

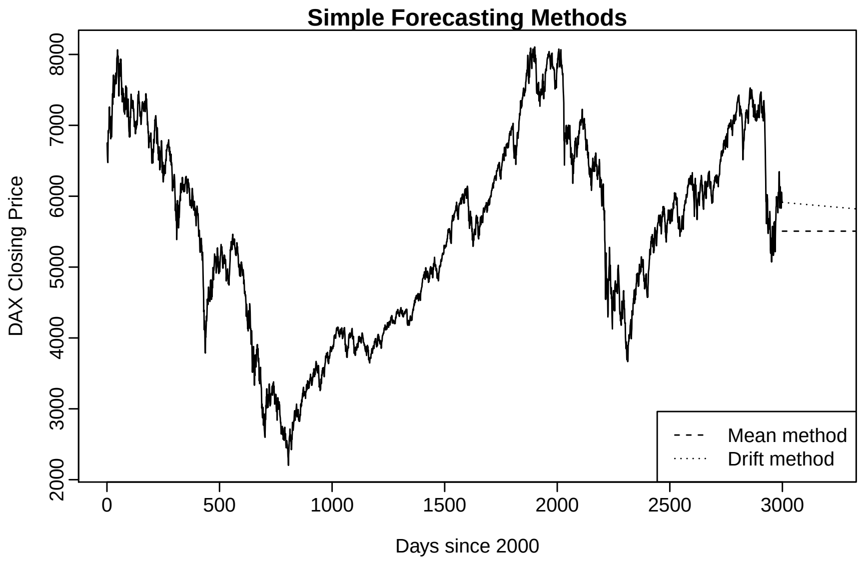

FIGURE 5.1: Forecasts generated by mean and drift methods.

Looking at Figure 5.2 it can be seen that neither the mean or the drift does a good job with the predictions for the DAX. The forecasts were based on the entire data set, treating it as a homogeneous whole. However, financial time series typically show a variety of behaviour at different periods. On occasions it is stationary and at other times trending. Thus, in the following sections, in order to generate forecasts a sliding window approach was adopted. A window of data (the last 30 observations) was used to generate a model and the one step ahead forecast, before the window was advanced one observation to the next period. In this way the model is constantly adapting and changing. Using this approach forecasts and models for use in a trading system from a mean, drift and exponential smoothing methodology were developed.

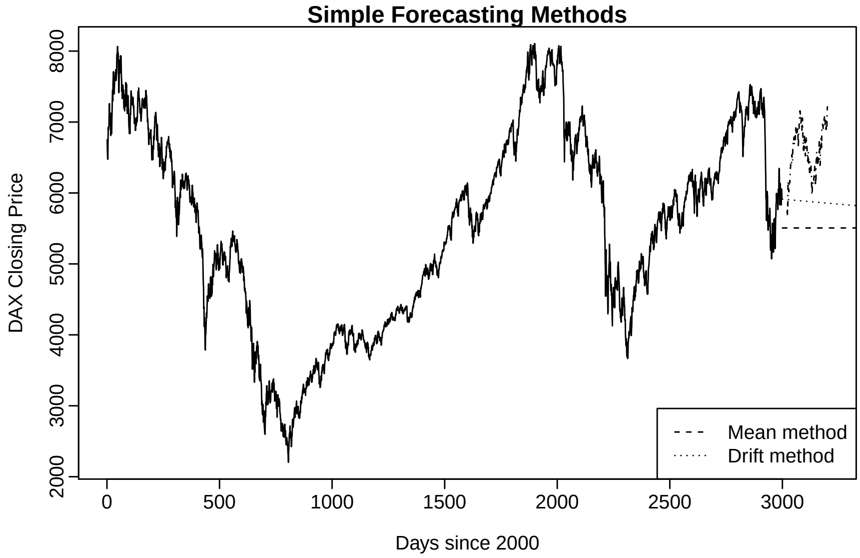

FIGURE 5.2: Forecasts generated by mean and drift methods with actual data in forecast period added.

5.1.2 Trading System Based on Mean Model

Results from a trading system using the one step ahead forecasts generated by the mean model using a moving window are listed in Table 5.2. A trading algorithm, which can be seen in Appendix A section A.2.1, used these forecasts to decide in which direction to trade. If the forecast was higher than the closing price a long trade was entered the following day and likewise if it was below the close a short trade was entered.

TABLE 5.2: Results from trading the predictions generated by a mean exponential smoothing system.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-1640	-1505	50	-1	45	-1
CAC	-1086	3553	52	-1	51	2
FTSE	1680	345	53	1	49	0
Dow	8356	-2126	54	7	46	-1
Nikkei	-32	10646	51	0	53	6
AORD	-1333	-2149	50	-1	46	-1

5.1.3 Trading System Based on Drift Model

In a similar way to the mean model of section 5.1.2 the predictions generated by the drift model were passed to the trading algorithm and the results can be seen in Table 5.3.

Table 5.3: Results from trading the predictions generated by a drift exponential smoothing system.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	2310	2445	54	1	50	2
CAC	-2422	2217	49	-1	48	2
FTSE	-518	-1853	51	0	47	-1
Dow	5416	-5066	54	3	46	-4
Nikkei	-6939	3739	48	-4	50	3
AORD	1476	660	53	1	49	1

5.1.4 Trading System Based on Exponential Smoothing Model

Using Rob J Hyndman's forecast package and the ets() function, a variety of exponential smoothing methods can be applied to sample data (Hyndman and Yeasmin, 2008). Table 5.4 lists fifteen possibilities when one combines trend and seasonality, both additive and multiplicative. In fact Hyndman extends this further by allowing the error term to be either added or multiplied against the results.

Table 5.4: Taxonomy of exponential smoothing methods.

	Seasonal Component		
Trend Component	N (None)	A (Additive)	M (Multiplicative)
N (None)	(N,N)	(N,A)	(N,M)
A (Additive)	(A,N)	(A,A)	(A,M)
Ad (Additive damped)	(Ad,N)	(Ad,A)	(Ad,M)
M (Multiplicative)	(M,N)	(M,A)	(M,M)
Md (Multiplicative damped)	(Md,N)	(Md,A)	(Md,M)

Using a sliding window approach, one step ahead forecasts were generated using the ets() function. Because a different sample of data was contained in each window, the exponential smoothing function selects the best model for each window of data independently of the data set as a whole. Thus the ets() function applies a variety of models to the windows of data including:

- ETS(A,N,N)
- ETS(M,N,N)
- ETS(M,A,N)
- ETS(A,A,N)
- ETS(A,Ad,N)

- ETS(M,Md,N)
- ETS(M,Ad,N)
- ETS(M,M,N)

The one step ahead forecasts generated from these models were once again passed to the same trading algorithm listed in Appendix A section A.2.1 and the results can be seen in Table 5.5.

TABLE 5.5: Results from trading the predictions generated by a moving window exponential smoothing system.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-2029	-1894	53	-1	47	-1
CAC	-266	4048	52	0	51	2
FTSE	3866	2531	53	3	50	2
Dow	12901	2419	57	8	50	2
Nikkei	-2741	7937	49	-2	51	5
AORD	645	-171	52	0	48	0

5.2 ARIMA Models

The use of Auto-Regressive Integrated Moving Average (ARIMA) models, see section 2.2.4 for details, was explored in order to forecast future prices for financial markets. The process of fitting an ARIMA model to a time series is quite challenging and involves the following general steps:

1. Plot the data to get a general feel for the time series and to establish if it is stationary.
2. Stabilize any variance in the data with a transformation process such as the Box-Cox method.
3. ARIMA models work with stationary data, so if necessary, take differences of the data until it is stationary.
4. Examine the auto-correlation and partial auto-correlation (ACF/PACF) plots in order to determine if an AR(p) or MA(q) model is appropriate.
5. Test the chosen model(s), using the AICc to determine if a better model is available.

6. Check the residuals from the best model by plotting the ACF, and doing a portmanteau test on them. If the results from these tests do not look like white noise, a modified model may be required.

7. Finally, once the residuals have a similar pattern to white noise, the model can be used to generate forecasts.

In recent years automatic forecasting algorithms have become available and are widely used (Hyndman and Yeasmin, 2008). These are necessary in a variety of circumstances, especially when organisations are faced with the need to repeatedly carry out a large number of forecasts and the human effort required renders manual means impractical. The auto.arima() function found in R's "forecast" package is an example of an automatic algorithm for ARIMA models. This function automates steps 3, 4, and 5 of those outlined previously, in the general steps required for ARIMA modelling. In the following sections, the general steps are used to generate an ARIMA model manually, and then the automatic algorithm is utilised to build one.

5.3 Manual Generation of ARIMA Models

5.3.1 Data Exploration

The first step, as always is to explore the data. Figure 5.3 shows the UK's FTSE 100 index between the years 2000 to 2013. Over this time period the series has shown strong trends to move up and down and a uniform variance. Because the time series is non-stationary it will need to be transformed into a stationary series before ARIMA modelling can be undertaken.

5.3.2 Adjusting for non-uniform variance and non-stationariness

The variance within the FTSE time series is relatively uniform and thus this data set doesn't need stabilizing with regard to this. If it did, a Box-Cox transformation could be used. However, over this time period the FTSE 100 exhibits marked non-stationariness and requires adjusting accordingly. One such technique to make a data set stationary is differencing. Instead of using the actual observations the differences between two adjacent points are used and this is known as the first difference. If the data set still isn't stationary the difference between consecutive points in the differenced data set can used, this is the difference of the differences and is known as the second difference. Figure 5.4 shows the FTSE data set after the first differences have been taken. The resulting data set is now stationary.

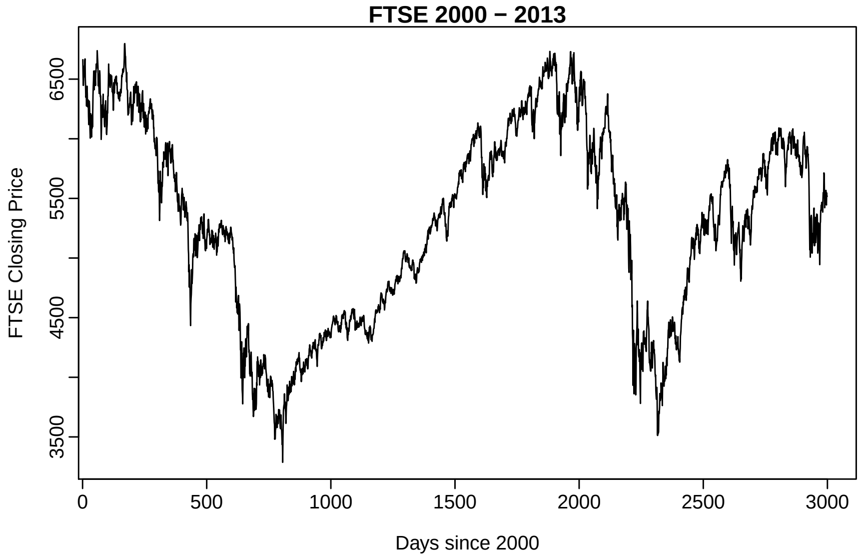

Figure 5.3: UK's FTSE 100 index between the years 2000 to 2013.

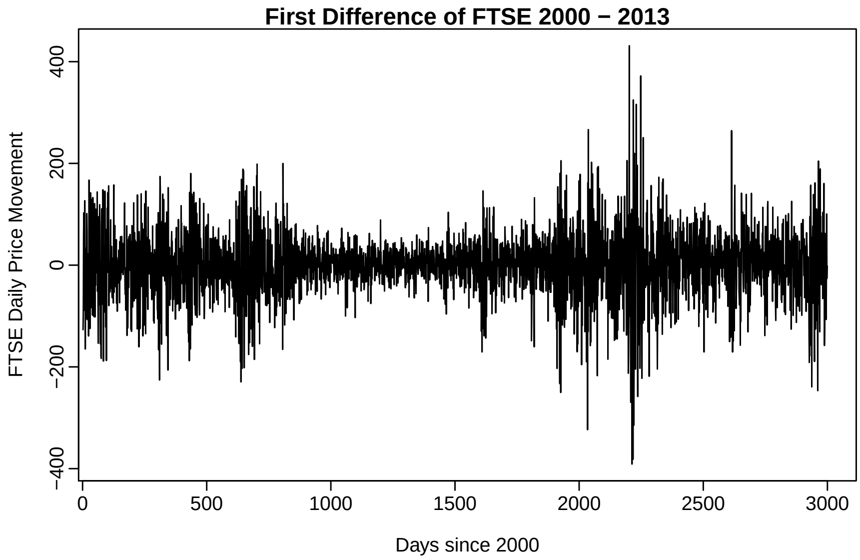

Figure 5.4: First difference of FTSE 100 between the years 2000 to 2013.

5.3.3 Examine ACF/PACF

With a stationary data set, the next stage is to investigate the auto-correlation and partial auto-correlation (ACF/PACF) plots in order to help in the model selection process (see section 2.2.5 for details of ACF and PACF). The ACF and PACF for the FTSE data set can be seen in Figures 5.5 and 5.6.

If ultimately the ARIMA model is of the form ARIMA(p,d,0) or ARIMA(0,d,q) then the ACF and PACF plots are useful in helping to define values for p or q. In the event that both p and q are positive, the ACF and PACF are not helpful in deducing the values for p and q. An ARIMA(p,d,0) model may be appropriate if the ACF and PACF plots of the stationary data exhibit an exponentially decaying pattern in the ACF and a large spike at lag p in PACF plot. Conversely an ARIMA(0,d,q) model may be appropriate if the PACF is decaying exponentially and there is there is a significant spike in the ACF plot at lag q. Considering the ACF and PACF plots in Figures 5.5 and 5.6, neither of the two patterns are observed and thus an ARIMA model where both p and q are positive is likely.

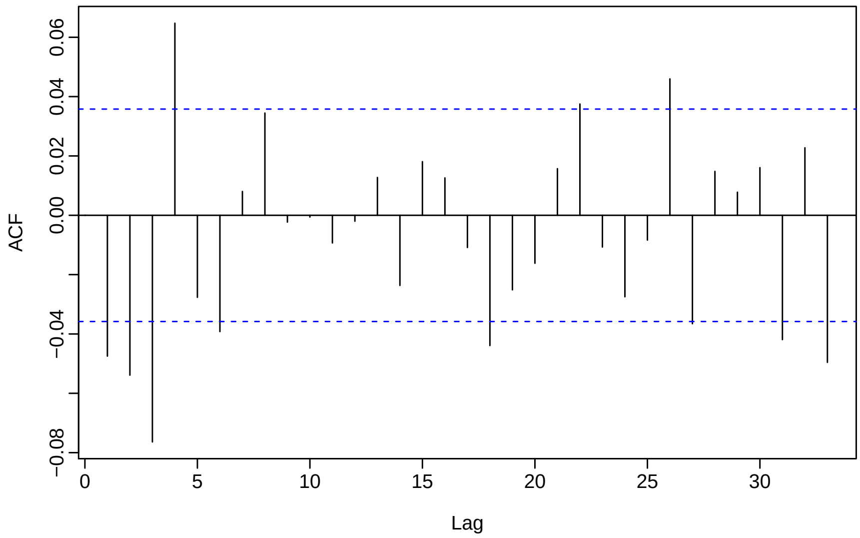

FIGURE 5.5: Auto-correlation plot of differenced data from FTSE 100 between the years 2000 to 2013.

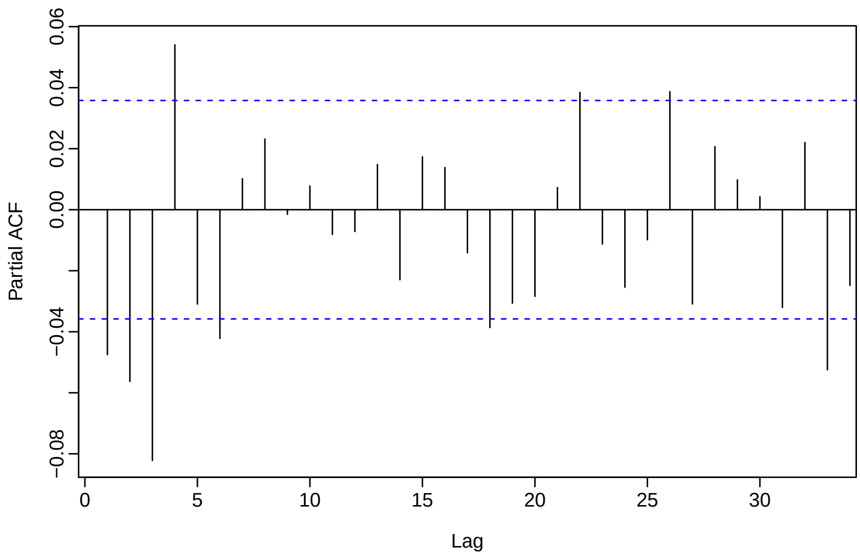

FIGURE 5.6: Partial auto-correlation plot of differenced data from FTSE 100 between the years 2000 to 2013.

5.3.4 Try the chosen model(s)

The next step is to try the chosen model along with a few viable alternatives. Akaike's Information Criterion (AIC) and Bayesian Information Criterion (BIC) are useful for determining the optimum order of an ARIMA model, and are typically used as a measure of how well the model fits the data. AIC can be given by:

$$AIC = -2log(L) + 2(p + q + k + 1)$$

where:
L is the likelihood of the data and $k = 1$ if $c \neq 0$ and $k = 0$ if $c \neq 0$, the last term in parentheses is the number of parameters in the model.

For ARIMA models, the corrected AIC can be written as:

$$AIC_c = AIC + \frac{2(p + q + k + 1)(p + q + k + 2)}{T - p - q - k - 2}$$

The Bayesian Information Criterion can be expressed as:

$$BIC = AIC + log(T)(p + q + k + 1)$$

Table 5.6 shows the AIC, AICc and BIC accuracy measures for a selection of ARIMA models applied to the FTSE data set. On all three measures the ARIMA(2,1,3) model has the lowest value.

Table 5.6: AIC, AICc and BIC results from alternative ARIMA models.

Model	AIC	AICc	BIC
ARIMA(3,1,1)	33598.5	33598.5	33628.5
ARIMA(3,1,2)	33594.6	33594.6	33630.6
ARIMA(3,1,3)	33596.1	33596.1	33638.1
ARIMA(2,1,1)	33616.4	33616.4	33640.4
ARIMA(2,1,2)	33618.1	33618.1	33648.1
ARIMA(2,1,3)	33594.1	33594.1	33630.1

5.3.5 Model Residuals

A so-called residual is the difference between an observation and its forecast. In forecasting a time series, residuals are calculated from a one-step forecast. A one-step forecast is based on all observations from the start of the series until the previous observation to which the forecast applies to. Thus the number of data points used to calculate the one-step forecast increases as the forecast proceeds through the time series. An alternative is cross-sectional forecasting which uses all the points in the data set except the observation being predicted.

Knowledge of the residuals from the application of a model is important in establishing the validity of the model. There are two essential and two valuable properties that can be established by inspecting the model residuals. A good method of forecasting will produce a model in which the residuals are uncorrelated and have a zero mean. If a forecasting method doesn't comply with these two properties it can be improved upon. Correlation in residuals means that information is present in them that the model has missed and a non-zero mean is evidence of bias in the forecast. Adjusting for bias is straight forward, the mean value observed in the residuals can simply be added to all forecasts. Looking at Figure 5.7 it can be seen that the mean of the residuals is close to zero and this model doesn't have any bias. Figure 5.8 is the plot of the residuals of the ARIMA model applied to the FTSE data set. The lower order lags are all within the confidence boundaries and is indicative of a good model.

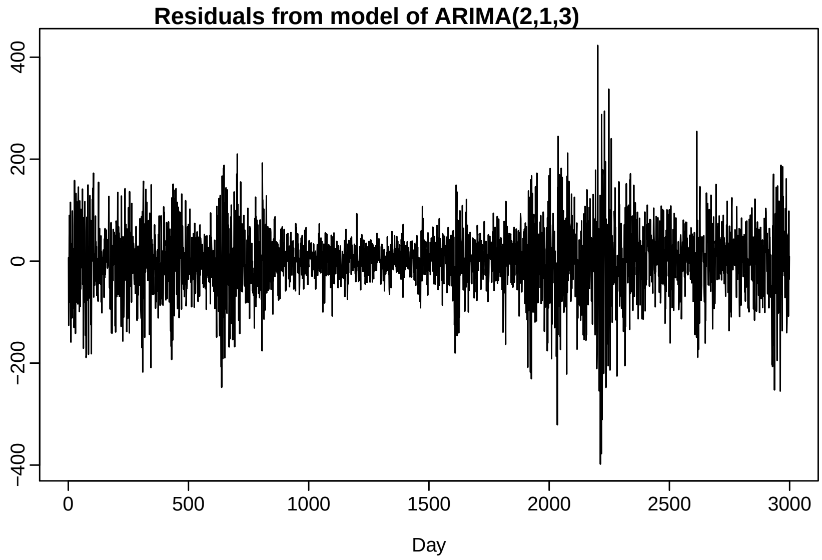

Figure 5.7: The residuals after applying the ARIMA(2,1,3) model to the FTSE data set.

Two additional properties of the residuals that are desirable, though not necessary, are constant variance and normal distribution. If these two conditions are met, the calculation of the prediction interval in the forecast step is easier. From Figure 5.7 it can be seen that the residuals have relatively constant variance and from Figure 5.9, a histogram of the residuals, it can be seen that they are normally distributed.

Consideration of the ACF plots provides evidence for auto-correlation. However a more formal approach is to consider auto-correlation values together as a group as opposed to individually. The Box-Ljung portmanteau test is just one such approach and Table 5.7

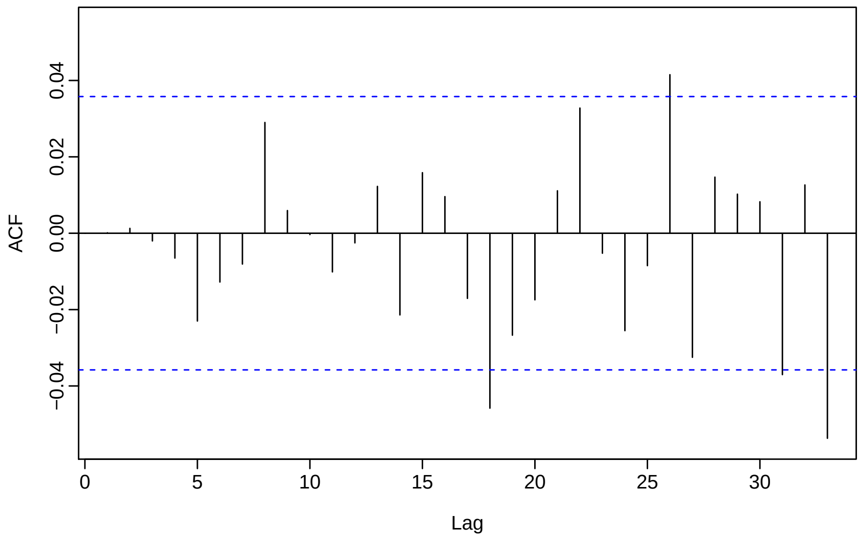

FIGURE 5.8: ACF plot of the residuals after applying the ARIMA(2,1,3) model to the FTSE data set.

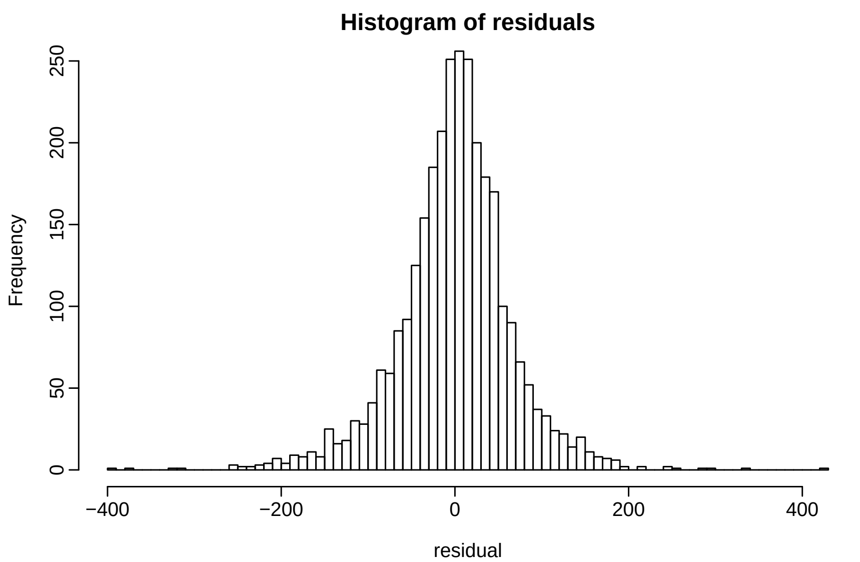

FIGURE 5.9: Histogram of the residuals after applying the ARIMA(2,1,3) model to the FTSE data set.

lists the results of the Box-Ljung portmanteau test being applied to the residuals of the ARIMA(2,1,3) model. A large p-value is indicative of white noise and is the desirable situation for a good ARIMA model. Taking all the evidence together the ARIMA(2,1,3) model appears a good option for the FTSE data set.

TABLE 5.7: Box Ljung test of FTSE 100 ARIMA model residuals.

	p-value	x-squared	df
ARIMA(2,1,3)	0.2328	20	24

5.3.6 Calculate forecast

Finally, after developing a model that meets the previous criteria a forecast can be generated. Table 5.8 shows the one-step forecast produced when the ARIMA(2,1,3) model developed in the previous section is applied to the FTSE data set.

TABLE 5.8: One step ahead forecast for FTSE 100 generated from ARIMA(2,1,3) model.

Date	Open	High	Low	Close	Forecast
20/12/2013	6585	6617	6577	6607	6560
23/12/2013	6607	6679	6606	6679	6598
24/12/2013	6679	6712	6672	6694	6666
27/12/2013	6694	6754	6694	6751	6692
30/12/2013	6751	6768	6718	6731	6743
31/12/2013	6731	6757	6731	6749	6730

5.4 Automatic Generation of ARIMA Models

As explained previously the automatic ARIMA modelling algorithm in the R forecast package, auto.arima(), automates steps 3 to 5 in the general steps used in the modelling process as outlined in section 5.2. The function uses a variation of the Hyndman and Khandakar algorithm which obtains an ARIMA model by the minimisation of the AICc and combination with unit root tests. KPSS tests are used to establish the number of differences, d, required to get a stationary time series. The p and q values are then obtained by choosing the model that minimises the AICc for the differenced data.

The results of passing the indice data sets to the auto.arima() function can be seen in Table 5.9. For the FTSE data set the automatic procedure selects the ARIMA(2,1,3) as being the most appropriate, which matches the conclusion of the work from the manual model selection process described earlier in section 5.3.

TABLE 5.9: ARIMA models chosen to forecast future values in the national indice data sets.

Market	ARIMA Model
DAX	ARIMA(3,1,3)
CAC	ARIMA(2,1,3)
FTSE	ARIMA(2,1,3)
Dow	ARIMA(1,1,2)
Nikkei	ARIMA(2,1,3)
AORD	ARIMA(1,1,0)

5.5 Trading the ARIMA Models

Having developed forecasts based on ARIMA models these can be passed into a trading system. Two ideas are presented here, in the first the previous closing price is compared against the prediction and if it is lower than the forecast a long trade is entered. This first system will be referred to as System 1. In the second algorithm the current forecast is compared with the previous prediction. When the previous forecast value is lower than the current prediction the system trades long. This algorithm will be referred to as System 2.

The data sets containing fourteen years of indice data was divided into a training set, containing the first ten years of data, and a test set holding the remaining four years worth of data. Models were trained on the training sets before being applied to the unseen data in the test sets.

5.5.1 System 1 - Close Price vs Forecast

Using the ARIMA models listed in Table 5.9 a series of amended data sets were generated by applying the models to the national indice data sets used throughout this study. The amended data sets contained the original Date, Open, High, Low and Close attributes plus a new one called Forecast, in a similar manner to the data seen in Table 5.8. Table 5.10 are results produced from passing the newly generated data sets to the algorithm listed in Appendix A section A.2.2. This system uses the relative position of the close price and the forecast to determine the direction of the trade. If the forecast is higher than the close a long trade is made the next day and when the prediction is lower than the close price a short trade is made.

TABLE 5.10: Results from trading System 1 using the forecasts generated by the ARIMA models.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-1285	-2522	50	-5	42	-9
CAC	872	167	57	3	51	1
FTSE	990	-249	53	4	46	-1
Dow	1539	-3356	56	7	47	-11
Nikkei	4268	3071	53	21	49	13
AORD	635	-247	55	2	49	-1

5.5.2 System 2 - Forecast vs Previous Forecast

Table 5.11 lists the results from passing the amended indice data sets with the forecasts generated from the auto.arima() function, described in the previous section, to the System 2 algorithm. The R code of this system can be seen in Appendix A section A.2.3. System 2 uses the relative values of the forecasts themselves to decide which direction to trade. If the prediction is higher than the previous day's prediction a long trade is initiated the following day and in the opposite circumstances when the previous forecast is higher than the current forecast a short trade is made.

Table 5.11: Results from trading System 2 using the forecasts generated by the ARIMA models.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	733	-505	55	3	46	-2
CAC	545	-80	53	2	47	0
FTSE	941	-383	54	3	46	-2
Dow	2598	-2221	55	9	46	-10
Nikkei	179	-916	50	1	47	-4
AORD	811	-117	53	3	46	0

5.6 Hybrid ARIMA Models

A hybrid ARIMA model is one in which the moving averages of a stationary data set (possibly a non-stationary data set that has been differenced) are combined with data mining learners other than regression. A variety of combinations were tried, with a combination of the three previous closing prices and differences and the four previous moving averages providing good results. Possible learners include k-Nearest Neighbour (k-NN) algorithms, Artificial Neural Networks (ANN) and Support Vector Machines (SVM). RapidMiner, an open source data mining tool is a powerful solution for building hybrid ARIMA models. Figure 5.10 shows the RapidMiner process used to generate hybrid ARIMA models. The Validation operator in the model below can hold a variety of learners depending upon the task and data types involved. The various components in Figure 5.10 are as follows:

- Read CSV - reads in the appropriate data set.
- Select Attribute (1) - selects the attribute that will be processed in the following steps.

- Rename - renames the attribute selected in Select Attribute (1) to "attr1" which is then used in the rest of the steps. This component is used to make it easy to change the attribute without having to rename all the subsequent steps.
- Moving Average - calculates a moving average of the time series (see section 2.2.1.1 for details.) This provides the q in ARIMA(p,d,q) models.
- Differentiate - calculates the difference in the time series and provides the d in ARIMA(p,d,q) models.
- Lag - creates lag variables which are values of the attribute (the attribute itself, the moving average or the difference value) at earlier points in the time series. The three previous closing prices and differences and the four previous moving averages were typically used.
- Select Attribute (2) - selects the attributes that will be passed to the validation block, including the number of previous moving averages and differences.
- Set Role - sets an attribute as the label to be predicted.

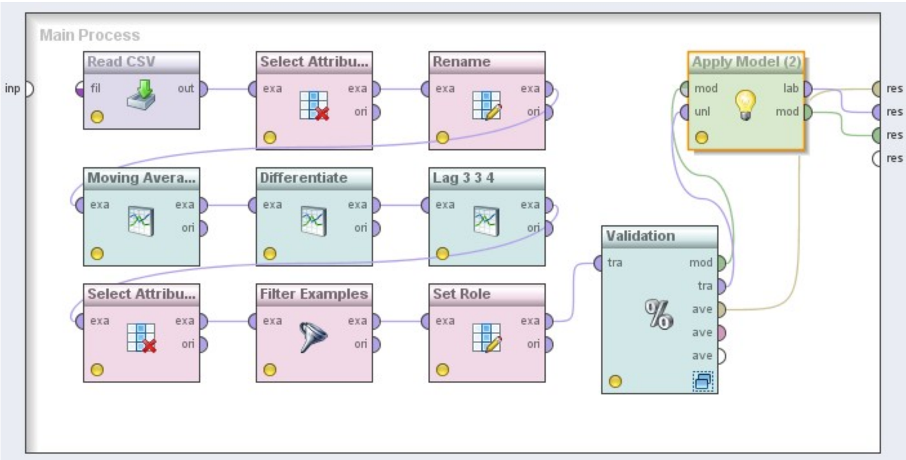

Figure 5.10: Rapid Miner hybrid ARIMA process.

Figure 5.11 shows the cross-validation operator of the hybrid ARIMA Rapid miner process. This operator can hold alternative learners other than the standard regression operator found in ARIMA models. In the diagram there is an Artificial Neural Network (ANN) operator shown, other options include k-Nearest Neighbour (k-NN) and Support Vector Machine (SVM) operators.

Using the hybrid ARIMA methods two types of predictions were carried out. Firstly, in section 5.7 the actual closing price of the following observation was predicted, which is a numeric label. This was done by combining ANN and k-NN with ARIMA, as these learners can output numeric class labels. Secondly, in section 5.8 the more general

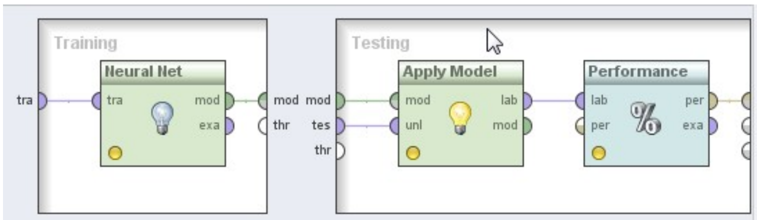

FIGURE 5.11: Rapid Miner cross-validation operator within the hybrid ARIMA process.

situation of whether the market increased or fell in value over the course of the day, was predicted. In this case there are two options, either the market moved up or down. For this second situation a combination of ARIMA with ANN, k-NN and SVM was used, as all three can forecast a binary label.

5.7 Predicting Closing Price

As mentioned previously ARIMA and hybrid ARIMA models were used to predict either the value of the one-step ahead close price or the binary value of whether the market moved up or down. In this section the ability of hybrid ARIMA models to forecast the future price of financial markets (as opposed to the general direction up or down) is explored.

5.7.1 ARIMA/Artificial Neural Networks (ANN)

An ARIMA/ANN method was used to generate predictions for the closing price of the indice data sets under study. As described in section 5.6, previous values of the label (closing price), along with past values for the difference and moving averages are passed to an ANN learner within a rapid miner process. Artificial Neural Networks can be used to predict categorical or numeric labels, but can only use numeric inputs. They are slow to learn, but their strength lies in their ability to model complex patterns. Essentially they are designed to mimic the human brain by the use of layers of neurons with connections between them. The neurons are found in input and output layers as well as optional hidden layers. The model is constructed by applying weights to the connections which are used to calculate the label.

Rapid miner implements ANN as a feed-forward neural network trained by back propagation. Feed-forward refers to the fact that information only moves forward from the input nodes to the output neurons. Back propagation is an algorithm in which errors are fed back into the system so that the algorithm can adjust the connection weights between nodes until the error converges to some state. Typically the best results were obtained from using a small learning rate and momentum and using a hidden layer of neurons.

For each data set applying the hybrid model produces a new one-step forecast attribute, in a similar manner to the forecast attribute seen in Table 5.8, which can be used in the System 1 and 2 algorithms previously introduced in section 5.5. Table 5.12 are the results generated by passing the output of the ARIMA/ANN models to trading System 1 (which compares the previous closing price with the current forecast).

Table 5.12: Results from passing closing price predictions from hybrid ARIMA/ANN model to System 1.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-446	-645	53	-1	46	-2
CAC	532	1527	55	2	50	2
FTSE	625	-624	51	1	45	-2
Dow	2846	-3979	55	5	45	-9
Nikkei	913	2039	54	2	55	4
AORD	-3036	-371	54	-5	51	-1

Table 5.13 are the results of passing the output of the ARIMA/ANN models to trading System 2, which compares the value of the current forecast with the previous one.

Table 5.13: Results from passing closing price predictions from hybrid ARIMA/ANN model to System 2.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	3283	3110	56	6	49	7
CAC	-1832	-816	49	-3	46	-2
FTSE	1092	-182	52	2	48	0
Dow	3829	-2942	54	7	43	-7
Nikkei	-4485	-3229	48	-9	50	-7
AORD	-2783	-137	51	-5	47	0

5.7.2 ARIMA/k-Nearest Neighbour (k-NN)

An ARIMA/k-NN method was used to generate predictions for the closing price of the indice data sets. The k-Nearest Neighbour algorithm operates by comparing the current set of attributes to others in the data set and finding ones that are "close" to it. Closeness is usually defined by a distance measure, such as Euclidean, after the attributes have been stored in a n-dimensional pattern space. The k closest neighbours are selected and the most common class of these used to classify the label. In a similar manner to the ANN modelling of section 5.7.1, lag variables of the close price (label attribute to be forecast), moving average and differences of the close price were passed to a k-NN learner in Rapid Miner in order to predict the one-step ahead forecast.

The k-NN learner has only a small number of parameters such as k, the number of close neighbours to be considered and the types of measures to be used. Fifteen was found to be a good value for k and either euclidean or cosine similarity a good choice for the distance measure. Table 5.14 shows the results of passing data sets containing forecasts generated with the hybrid ARIMA/k-NN to trading System 1. Table 5.15 shows the results of passing data sets containing forecasts generated with the hybrid ARIMA/k-NN to trading System 2.

Table 5.14: Results from passing closing price predictions from hybrid ARIMA/k-NN model to System 1.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	12	-174	52	0	45	0
CAC	-684	312	52	-1	50	1
FTSE	699	-550	54	1	50	-1
Dow	4436	-2389	57	11	46	-4
Nikkei	-66	1060	48	0	50	2
AORD	497	3162	53	1	50	7

Table 5.15: Results from passing closing price predictions from hybrid ARIMA/k-NN model to System 2.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	489	731	54	1	47	2
CAC	-1458	-441	49	-3	46	-1
FTSE	-388	-1662	50	-1	45	-4
Dow	2969	-3411	54	5	43	-9
Nikkei	-2916	-1660	47	-6	49	-4
AORD	-3449	-804	52	-6	48	-2

5.8 Predicting Up or Down - Categorical Label

In this section the ability of hybrid ARIMA models to forecast whether a financial market will rise or fall is investigated. A categorical attribute taking values “U” and “D”, representing whether the market moved up (“U”) or down (“D”) was introduced into the indice data sets depending upon which way the market moved that day. the final six observations of the FTSE data sat with the binary label added can be seen in Table 5.16. Hybrid ARIMA models using ANN, k-NN and SVM learners, were used to generate one-step ahead forecasts for this categorical label.

Table 5.16: FTSE 100 data set with "U" and "D" label introduced.

Date	Open	High	Low	Close	U/D
20/12/2013	6585	6617	6577	6607	U
23/12/2013	6607	6679	6606	6679	U
24/12/2013	6679	6712	6672	6694	U
27/12/2013	6694	6754	6694	6751	U
30/12/2013	6751	6768	6718	6731	D
31/12/2013	6731	6757	6731	6749	U

5.8.1 ARIMA/Artificial Neural Networks (ANN)

Lag values from the moving average of the closing price, the difference values between consecutive observations and the closing price were passed to an ANN learner to create one-step ahead forecasts of the "U" and "D" values. The R code for a trading system using the forecasts from a hybrid model can be seen in Appendix A section A.2.4. The algorithm simply uses the prediction from the hybrid ARIMA model to decide whether to trade long or short. Table 5.17 lists the results from the trading system using the hybrid ARIMA/ANN forecasts.

Table 5.17: Results from a trading system using the forecast of categorical label "U/D" from hybrid ARIMA/ANN model.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	199	0	53	0	NaN	NaN
CAC	0	996	NaN	NaN	49	1
FTSE	1400	151	53	2	49	0
Dow	5218	-1607	55	6	45	-10
Nikkei	234	1360	53	12	51	1
AORD	-2712	-47	52	-3	25	-12

5.8.2 ARIMA/k-Nearest Neighbour (k-NN)

In a similar manner to section 5.8.1, an ARIMA/k-NN model was also employed in an attempt to predict the categorical label indicating whether the financial markets would move up or down. The forecasts produced from these hybrid models were also applied to the trading algorithms listed in A section A.2.4. Table 5.18 lists the trading results from this combination.

As the results from Table 5.18 were good, the algorithm was re-run but this time a stop loss was introduced. A stop loss of 100 points was applied to all the markets and the amended results can be seen in Table 5.19. In a similar manner as encountered

Table 5.18: Results from a trading system using the forecast of categorical label "U/D" from hybrid ARIMA/k-NN model.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-1553	-1752	54	-3	47	-4
CAC	270	1265	52	1	49	2
FTSE	1764	515	55	3	50	1
Dow	3211	-3614	55	6	44	-8
Nikkei	2707	3834	50	6	52	8
AORD	748	3413	54	1	50	7

previously, the use of the stop loss was beneficial for all the markets except the Dow in which case it had a large detrimental affect.

Table 5.19: Results from a trading system with a stop loss using the forecast of categorical label "U/D" from hybrid ARIMA/k-NN model.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-430	-1444	53	-1	47	-4
CAC	203	1326	52	0	49	2
FTSE	1919	526	54	4	50	1
Dow	5475	-1922	51	11	42	-4
Nikkei	4021	2804	48	9	49	6
AORD	570	3424	53	1	50	7

5.8.3 ARIMA/Support Vector Machine (SVM)

ARIMA was also married with a SVM learner in order to predict the categorical value, "U" or "D". SVMs are linear classifiers that can only operate with binary class labels and thus assign observations in a data set to one of two options, in this case "U" or "D".

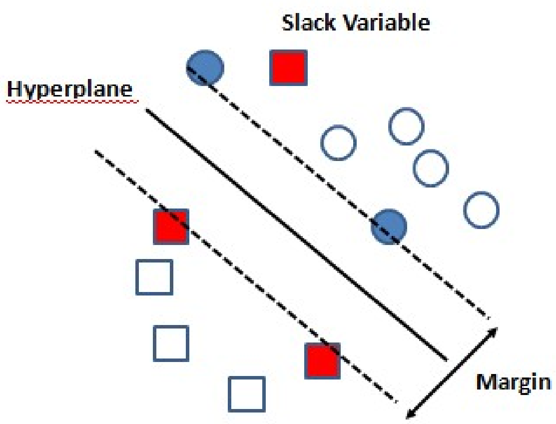

Figure 5.12: SVM margins and slack variables.

Support vector machines operate by finding the straight line that dissects a data set such that the points close to the dividing line are as far away from it as possible. In high dimensional data sets the dividing lines can be considered as a set of hyperplanes in a high dimensional space. The algorithm works by choosing the hyperplanes that are furthest from the boundary data points, which are known as the support vectors. This intuitively would provide the best classification between two sets of data points. Observations that are classified into the wrong class are known as slack variables, the number of which increases as width of the dividing margin increases.

Figure 5.12 illustrates the margin calculated from a SVM between two classes of observations, and the presence of a wrongly classified slack variable. Therefore in SVMs there is a trade off between a narrow margin with a small number of slack margins, which could represent over-fitting, and a wider margin with more wrongly classified data points. In the Rapid Miner SVM operator, the parameter “C” is a cost function, that applies a cost to slack variables. If this parameter is set to a high value there is a high cost applied to the creation of a slack variable and therefore the model produces tight margins and risks over-fitting the model. For low values of C, models with wider margins with more slack variables are produced.

In order to allow for non-linear boundaries between classes non-linear kernel functions are used in SVM. These map the original data set into a data set with a larger number of dimensions which can then be split with a linear function.

Table 5.20 lists the results of passing forecasts made using hybrid ARIMA/SVM models to the trading algorithm listed in A section A.2.4. Typically small values were used for the cost parameter in Rapid Miner and a radial non-linear kernel function applied.

Table 5.20: Results from a trading system using the forecast of categorical label ”U/D” from hybrid ARIMA/SVM model.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	-123	-322	53	0	46	-1
CAC	-140	855	54	0	50	1
FTSE	2115	866	54	5	49	1
Dow	2138	-4686	57	10	45	-6
Nikkei	9	1135	48	0	51	2
AORD	-2364	301	53	-4	49	1

Chapter 6

Analysis

6.1 Introduction

In chapters 4 and 5 a wide variety of analytical techniques were applied to a range of time series data sets. In Chapter 4 a number of trading algorithms were developed based on technical analysis indicators, with the intention of automating the decision of whether to buy or sell a market. For comparison purposes, two simple so called "naive" systems were explored to act as a baseline against which the technical analysis indicators could be compared. The technical indicators were grouped together into their general area of applicability, namely trend detection indicators, reversal, momentum and candlestick indicators.

Chapter 5 continued the exploration of financial time series through the use of exponential smoothing, ARIMA and hybrid ARIMA techniques. The generated models were used to create one-step ahead forecasts which were then added to the original data sets. These amended data sets were then fed into trading algorithms which used the forecast values to make buying decisions.

6.2 Technical Analysis

6.2.1 Baseline Systems

Initially two simple, naive systems were explored so that they could be used as a baseline against which the developed predictive models could be compared. These systems were the Naive Long System which mirrors a buy and hold strategy and a Naive Reversing System which simply trades in the opposite direction to the previous days market movement.

The first baseline system tried was the Naive Long system in which a market buy is placed each day and is similar to the so-called “Buy and Hold” technique. The assumption here is that the market rises over time and if an investor simply holds a security it will eventually generate a profit. The total profit is the price at the start, in this case the data set started in 2000, subtracted from the price at the end of the period, which in this case was the end of 2013.

The first iteration of the algorithm placed a buy at the start of the trading session and closed it at the end and thus the system was out of the market overnight. This resulted in significant discrepancies from the returns expected from a buy and hold system. Table 6.1 lists the expected returns from a Buy and Hold system over this period, with the Difference column being the profit or loss over the time.

Table 6.1: Returns from a “Buy and Hold” technique.

Date	Start 2000	End 2013	Difference	% Change
DAX	6961	9552	+2591	+37
CAC	6024	4250	-1774	-29
FTSE	6930	6749	-181	+-3
Dow	11501	16576	+5075	+44
Nikkei	18937	16291	-2646	-14
AORD	3152	5353	+2201	+70

From simply trading long during market hours the DAX generated a loss as opposed to the 2591 profit expected, likewise the CAC showed a much larger loss than expected and the Nikkei resulted in a large loss when a small loss was expected. The Dow, FTSE and AORD were similar to the expected values. The discrepancies in the returns between the trading algorithm and a Buy and Hold approach was due to the fact that the algorithm opened and closed trades each day as opposed to simply opening the trade and waiting several years. This first algorithm was simply trading within market hours, approximately 8am to 5pm local time, and was not in the market for the full 24 hours of the day.

Changing the algorithm such that the trades ran from the market close time until the close time of the following day and thus covered the full 24 hour period resulted in system results that matched those expected from a buy and hold approach. Clearly the discrepancies from the first algorithm were due to the relative amounts the markets moved during the day as opposed to during the “out of hours” trading. There is a slight bias for the markets to move upwards overnight and over the course of the study (14 years) adds up to significant values.

The second naive system was termed the “Naive Reversing” system and simply places a trade today in the opposite direction from the previous day. This idea produced reasonable returns, with every market making money. From these results it can be concluded that the markets have a tendency to ”flip flop” and reverse back on themselves, and the phenomena of market reverses is well understood. This second concept produced far better results than the first and was thus used as the primary basis of comparison for the algorithms and systems developed from technical indicators and time series methods. For convenience the results from this system are reproduced in Table 6.2.

Table 6.2: Results from a naive trading system which simply trades in the opposite direction to the previous day’s movement.

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	947	3131	53	1	49	2
CAC	940	7810	53	1	53	4
FTSE	4284	4115	53	3	50	2
Dow	15799	6047	56	10	49	3
Nikkei	2324	20486	51	1	54	12
AORD	1264	237	53	1	48	0

6.2.2 Trend Detection

The first group of the technical analysis indicators studied were the trend detection indicators. Identification of trend direction and strength is very important in the world of financial trading and one of the most widely encountered phrases is ”the trend is your friend”, as most authorities advocate trading in the direction of the trend. (In fact on a recent webinar it was claimed that 80% of all money made is made trading in the direction of the trend.) Well known indicators that purport to assist the trader in identifying trends are the simple moving average (SMA), the moving average convergence/divergence indicator (MACD) and the Aroon indicator.

The use of simple moving average is wide-spread in the financial markets. Market participants track moving averages or even more than one and make a decision which way to trade based on the position of the current price relative to it. Popular values to use in the SMA are 25, 50 and 200 for the look back period. The results of a trading system based on SMA is presented in Table 4.5 of Chapter 4. The algorithm places a buy trade if the current price is above the SMA and a sell trade if it is below it.

Results are mixed from using the SMA, with some markets producing positive results and some ending in losses. The German DAX and Australian AORD produce positive results across all the SMA values with returns from trading short (predicting the market will decline) doing best. The Japanese Nikkei and French CAC display different behaviour in that all the SMA values tried produce negative results in trying to predict long trades but positive returns when attempting to predict short trades. The UK's FTSE 100 is different again, producing negative results across the board. Finally, the Dow produces positive results for trades on the long side but losses for trading short.

In an attempt to improve the returns from the trading system a stop loss was introduced. Comparing Tables 4.5 and 4.6 of Chapter 4 it can be seen that applying the stop loss has been on the whole beneficial to the results obtained, with the exception of those from the Dow which were negatively impacted. Essentially losing trades have been truncated while winning trades have been left to develop. This general pattern of a stop loss being beneficial to all the markets except the US Dow was seen multiple times with the systems tested.

The second trend detection indicator explored was the Moving Average Convergence Divergence (MACD) indicator, full details of which can be found in section B.1 of Appendix B. The MACD can generally be used two ways, as a trend detection indicator and as an over-bought/over-sold indicator in which case traders use it to identify potential market reversals. In this section the indicator was used as a trend detector and the results from a system based on the MACD indicator can be seen in Table 4.7 in Chapter 4. The algorithm trades long when the value of MACD is greater than the value of the signal line, see Appendix A section A.1.3.2 for details of the R code used. The results are not very impressive, only the Nikkei producing reasonable profits, although they wouldn't beat the baseline Naive Reversing system.

The final trend detection indicator examined was Aroon. This indicator measures the number of intervals since the previous high or low within a certain time window. The algorithms presented here used a time window of 20 days. If the current day was the highest price in the last 20 days trading, the indicator would take a value of 100 and for each following day that doesn't make a new high the indicator falls by 5 (100 divided by the lag period which is 20). Thus if the highest price was four days ago the AroonUp value would be 80. The opposite situation occurs with regard to the low price. A value of 70 or above for the AroonUp is indicative of a upward trending market and likewise a value of 70 and above for AroonDn suggests a falling market.

The results from an algorithm using the Aroon indicator can be seen in Table 4.8 of Chapter 4. Overall the results are encouraging with the DAX, FTSE, Dow and AORD all producing positive returns for both long and short trades, while the CAC and Nikkei are positive when trading short. Table 6.3 lists the values derived from the Aroon system with those from the baseline Reversing system (see Chapter 4 section 4.2.2) subtracted. Because the Aroon system doesn't execute trades each day it only makes sense to compare the average daily returns as opposed to the total returns. As can be seen from Table 6.3, compared with the baseline system for some markets the Aroon indicator outperforms the baseline while for others it is worse, notably the Nikkei. Considering the second column in Table 6.3, "Diff in Mean Long PL" only the DAX and AORD outperformed the baseline reversing system in producing long winning trades. Alternatively, the system based on the Aroon indicator was superior in predicting winning short trades for all the markets except the Nikkei, as seen in the third column "Diff in Mean Short PL".

Table 6.3: Results from baseline Reversing System subtracted from Aroon results.

Mkt	Diff in Mean Long PL	Diff in Mean Short PL
DAX	2	2
CAC	-2	0
FTSE	-1	3
Dow	-3	0
Nikkei	-4	-2
AORD	1	3

The trading system based on the Aroon indicator was re-run with a stop loss value of 100. Overall the use of a stop loss improves the returns, with the exception of the Dow. One again using a stop loss with the Dow shows very marked negative impacts on profits. These results can be seen in 4.9 of Chapter 4.

6.2.3 Market Reversal Indicators

In this section two indicators that purport to assist in identifying market reversals are examined, namely the Parabolic Stop-and-Reverse (SAR) and the Moving Average Convergence Divergence (MACD) used as an over-bought/over-sold indicator.

The first market reversal indicator explored was the Parabolic Stop-and-Reverse (SAR), an indicator initially developed for traders who were always in the market with either long or short position. The SAR is used to judge when the position should be reversed from long to short or vice versa. The trading algorithm reported here trades each day (i.e opens a trade at the start of the trading session and closes it out at the end) and makes a decision regarding the direction of the trade based on the SAR indicator. If the market opening is above the SAR a long trade is initiated and vice versa if the market is below the SAR value.

The results from the trading system based on the SAR can be seen in Table 4.11 of Chapter 4 and are very poor. Only the Nikkei trading short produces reasonable results, but these are much worse than the baseline Naive Reversing method introduced previously.

As previously mentioned the MACD indicator can be used as a market reversal indicator. Once the MACD value reaches its extreme values, the market is considered over-bought or over-sold and likely to reverse back on itself. The trading algorithm using this concept expects a market reversal once the MACD crosses above the 85% quantile (of the MACD range) or below the 15% quantile. Short trades are initiated once the MACD crosses above the 85% quantile value and short trades once it has passed below the 15% quantile. The results from this trading system can be seen in Table 4.12 and are very unimpressive being inferior to the baseline method.

Both market reversal indicators resulted in poor results. However, the trading horizon was short at just one day. Given the difficult task at hand, trying to pin point the reversal of a financial market perhaps a longer trading horizon may have helped. This might have allowed leeway in the timing and caught any reversals that may have occurred two or three days later. This is an area for future research.

6.2.4 Momentum Indicators

A third category of technical indicators are the momentum indicators, which are related to the trend detection indicators (Menkhoff et al., 2012). Two such indicators are studied here, the stochastic and the rate of change (ROC). The stochastic oscillator is one of the oldest and most widely used of the technical indicators. It measures the percentage position the current close is in relation to the high low range of the period of interest. For example the current close could be 80% of the way between the low and high of the last 10 days. Thus it has conceptual similarities to the Aroon indicator. The stochastic is usually represented by two lines %K which is the position of the price within this high low envelope described above, and %D a moving average of %K (see Appendix B section B.4 for more details).

The trading algorithm utilising the stochastic initiates long trades when %K is above %D and short trades when %K is below %D. Results from an algorithm implementing these ideas can be seen in Table 4.13 in Chapter 4. The results of this system are poor being significantly worse than the baseline Naive Reversing system.

The second momentum indicator is the Rate Of Change (ROC) indicator, and this is simply the difference between the current price and a price a certain number of days previously. If this value is positive the market is considered to be trending up and the larger the value the greater the trending momentum. The results from an algorithm using these ideas is presented in Table 4.13 of Chapter 4. The results are positive but very modest and inferior to the baseline Reversing system.

6.2.5 Breakout systems

The fourth area of technical analysis explored the idea of trade signals being generated by a particular value from the previous day, so-called breakout systems. Two particular values are used as the trigger price for a trade, the previous day's high/low or the 90% quantile of the minor move (see section 3.2.5 of Chapter 3).

The first idea explored was to use the previous time period's high or low price as a trigger for a buy or sell. If the current day's high price exceeded the previous day's high price a long trade would be made and in a similar manner if today's low price is lower than previous day's low a short trade is initiated. Results from using the previous day's high price or low price as a trigger to trade long or short can be seen in Table 4.16. Generally the results are very good with the exception of the Dow. These results can be linked to the data exploratory work shown in Table 3.6 of section 3.2.3. The best returns were generated in the Nikkei, a market which had the highest number of times closing outside the previous day's high or low. Conversely, the lowest ranked market in terms of closing outside yesterday's high low range was the Dow, and this was the one market that produced negative results in the break-out system. Table 6.4 lists the returns from the high low breakout system with the profits from the baseline Naive Reversing system subtracted. As can be seen, with the exception of the Dow, the method out-performs the baseline system markedly.

Table 6.4: Results from Daily High/Low Breakout System compared with Naive Reversing System

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	6894	3240	3	5	4	2
CAC	1707	-2725	1	1	-1	-1
FTSE	6474	11180	3	4	4	8
Dow	-46061	-40901	-17	-34	-12	-31
Nikkei	21282	11344	7	15	2	8
AORD	15466	19120	10	8	14	12

The second break-out system used the minor fluctuation 90% quantile value as the trigger level to trade long or short. Once the market moved above this level a long trade was made or if the market moved below this level a short trade was executed. Overall this methodology produces good results with the exception of the Dow and CAC. Table 6.5

lists the difference in results between this breakout methodology and the baseline Naive Reversing system.

Table 6.5: Results 90% Quantile level Breakout System compared with Naive Reversing System

Mkt	LongPL	ShortPL	L Win %	Av L PL	S Win %	Av S PL
DAX	6894	3240	3	5	4	2
CAC	1707	-2725	1	1	-1	-1
FTSE	6474	11180	3	4	4	8
Dow	-46061	-40901	-17	-34	-12	-31
Nikkei	21282	11344	7	15	2	8
AORD	15466	19120	10	8	14	12

6.2.6 Candlestick Patterns

A number of so-called candlestick patterns were explored for predictive properties in financial markets. The patterns tested were essentially market reversal patterns. Firstly, Hammer and Inverted Hammer were considered. When these patterns occur it is considered a sign that the market will move upwards, especially when they are encountered in a down trend, thus reversing direction. Table 4.18 lists the results from placing buy trades after all occurrences of either pattern while Table 4.19 shows the results from initiating buy trades when these patterns occur in trending markets. The Aroon indicator detailed in section B.2 of Appendix B was used to determine if the market was in a trending phase. Overall the results from using the Hammer or Inverted Hammer candlestick pattern to predict market movement was poor. Only the Dow and FTSE showed positive results, although the per trade profit from the Dow was good. Another consideration is the small number of times in which these patterns occur, only 22 trades in the 14 years of the Dow data were made. Clearly these visual patterns are quite subjective and in reality a trader would use judgement as to whether the pattern constituted a Hammer or not. However, in developing an algorithm to recognise and trade them no such latitude is possible and thus the number of trades taken by the algorithms is likely to be less than in reality.

The next pattern tested was the Engulfing pattern. This pattern occurs when a candlestick has a lower low and a higher high than the previous day's candlestick, it engulfs it. The presence of this pattern is supposed to indicate that the market will change direction. The results of a trading algorithm that trades long or short depending upon the presence of an Engulfing candlestick can be seen in Table 4.20. The results shown in Table 4.21 are similar to Table 4.20 except trades are only taken if the market is trending, with the Aroon indicator used to determine if the market is in a trending phase.

The results from both algorithms were very poor, with most markets showing negative results.

The final pattern tested was the Doji, one of the best known candlestick patterns. Again the presence of this pattern in a trending market is supposed to give warning to the market participants that a reversal may be imminent. Table 4.22 shows the results of a trading system that uses the presence of a Doji in a trending market to initiate a trade. Again the results are very poor with mostly negative returns.

None of the candlestick patterns produced good results in the trading algorithms and time frames used. Given the subjective nature of what constitutes the particular pattern and the context in which it occurs it is difficult to generate trading systems through computer programming. Further, it may be beneficial to alter the trading time frame to a longer period and perhaps combine these patterns with additional technical analysis indicators, and is an area suitable for further work.

6.3 Time Series Analysis

Exponential smoothing, ARIMA and hybrid ARIMA models were used to generate forecasts of the closing prices and the more general situation of whether the market would rise or fall. In modelling the more general situation of market direction, a categorical label was employed. The categorical label used “U” to represent occasions when the market prices increased and “D” for when it decreased in value.

6.3.1 Exponential Smoothing

Exponential smoothing was used to make one-step ahead forecasts for the indice data sets. Initially two base systems were explored in order that they could be used as a baseline against which later results could be compared. The two methods generated predictions using a mean method in which the forecast was simply the average of the sample and a drift method which is the extrapolation of a straight line between the first and last point in a data sample. For all the forecasts generated in this section a moving window approach was adopted. A window of sample data points was used, typically the last 30 observations, for which a forecast was generated using one of the methods. This window was then advanced one observation forward and the forecast for this data set calculated and the process repeated for the entire data set.

Once the forecasts were generated they were passed to a trading algorithm which made decisions regarding whether to trade long or short based on the value of the forecast. If the forecast was higher than the previous closing price a long trade was initiated or if it was lower a short trade was made with the expectation that the market would fall. Results from the forecasts generated from the mean method can be seen in Table 5.2 while results from the drift method can be seen in Table 5.3. Both systems produced poor results and wouldn't generate enough profits to offset any transaction costs that would be incurred in the real world.

Next exponential smoothing was used to generate forecasts. Allowing for trend, seasonality and whether there are additive or multiplicative affects, a variety of models may be used to generate predictions. The ets() function of the forecast package in R can create models from all the available possibilities. Once again using a moving window approach, one-step ahead forecasts were created for the indice data sets. As the window moved through the data, different models were selected as being the best fit and these used to generate the forecasts. Having generated the predictions they were passed to the same trading algorithm as used for the mean and drift models, and the results can be seen in Table 5.5.

On the whole the results from the models generated from exponential smoothing are quite poor, with only the Dow producing reasonable, though very modest results. In comparison to the base systems developed from mean and drift methods, the adaptive exponential smoothing concept produces higher profits in general when all the results are considered, but there are much better results when other time series techniques or technical analysis methods are considered.

6.3.2 ARIMA Models

The auto.arima() function of the R forecast package was used to assist in generating ARIMA models for the national indice data sets used in this study. For convenience the models selected are listed in Table 6.6.

Table 6.6: ARIMA models chosen to forecast future values in the national indice data sets.

Market	ARIMA Model
DAX	ARIMA(3,1,3)
CAC	ARIMA(2,1,3)
FTSE	ARIMA(2,1,3)
Dow	ARIMA(1,1,2)
Nikkei	ARIMA(2,1,3)
AORD	ARIMA(1,1,0)

The one-step forecasts generated from these models were then used in two trading systems. In the first algorithm the decision to trade long or short was dependant upon on the relative values of the previous close price and the forecast. If the forecast was higher than the close price a long trade was entered the next day in the expectation that the market would rise towards the prediction. The opposite situation was expected for when the forecast was lower than the close price. The R code for this first algorithm can be seen in Appendix A section A.2.2 and is labelled system 1.

The second trading algorithm used the relative values of the predictions themselves in order to decide whether to trade long or short. If the current forecast was higher than the previous one a long trade was entered the following day and vice versa. The R code for this second algorithm can be seen in Appendix A section A.2.3 and is labelled system 2.

The results from both systems were poor. The difference in mean PL per trade between the first system based on the auto.arima models (previous close in comparison to forecast) and the mean PL for the Naive Reversing system from section 4.2.2 Chapter 4 (the best of the baseline systems) can be seen in Table 6.7. Most of the results are worse than the naive baseline system except for the French CAC and US Dow when trading long.

TABLE 6.7: Mean Long/Short PL from system using predictions from ARIMA models with the results from the Naive Reverse system subtracted.

Mkt	Diff in Mean Long PL	Diff in Mean Short PL
DAX	-6	-11
CAC	2	-3
FTSE	1	-3
Dow	-3	-14
Nikkei	20	1
AORD	1	-1

6.3.3 ARIMA Hybrids - Predicting Closing Price

Hybrid ARIMA models in which Artificial Neural Networks and k-Nearest Neighbour algorithms were used instead of regression in the ARIMA algorithm to predict the closing prices of financial markets, see Chapter 5 section 5.6 for details.

6.3.3.1 ARIMA/Artificial Neural Networks (ANN)

Overall the use of the forecasts from the models generated from hybrid ARIMA/ANN algorithms to create trading systems was not very successful. The results from passing the indice data sets augmented with a forecast attribute generated by the hybrid ARIMA models can be seen in Tables 5.12 and 5.13 of Chapter 5. System 1 compares the price of the forecast with the price of the previous close and in the event that the prediction is higher than the previous closing price a long trade is entered. The opposite is true when the forecast is lower than the closing price and a short trade is made. System 2 is similar but compares the forecast with the last forecast. In the event that the current prediction is greater than the previous one a long trade is initiated.

Considering the results in Tables 5.12 and 5.13 it can be seen that System 1 outperforms System 2 quite markedly. Even so, the results are quite modest across most of the indices and especially poor for the DAX. The results prove inferior to the baseline Naive Reversing System introduced in 4.2.2 Chapter 4 as shown in Table 6.8.

Table 6.8: Results from a trading system based on forecasts of closing price generated by the ARIMA/ANN model compared to baseline Naive Reversing methodology.

Mkt	Diff in Mean Long PL	Diff in Mean Short PL
DAX	-2	-4
CAC	1	-2
FTSE	-2	-4
Dow	-5	-12
Nikkei	1	-8
AORD	-6	-1

6.3.3.2 ARIMA/k-Nearest Neighbour (k-NN)

An alternative to the ARIMA/ANN methodology is to replace ANN with a k-Nearest Neighbour learner, that looks for neighbouring data points that are similar or close (usually defined by some measure distance) to it. Results from using the forecasts generated in the two trading systems introduced in section 5.5 can be seen in Tables 5.14 and 5.15.

The results from System 1 are similar to those from the hybrid ARIMA/ANN of the previous section. In comparison to the baseline Naive Reversing approach they are likewise inferior, although for trading long they produce similar winning percentages. Table 6.9 lists the difference in results between those generated with System 1 and the ARIMA/k-NN models and the baseline system.

6.3.4 ARIMA Hybrids - Predicting Up Down with Categorical Label

An alternative to forecasting the closing price of a financial market is to predict the general direction it will move in the short term either up or down. To this end an

Table 6.9: Results from a system using forecasts from a ARIMA/k-NN model with the results of the Naive Reversing System subtracted.

Mkt	L Win %	Av L PL	S Win %	Av S PL
DAX	-1	-1	-4	-2
CAC	-1	-2	-3	-3
FTSE	1	-2	0	-3
Dow	1	1	-3	-7
Nikkei	-3	-1	-4	-10
AORD	0	0	2	7

additional categorical label to indicate whether the market increased or fell in value over the course of the day was introduced into the data sets. This new attribute had the value "U" if the market increased and "D" if it decreased. Hybrid ARIMA models were then employed to predict this label.

6.3.4.1 ARIMA/Artificial Neural Networks (ANN)

The first methodology employed was to combine ARIMA with Artificial Neural Networks (ANN) in order to generate a forecast of the categorical label that indicated whether the market increased in value or fell over the course of the day. Once the forecast was generated and added to the data set in the form of a new attribute it was passed to a trading algorithm which based the decision whether to trade long or short on the forecast generated. The R code for the trading algorithm can be seen in Appendix A section A.2.4 and the results generated in Table 5.17. Overall the results were poor and inferior to the baseline system used for comparison.

6.3.4.2 ARIMA/k-Nearest Neighbour (k-NN)

Replacing the ANN learner from the previous section with a k-NN method resulted in similar results. Again the returns from using the hybrid methodology was inferior to the baseline methodology. Table 5.18 lists the results of passing the forecasts from this combination to the trading algorithm in Appendix A section A.2.4. Table 6.10 lists the difference in results between using this hybrid ARIMA approach and the usual baseline returns.

6.3.4.3 ARIMA/Support Vector Machine (SVM)

Finally, the ARIMA methodology was coupled with a Support Vector Machine (SVM) learner. The SVM is appropriate here because the categorical label being forecast is

Table 6.10: Results from Naive Reversing System subtracted from results generated from predicting Up/Down categorical label using ARIMA/k-NN.

Mkt	L Win %	Av L PL	S Win %	Av S PL
DAX	1	-4	-2	-6
CAC	-1	0	-4	-2
FTSE	2	0	0	-1
Dow	-2	-9	-5	-17
Nikkei	-1	5	-2	-4
AORD	1	0	2	7

binary, there are only the values "U" or "D" representing up (the market increased) and down (the market fell) respectively. The results from passing the generated forecast to the same trading algorithm as that used in the previous section and described in Appendix A section A.2.4 can be seen in Table 5.20 of Chapter 5. Overall the results were poor with the exception of the FTSE and the CAC and Nikkei trading long.

Table 6.11 lists the difference in results between using this hybrid ARIMA approach and the usual baseline returns.

Table 6.11: Results from Naive Reversing System subtracted from results generated from predicting Up/Down categorical label using ARIMA/SVM.

Mkt	L Win %	Av L PL	S Win %	Av S PL
DAX	0	-1	-3	-3
CAC	1	-1	-3	-3
FTSE	1	2	-1	-1
Dow	1	0	-4	-9
Nikkei	-3	-1	-3	-10
AORD	0	-5	1	1

6.4 Conclusion

This study delved into the issue of whether financial markets can be predicted with the use of technical analysis or times series modelling. To this end a wide variety of technical analysis indicators were explored along with a range of time series models. One aspect of technical analysis worth noting is that opinion is divided as to its value, with many voices in academia being critical of it (Kuang et al., 2014, Fang et al., 2014, Bajgrowicz and Scaillet, 2012). Having stated that, it is also a fact that it is widely, almost ubiquitously, used by participants of the financial markets (Taylor and Allen, 1992). The widespread use of technical analysis includes the large body of amateur traders, as may be expected, as well as highly educated professionals. Indeed it was reported by Menkhoff (2010) that most fund managers who were polled, sophisticated professionals in this arena, employ technical analysis.

The results from the technical analysis presented in Chapter 4 were grouped into the general area in which they purport to operate. These area were trend detection, market reversals, momentum indicators, break-out patterns and candlestick patterns. Of all the technical indicators explored in this study the break-out and trend detection methods seem to hold the most promise. This is in agreement with Brock et al. (1992) who also found merit in these techniques. The use of Moving Average Convergence/Divergence (MACD) is one area where the results of this study divert from the literature. The results from Chapter 4 section 4.3.2 are quite poor whereas Ulku and Prodan (2013) report profitable returns from using MACD to forecast national stock market indices.

Another area in which conflicting results are found in the literature is the use of candlestick patterns. Marshall et al. (2006) reported on unsuccessful studies using a range of candlestick patterns to assist in forecasting the Dow Jones Industrial Average. This is contrasted by Lu (2014) who reports on the successful use candlestick charts in finding profitable trades in the Taiwan stock market. Results from this study reflect the findings of Marshall et al. (2006), in that predictions from the candlestick patterns produced poor returns. However, an element of caution is required before dismissing the usefulness of candlestick patterns. Being visual patterns they need an element of subjectivity to be applied to their use and this is difficult to achieve in the rigid arena of computer programming.

The second area of study concentrated on established time series modelling techniques with a firmer basis in academia. This included the use of exponential smoothing, ARIMA and hybrid ARIMA techniques. The Holt-Winters method of exponential smoothing defines various aspects of a time series and applies a smoothing function to each (Winters, 1960). Thus models can be defined to account for stationary, trending or data sets with a seasonal element. Financial markets exhibit a variety of behaviour at different periods and thus one single smoothing model is not appropriate. The approach adopted in this study was to use a moving window technique. Thus as the window moved through the data set an appropriate model could be selected and applied for each subset of observations. The results generated from this exponential moving methodology were poor, barely producing returns better than a baseline approach that simply used the average of the data sample.

Another widely used time series modelling technique is Auto-Regressive Integrated Moving Average (ARIMA). ARIMA models were built for the data sets under study and the forecasts generated used in trading algorithms, see Chapter 5 section 5.5 for details. The results were poor with the systems unable to produce a profit.

One major limitation in the ARIMA model is the need for the relationship in the data to be linear. The use of hybrid ARIMA methods is an attempt to overcome this limitation, coupling ARIMA with a different learner such as Artificial Neural Networks (ANN), k-Nearest Neighbour (k-NN) or Support Vector Machines (SVM). There are many reports in the literature of good results obtained from the use of these hybrid methods. Wang et al. (2012), Khashei et al. (2009) and Zhang (2003) reported promising results from the use of ARIMA/ANN and Pai and Lin (2005) good results from ARIMA/SVM. This study showed only very modest returns from the use of ARIMA/ANN, ARIMA/k-NN or ARIMA/SVM.

6.4.1 Research question revisited

This study had the aim of answering the following question:

"Can the use of technical indicators or time series analysis help to predict the future direction and movement of financial markets?"

Essentially there are two elements to this question, the ability of technical analysis to aid in predicting future market movements and the usefulness of time series modelling techniques to predict market movements. Considering technical analysis first the results overall are quite poor and most techniques tried produced poor results, rarely beating the baseline system of simply trading in the opposite direction to the previous day, especially when this concept is married with a stop loss. Overall the breakout idea and the aroon indicator produced the best results from the technical analysis indicators. Of the times series modelling techniques explored the hybrid models produced the best results, although these were very modest.

6.4.2 Future Work

Restraints of time and resources restricted this study to a selection of technical analysis indicators and times series models. Further, the data selects were limited to national indices of daily data. There are a large number of other technical indicators that can be explored and many other financial markets including individual stocks, commodities such as gold and oil and a wide range of currency pairs. The techniques reported in this study were compared in isolation. In many situations traders use combinations of methods and this could form the basis of future work. Given the huge number of possible combinations careful consideration would have to be given to the selection of the techniques to use.

The time frame of the data used is another important consideration. This report used daily open, high, low and close data and used trading algorithms based around the day. Data sets in different times frames, particularly shorter ones such as minute or even tick data (the most granular of financial data) could be explored. Trading algorithms that hold trades for days or weeks could also be tried and this idea maybe particularly interesting for techniques such as candlestick patterns for which the predictive benefit may not be immediate.

Appendix A

R Code

A.1 Chapter 4

The R code used to generate the results and tables in Chapter 4 is shown in listing A.1.1. This is followed by the individual files containing the algorithms used in the chapter.

A.1.1 Chapter 4 Results Generation

```
# Chapter 4
setwd("D:/Allan/DropBox/MSc/Dissertation/Thesis/RCode")

# libraries to include
library(xtable)
library(TTR)
library(candlesticks)

# files to include
source("../RCode//Utils.R")
source("../RCode//NaiveLongSystem.R")
source("../RCode//NaiveLongSystem2.R")
source("../RCode//NaiveReversePrev.R")
source("../RCode//SMA_sys.R")
source("../RCode//MACD_XO.R")
source("../RCode//Aroon.R")
source("../RCode//SAR.R")
source("../RCode//Stoch.R")
source("../RCode//ROC.R")
source("../RCode//MACD_OB.R")
source("../RCode//Bout_sys_2.R")
source("../RCode//Quant90_sys.R")
source("../RCode//Candle_Hammer.R")
source("../RCode//Candle_Hammer_aroon.R")
source("../RCode//Candle_Engulf.R")
source("../RCode//Candle_Engulf_aroon.R")
```

```
source("../RCode//Candle_Doji_aroon.R")

fil <- c("../Data/Dax_2000_d.csv",
         "../Data/CAC_2000_d.csv",
         "../Data/F100_2000_d.csv",
         "../Data/Dow_2000_d.csv",
         "../Data/N225_2000_d.csv",
         "../Data/Oz_2000.csv")

df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11)) # to hold results
std6 <- c(1,3,4,5,7,8,10)
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
NaiveRev <- run_NaiveReversePrev(fil, 0, nm)
misc_col <- 11

# ----------------------------------------------
# ---------  1. Naive Long Base System   --------

run_NaiveLongSystem <- function(fil, SLoss, nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    a <- NaiveLongSystem(Mkt, SLoss, nm[i])
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res1 <- run_NaiveLongSystem(fil,0,nm)
res1[misc_col] <- 'Naive Long'

# for summary results
total_res <- res1

# produce latex table
dat <- res1[,c(1,3,5,7)]
dig <- 2
cap = c('Naive Long System. A very simple system in which the algorithm assumes
    the market will rise and enters a long trade each day.',
            'Results from the Naive Long System')
lab = 'tab:nlng_results'
filname ='../Tables/chp_ta_naive_long.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -------------------------------------------------------
# ---------   Naive Long Base System          ----------
# --------- previous close and today's close ----------

run_NaiveLongSystem2 <- function(fil,SLoss, nm){
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
for(i in 1:length(fil)){
```

```
  Dax <- read.csv(fil[i])
  a <- NaiveLongSystem2(Dax, 0, nm[i])
  df10 <- rbind(df10, a)
}
df.name <- names(a)
names(df10) <- df.name
df10 <- df10[-1,]
return(df10)
}

res2 <- run_NaiveLongSystem2(fil,0,nm)
res2[misc_col] <- 'Naive Long 2'

# Add to total results
total_res <- rbind(total_res, res2)

# produce latex table
dat <- res2[,c(1,3,5,7)]
dig <- 2
cap = c('Naive Long System changed such that the trading period is the previous
    close price minus today\'s close.',
            'Results from the Naive Long System trading close to close')
lab = 'tab:nlng_results_2'
filname ='../Tables/chp_ta_naive_long_ctoc.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


# -----------------------------------------------------------------------------
# ------------ Reverse Previous Baseline System ------------------------------
# -----------------------------------------------------------------------------

res3 <- run_NaiveReversePrev(fil, 0, nm)
res3[misc_col] <- 'Reverse Prev'

# Add to total results
total_res <- rbind(total_res, res3)

# produce latex table
dat <- res3[,c(1,3,4,5,7,8,10)]
dig <- 2
cap = c('Results from a naive trading system which simply trades in the opposite
    direction to the previous day\'s movement.',
                      'Results from the Naive Reversing System.')
lab = 'tab:n_rev_results'
filname ='../Tables/chp_ta_naive_reverse_prev.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# repeat latex table for Chp6 - affects numbering if re-use from Chp5
dat <- res3[,c(1,3,4,5,7,8,10)]
dig <- 2
cap = c('Results from a naive trading system which simply trades in the opposite
    direction to the previous day\'s movement.',
        'Results from the Naive Reversing System.')
```

```
lab = 'tab:n_rev_results_chp6'
filname ='../Tables/chp_ta_naive_reverse_prev_chp6.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# repeat with a stop loss
res3a <- run_NaiveReversePrev(fil, -75, nm)
res3a[misc_col] <- 'Reverse Prev Stop Loss'

# Add to total results
total_res <- rbind(total_res, res3a)

# produce latex table
dat <- res3a[,std6]
dig <- 2
cap = c('Naive system which reverses the previous day\'s trade direction with
    stop loss.',
        'Naive Following System.')
lab = 'tab:n_rev_results_sl'
filname ='../Tables/chp_ta_naive_reverse_prev_sl.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


# ------------------------------------------------------------
# ----------------- Trend Detection Indicators ----------------

# --------------- SMA
run_BaseSystem1SMA <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Dax <- read.csv(fil[i])
    a <- BaseSystem1SMA(Dax, 5, SLoss, nm[i])
    b <- BaseSystem1SMA(Dax, 25, SLoss, nm[i])
    c <- BaseSystem1SMA(Dax, 50, SLoss, nm[i])
    d <- BaseSystem1SMA(Dax, 100, SLoss, nm[i])
    e <- BaseSystem1SMA(Dax, 200, SLoss, nm[i])
    df10 <- rbind(df10, a, b, c, d, e)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res4 <- run_BaseSystem1SMA(fil,0,nm)

# Add to total results
total_res <- rbind(total_res, res4)

dat <- res4[,c(1,3,4,5,7,8,10,11)]
dig <- 2
cap = c('Results from a system based on SMA.','Results from a system based on SMA
    ')
lab = 'tab:sma_results'
```

```
filname ='../Tables/chp_ta_sma.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# SMA SLoss ---------------------
run_BaseSystem1SMA2 <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Dax <- read.csv(fil[i])
    h <- BaseSystem1SMA(Dax, 100, -50, nm[i])
    hh <- BaseSystem1SMA(Dax, 100, -100, nm[i])  #don't use i !!!!!
    df10 <- rbind(df10,h,hh)
  }
  df.name <- names(hh)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res5 <- run_BaseSystem1SMA2(fil,0,nm)

# Add to total results
total_res <- rbind(total_res, res5)

dat <- res5[,c(1,3,4,5,7,8,10,11)]
dig <- 2
cap =  c('Results from a system based on SMA with stop loss.',
                      'Results from a system based on SMA with stop loss')
lab = 'tab:sma_results_Sloss'
filname ='../Tables/chp_ta_sma_sloss.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -----------------------------------------------------------------------
# --------- Moving Average Convergence/Divergence (MACD)}

run_MACD_XO <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    ma <- MACD( Mkt[,"Open"], 12, 26, 9, maType="EMA" ) #calc MACD values
    Mkt <- cbind(Mkt, ma)
    a <- MACD_XO(Mkt, SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res6 <- run_MACD_XO(fil,0,nm)
res6[misc_col] <- 'MACD'

# Add to total results
```

```
total_res <- rbind(total_res, res6)

dat <- res6[,std6]
dig <- 2
cap =  c('Results from a system using MACD as a trend indicator.',
         'Results from a system using MACD as a trend indicator')
lab = 'tab:mac_trend_results'
filname ='../Tables/chp_ta_macd.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


# -------------------------------------------
# ----------- Aroon -----------------------

run_aroon_sys <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    ar <- aroon(Mkt[c(3,4)], n=20)                  #calc Aroon values
    Mkt <- cbind(Mkt, ar)                            #Add Aroon values to orig
    data set
    a <- aroon_sys(Mkt, SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res7 <- run_aroon_sys(fil,0,nm)
res7[misc_col] <- 'Aroon'

# Add to total results
total_res <- rbind(total_res, res7)

dat <- res7[,std6]
dig <- 2
cap =  c('Results from a system based on the Aroon indicator.',
                      'Results from a system based on the Aroon indicator')
lab = 'tab:aroon_results'
filname ='../Tables/chp_ta_aroon.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


# ---- Aroon with SLoss
aroondfsl <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
for(i in 1:length(fil)){
  Dax <- read.csv(fil[i])                          #read data
  ar <- aroon(Dax[c(3,4)], n=20)                   #calc Aroon values
  Dax <- cbind(Dax, ar)                             #Add Aroon values to orig data
    set
  a <- aroon_sys(Dax, -100, nm[i])                  #Call fnc
```

```
  aroondfsl <- rbind(aroondfsl, a)
}
df.name <- names(a)
names(aroondfsl) <- df.name

res7a <- run_aroon_sys(fil,-100,nm)
aroondfsl <- res7a

res7a[misc_col] <- 'Aroon Stop Loss'

# Add to total results
total_res <- rbind(total_res, res7a)

dat <- res7a[,std6]
dig <- 2
cap =  c('Results from a system based on the Aroon indicator with stop loss.',
                      'Results from a system based on the Aroon indicator with
    stop loss')
lab = 'tab:aroon_results_sloss'
filname ='../Tables/chp_ta_aroon_sloss.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# Aroon - Diffs - between Aroon and Aroon with Stop Loss
aroondfsldf <- as.data.frame(matrix(seq(3),nrow=1,ncol=3))
ln <- nrow(aroondfsl)
res <- 1:3
for(i in 1:ln){
  res[1] <- aroondfsl[i,1]
  res[2] <- as.numeric(res7a[i,3]) - as.numeric(res7[i,3])
  res[3] <- as.numeric(res7a[i,4]) - as.numeric(res7[i,4])
  aroondfsldf <- rbind(aroondfsldf,res)
}
df.name <- c("Market", "Long Difference", "Short Difference")
names(aroondfsldf) <- df.name
aroondfsldf <- aroondfsldf[-1,]

dat <- aroondfsldf[,c(1,2,3)]
dig <- 2
cap =  c('Impact of using stop loss with Aroon trend indicator.',
                      'Impact of using stop loss with Aroon trend indicator')
lab = 'tab:aroon_results_sloss_diff'
filname ='../Tables/chp_ta_aroon_sloss_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# Aroon compared to baseline system
res7_diff <- sub_df_av_pl(res7,NaiveRev)
#print table
dat <- res7_diff
dig <- 0
cap =  c('Results from baseline Reversing System subtracted from Aroon results.',
         'Aroon results minus baseline')
lab = 'tab:aroon_results_diff'
filname ='../Tables/chp_ta_aroon_diff.tex'
```

```
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# --------------------------------------------------------------------------
# ------------ Trend Reversal -------------------------

# ----------- SAR
run_sar_sys <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    sar <- SAR(Mkt[c(3,4)]) #HL
    Mkt <- cbind(Mkt,sar)
    a <- sar_sys(Mkt,SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res8 <- run_sar_sys(fil,0,nm)
res8[misc_col] <- 'SAR'

# Add to total results
total_res <- rbind(total_res, res8)

dat <- res8[,std6]
dig <- 2
cap =  c('Results from a system based on the SAR indicator.',
         'Results from a system based on the SAR indicator')
lab = 'tab:sar_results'
filname ='../Tables/chp_ta_sar.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


# -----------------------------------------------------------
# ----------------- MACD OB -------------------------------

run_MACD_OB <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    ma <- MACD( Mkt[,"Open"], 12, 26, 9, maType="EMA" ) #calc MACD values
    Mkt <- cbind(Mkt, ma)                               #Add MACD values to orig data set
    lw <- quantile(Mkt$macd, na.rm=T, probs=0.15)       #Calc low val for algo
    up <- quantile(Mkt$macd, na.rm=T, probs=0.85)       #Calc up val for algo
    a <- MACD_OB(Mkt, 0, nm[i], lw, up)
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
```

```
  df10 <- df10[-1,]
  return(df10)
}

res9 <- run_MACD_OB(fil,0,nm)
res9[misc_col] <- 'MACD Reversal'

# Add to total results
total_res <- rbind(total_res, res9)

dat <- res9[,std6]
dig <- 2
cap =  c('Results from a trading system based on MACD being used as a trend reveral indicator.',
                        'Results from a system based on MACD as trend reversal indicator')
lab = 'tab:mac_ob_results'
filname ='../Tables/chp_ta_macd_ob.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


#-----------------------------------------------------------
# ----------------  Stochastic ----------------------------

ln <- nrow(df10)
for(i in 1:length(fil)){
  Dax <- read.csv(fil[i])
  st <- stoch(Dax[c(3,4,5)]) #HL
  Dax <- cbind(Dax,st)
  a <- stoch_sys(Dax, 0, nm[i])
  df10 <- rbind(df10, a)
}
df10 <- df10[-c(1:ln-1),]

run_stoch_sys <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    st <- stoch(Mkt[c(3,4,5)]) #HL
    Mkt <- cbind(Mkt,st)
    a <- stoch_sys(Mkt, SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res10 <- run_stoch_sys(fil,0,nm)
res10[misc_col] <- 'Stoch'

# Add to total results
total_res <- rbind(total_res, res10)
```

```
dat <- res10[,std6]
dig <- 2
cap =  c('Results from a system based on the Stochastic indicator.',
                      'Results from a system based on the Stochastic indicator')
lab = 'tab:stoch_results'
filname ='../Tables/chp_ta_stoch.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -------- Stochastic plus Stop Loss
res10a <- run_stoch_sys(fil,-100,nm)
res10a[misc_col] <- 'Stoch Stop Loss'

# Add to total results
total_res <- rbind(total_res, res10a)

dat <- res10a[,std6]
dig <- 2
cap =  c('Results from a system based on the Stochastic indicator with a stop
    loss.',
         'Results from a system based on the Stochastic indicator with a stop
    loss')
lab = 'tab:stoch_results_sloss'
filname ='../Tables/chp_ta_stoch_sloss.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

#---------------------------------------------------------
# -------------------------- ROC -----------------------

run_roc_sys <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    roc <- ROC( Mkt$Close )                      #calc MACD values
    Mkt <- cbind(Mkt, roc)                       #Add MACD values to orig
    data set
    lw <- quantile(Mkt$roc, na.rm=T, probs=0.15)     #Calc low val for algo
    up <- quantile(Mkt$roc, na.rm=T, probs=0.85)     #Calc up val for algo
    a <- roc_sys(Mkt, SLoss, nm[i], lw, up)
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res11 <- run_roc_sys(fil,0,nm)
res11[misc_col] <- 'ROC'

# Add to total results
total_res <- rbind(total_res, res11)
```

```

dat <- res11[,std6]
dig <- 2
cap =  c('Results from a system based on the ROC indicator.',
         'Results from a system based on the ROC indicator')
lab = 'tab:mac_roc_results'
filname ='../Tables/chp_ta_roc.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ----------------------------------------------------------------
# ------------ Break-out Systems ------------------------------

# -----------  Daily Break Out -----------------------
run_BaseSystem2Bout <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    a <- BaseSystem2Bout2(Mkt, SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res12 <- run_BaseSystem2Bout(fil,0,nm)
res12[misc_col] <- 'Daily Breakout'

# Add to total results
total_res <- rbind(total_res, res12)

dat <- res12[,std6]
dig <- 2
cap =  c('Results from the Daily High/Low Breakout System.',
         'Results from the Daily High/Low Breakout System')
lab = 'tab:hl_bout_sys'
filname ='../Tables/chp_ta_b_out.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# comp to Naive
res_diff <- sub_df(res12,NaiveRev)

dat <- res_diff[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from Daily High/Low Breakout System compared with Naive
    Reversing System",
         "Daily High/Low Breakout System compared with Naive Reversing System")
lab = 'tab:hl_bout_sys_diff'
filname ='../Tables/chp_ta_b_out_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

```

```
#-------------------------------------------------------------------
# ---------------- 90% Quantile Break-out ------------------------
run_BaseSystem3Quant902 <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    a <- BaseSystem3Quant902(Mkt, SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res14 <- run_BaseSystem3Quant902(fil,0,nm)
res14[misc_col] <- '90% Quantile Breakout'

# Add to total results
total_res <- rbind(total_res, res14)

dat <- res14[,std6]
dig <- 2
cap =  c('Results from a system that breaks out from the 90\\% quantile level of
    the day\'s minor move.',
         'Results from a break out system using the day\'s the minor move')
lab = 'tab:q_90_results'
filname ='../Tables/chp_ta_90q.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# comp to Naive
res_diff <- sub_df(res14,NaiveRev)

dat <- res_diff[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results 90\\% Quantile level Breakout System compared with Naive
    Reversing System",
         "Daily 90\\% Quantile level Breakout System compared with Naive
    Reversing System")
lab = 'tab:chp_ta_90q_diff'
filname ='../Tables/chp_ta_90q_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -----------------------------------------------------------------------------
# -------  Candlestick Patterns
# -------- Hammer Pattern
run_candle_hammer <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors = FALSE)
    Mkt <- Mkt[,c(1,2,3,4,5)]
    Mkt$Date <- as.POSIXct(Mkt$Date,format='%d/%m/%Y')
    Mkt_xts <- xts(Mkt[,c(2,3,4,5)],Mkt$Date)
```

```
    hh <- as.data.frame(CSPHammer(Mkt_xts))
    hi <- as.data.frame(CSPInvertedHammer(Mkt_xts))
    Mkt <- cbind(Mkt,hh)
    Mkt <- cbind(Mkt,hi)
    a <- candle_hammer(Mkt,SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res15 <- run_candle_hammer(fil,0,nm)
res15[misc_col] <- 'Hammer Candlestick'

# Add to total results
total_res <- rbind(total_res, res15)

# latex table
dat <- res15[,c(1,3,5,6,7)]
dig <- 2
cap = c('Results from a system based on the Hammer and Inverted Hammer
    candlestick patterns.',
        'Results from a system based on the Hammer and Inverted Hammer
    candlestick patterns')
lab = 'tab:hammer_results'
filname ='../Tables/chp_ta_hammer.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# plus aroon
run_candle_hammer_aroon <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors = FALSE)
    Mkt <- Mkt[,c(1,2,3,4,5)]
    Mkt$Date <- as.POSIXct(Mkt$Date,format='%d/%m/%Y')
    Mkt_xts <- xts(Mkt[,c(2,3,4,5)],Mkt$Date)
    hh <- as.data.frame(CSPHammer(Mkt_xts))
    hi <- as.data.frame(CSPInvertedHammer(Mkt_xts))
    Mkt <- cbind(Mkt,hh)
    Mkt <- cbind(Mkt,hi)
    ar <- aroon(Mkt$Close,n=20)
    Mkt <- cbind(Mkt,ar)
    a <- candle_hammer_aroon(Mkt,SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res15a <- run_candle_hammer_aroon(fil,0,nm)
```

```

# latex table
dat <- res15a[,c(1,3,5,6,7)]
dig <- 2
cap =  c('Results from a system based on the Hammer and Inverted Hammer
    candlestick patterns occurring in a downtrend as defined by the aroon value.'
    ,
         'Results from a system based on the Hammer and Inverted Hammer
    candlestick patterns occurring in a downtrend')
lab = 'tab:hammer_aroon_results'
filname ='../Tables/chp_ta_hammer_d_trend.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ---------------------------------------------------------------
# -----------------  Engulfing Candlestick -------------------
run_candle_engulf <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors = FALSE)
    #create xts obj
    Mkt$Date <- as.POSIXct(Mkt$Date,format='%d/%m/%Y')
    Mkt_xts <- xts(Mkt[,c(2,3,4,5)],Mkt$Date)
    en <- as.data.frame(CSPEngulfing(Mkt_xts))
    #use data frame again
    Mkt <- cbind(Mkt,en)
    a <- candle_engulf(Mkt,SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res16 <- run_candle_engulf(fil,0,nm)
res16[misc_col] <- 'Engulfing Candlestick'

# Add to total results
total_res <- rbind(total_res, res16)

# latex table
dat <- res16[,std6]
dig <- 2
cap =  c('Results from a system based on the Engulfing candlestick pattern.',
          'Results from a system based on the Engulfing candlestick pattern')
lab = 'tab:engulf_results'
filname ='../Tables/chp_ta_englf.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)


# --------- Engulfing Candlestick with Aroon
run_candle_engulf_aroon <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
```

```
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors = FALSE)
    #create xts obj
    Mkt$Date <- as.POSIXct(Mkt$Date,format='%d/%m/%Y')
    Mkt_xts <- xts(Mkt[,c(2,3,4,5)],Mkt$Date)
    en <- as.data.frame(CSPEngulfing(Mkt_xts))
    #use data frame again
    Mkt <- cbind(Mkt,en)
    ar <- aroon(Mkt$Close,n=20)
    Mkt <- cbind(Mkt,ar)
    a <- candle_engulf_aroon(Mkt,SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res16a <- run_candle_engulf_aroon(fil,0,nm)
res16a[misc_col] <- 'Engulfing Candlestick in Trend'

# Add to total results
total_res <- rbind(total_res, res16a)

# latex table
dat <- res16a[,std6]
dig <- 2
cap =  c('Results from a system based on the Engulfing candlestick pattern in a
    trending market.',
        'Results from a system based on the Engulfing candlestick pattern in a
    trending market')
lab = 'tab:engulf_aroon_results'
filname ='../Tables/chp_ta_englf_aroon.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ----------------------------------------------------------
# ------------  Doji Candlestick   ------------------------
run_candle_doji_aroon <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors = FALSE)
    #create xts obj
    Mkt$Date <- as.POSIXct(Mkt$Date,format='%d/%m/%Y')
    Mkt_xts <- xts(Mkt[,c(2,3,4,5)],Mkt$Date)
    dj <- as.data.frame(CSPDoji(Mkt_xts))
    #back to data fram
    Mkt <- cbind(Mkt,dj)
    ar <- aroon(Mkt$Close,n=20)
    Mkt <- cbind(Mkt,ar)
    a <- candle_doji_aroon(Mkt,SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
```

```
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res17 <- run_candle_doji_aroon(fil,0,nm)
res17[misc_col] <- 'Doji Candlestick'

# Add to total results
total_res <- rbind(total_res, res17)

# latex table
dat <- res17[,std6]
dig <- 2
cap = c('Results from a system based on the Doji candlestick pattern in a
    trending market.',
        'Results from a system based on the Doji candlestick pattern in a
    trending market')
lab = 'tab:doji_aroon_results'
filname ='../Tables/chp_ta_doji.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

#---------------------------------------------------------
# -- Generate Summary tables for Appendic C -------------
# 1. Dax
colnames(total_res)[11] <- 'Methodology'
Dx <- total_res[total_res$Mkt == 'Dax',]
Dx2 <- Dx[c(11,3,4,7,10)]

# latex table
dat <- Dx2
dig <- 2
cap = c('Chapter 4 Dax Results',
        'Chapter 4 Dax Results')
lab = 'tab:chp6:dax_summary'
filname ='../Tables/chp_6_dax_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 2. CAC
Cc <- total_res[total_res$Mkt == 'CAC',]
Cc2 <- Cc[c(11,3,4,7,10)]

# latex table
dat <- Cc2
dig <- 2
cap = c('Chapter 4 CAC Results',
        'Chapter 4 CAC Results')
lab = 'tab:chp6:cac_summary'
filname ='../Tables/chp_6_cac_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 3. FTSE
```

```
Ft <- total_res[total_res$Mkt == 'FTSE',]
Ft2 <- Ft[c(11,3,4,7,10)]

# latex table
dat <- Ft2
dig <- 2
cap = c('Chapter 4 FTSE Results',
        'Chapter 4 FTSE Results')
lab = 'tab:chp6:ftse_summary'
filname ='../Tables/chp_6_ftse_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 4. Dow
Dw <- total_res[total_res$Mkt == 'Dow',]
Dw2 <- Dw[c(11,3,4,7,10)]

# latex table
dat <- Dw2
dig <- 2
cap = c('Chapter 4 Dow Results',
        'Chapter 4 Dow Results')
lab = 'tab:chp6:dow_summary'
filname ='../Tables/chp_6_dow_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 5. Nikkei
Nk <- total_res[total_res$Mkt == 'Nikkei',]
Nk2 <- Nk[c(11,3,4,7,10)]

# latex table
dat <- Nk2
dig <- 2
cap = c('Chapter 4 Nikkei Results',
        'Chapter 4 Nikkei Results')
lab = 'tab:chp6:nik_summary'
filname ='../Tables/chp_6_nik_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 6. Oz
Oz <- total_res[total_res$Mkt == 'AORD',]
Oz2 <- Oz[c(11,3,4,7,10)]

# latex table
dat <- Oz2
dig <- 2
cap = c('Chapter 4 AORD Results',
        'Chapter 4 AORD Results')
lab = 'tab:chp6:aord_summary'
filname ='../Tables/chp_6_aord_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)
```

```
# END
```

RCode/Chapter4.R

A.1.2 Naive Systems

A.1.2.1 Naive Long System

```
NaiveLongSystem <- function(Mkt, SLoss, MktName){
  # Calculates the profit/loss from simply trading long.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  # Buy Long
  Mkt$Long <- Mkt$Close - Mkt$Open
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  return(results)
}
```

RCode/NaiveLongSystem.R

A.1.2.2 Naive Long System trading close to close

```
NaiveLongSystem2 <- function(Mkt, SLoss, MktName){
  # Calculates the profit/loss from simply trading long each day.
  # Opening price is previous day's close price.
  #
  # Args:
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: name of market data
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)
```

```

  Mkt$prevCl <- c(NA,Mkt$Close[ - length(Mkt$Close) ])

  # Buy Long
  Mkt$Long <- Mkt$Close - Mkt$prevCl
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  return(results)
}
```

RCode/NaiveLongSystem2.R

A.1.2.3 Naive Reversing System

```
NaiveReversePrev <- function(Mkt, SLoss, MktName){
  # Calculates the profit/loss from trading according to a naive idea of trading
    in the opposite direction to the previous day.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector

  results <- createResultsVector(MktName, SLoss)

  Mkt$pl <- Mkt$Close - Mkt$Open
  Mkt$prevPL <- c( NA, Mkt$pl[ - length(Mkt$pl) ] )

  # Trade Long
  Mkt$Long <- ifelse(Mkt$prevPL<0,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$prevPL<0,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)

    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$prevPL>0,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
```

```
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$prevPL>0,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/NaiveReversePrev.R

A.1.3 Trend Detection Systems

A.1.3.1 SMA

```
BaseSystem1SMA <- function(Mkt, sma, SLoss, MktName){
  # Calculates the profit/loss from trading according to SMA.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   profit/loss from trading according to SMA.

  results <- createResultsVector(MktName, SLoss)

  sma.value <- SMA(Mkt["Open"], sma)  #create sma vector
  Mkt <- cbind(Mkt, sma.value)        #add sma vector as new col

  # Trade Long
  Mkt$Long <- ifelse(Mkt$Open > Mkt$sma.value, Mkt$Close - Mkt$Open, NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=T))
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$Open > Mkt$sma.value,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=T))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$Open < Mkt$sma.value, Mkt$Open - Mkt$Close, NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=T))
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$Open < Mkt$sma.value,
                                ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$
    Short),
```

```
                                Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=T))
  }

  #calculate Long results
  results[5:7] <- calcStats(Mkt$Long)

  #calculate Short results
  results[8:10] <- calcStats(Mkt$Short)

  if (SLoss == 0){
    results[11] <- paste("SMA",sma)
  } else {
    results[11] <- paste("SMA",sma)
  }


  return(results)
}
```

RCode/SMA_sys.R

A.1.3.2 MACD - trend indicator

```
MACD_XO <- function(Mkt, SLoss, MktName){
  # MACD cross-over system.
  #
  # Args:
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  # Trade Long
  Mkt$Long <- ifelse(Mkt$macd>Mkt$signal,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$macd>Mkt$signal,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$macd<Mkt$signal,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$macd<Mkt$signal,
```

```
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  #calculate Long results
  results[5:7] <- calcStats(Mkt$Long)

  #calculate Short results
  results[8:10] <- calcStats(Mkt$Short)

  return(results)
}
```

RCode/MACD_XO.R

A.1.3.3 Aroon trend indicator

```
aroon_sys <- function(Mkt, SLoss, MktName){
  # uses Aroon indicator to trigger trades
  #
  # Args:
  #   Mkt:      Data to run system on
  #   SLoss:    Stop Loss (if 0 not used)
  #   MktName:  Name of market
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)
  # Trade Long
  Mkt$Long <- ifelse(Mkt$aroonUp >= 70,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$aroonUp >= 70,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$aroonDn >= 70,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
      Mkt$Short <- ifelse(Mkt$aroonDn >= 70,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  #calculate Long results
  results[5:7] <- calcStats(Mkt$Long)
```

```

  #calculate Short results
  results[8:10] <- calcStats(Mkt$Short)

  return(results)
}
```

RCode/Aroon.R

A.1.4 Market Reversal Indicator

A.1.4.1 SAR reversal indicator

```
sar_sys <- function(Mkt, SLoss, MktName){
  # uses Parabolic SAR indicator to trigger trades
  #
  # Args:
  #   Mkt:      Data
  #   SLoss:    Stop Loss (if 0 not used)
  #   MktName:  Name of market
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$prevsar <- c( NA, Mkt$sar[ - length(Mkt$sar) ])

  # Trade Long
  Mkt$Long <- ifelse(Mkt$Open > Mkt$prevsar,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$Open > Mkt$prevsar,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$Open < Mkt$prevsar,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$Open < Mkt$prevsar,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  #calculate Long results
  results[5:7] <- calcStats(Mkt$Long)

  #calculate Short results
```

```
  results[8:10] <- calcStats(Mkt$Short)

  return(results)
}
```

RCode/SAR.R

A.1.4.2 MACD as Reversal Indicator

```
MACD_OB <- function(Mkt, SLoss, MktName, lw, up){
  # MACD over-bought/sold system.
  #
  # Args:
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  #   lw: value of MACD that signals end of bear runs and rev
  #   up: value of MACD that signals end of bull runs and rev
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  # Trade Long
  Mkt$Long <- ifelse(Mkt$macd < lw,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$macd < lw,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$macd > up,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$macd > up,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/MACD_OB.R

A.1.4.3 Stochastic reversal indicator

```
stoch_sys <- function(Mkt, SLoss, MktName){
  # Trading system useing Stochastic Oscillator to trigger trades
  #
  # Args:
  #   Mkt:      Data
  #   SLoss:    Stop Loss (if 0 not used)
  #   MktName:  Name of market
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$PrevfastD <- c( NA, Mkt$fastD[ - length(Mkt$fastD) ])
  Mkt$PrevslowD <- c( NA, Mkt$slowD[ - length(Mkt$slowD) ])

  # Trade Long
  Mkt$Long <- ifelse(Mkt$PrevfastD > Mkt$PrevslowD,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$PrevfastD > Mkt$PrevslowD,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$PrevfastD < Mkt$PrevslowD,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$PrevfastD < Mkt$PrevslowD,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/Stoch.R

A.1.4.4 Rate of Change(ROC)

```
roc_sys <- function(Mkt, SLoss, MktName,lw, up){
  # Rate of Change (ROC) system.
  #
  # Args:
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  #   lw: value of ROC for reversal up
  #   up: value of ROC for reversal down
  #
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$prevROC <- c( NA, Mkt$roc[ - length(Mkt$roc) ] )

  # Trade Long
  Mkt$Long <- ifelse(Mkt$prevROC < lw,Mkt$Close-Mkt$Open,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$prevROC < lw,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$prevROC > up,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$prevROC > up,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/ROC.R

A.1.5 Break Out Systems

A.1.5.1 Break Out

```
BaseSystem2Bout <- function(Mkt, SLoss, MktName){
  # Trading system based on the break out of the previous day's high/low value.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$prevHigh <- c( NA, Mkt$High[ - length(Mkt$High) ] )
  Mkt$prevLow <- c( NA, Mkt$Low[ - length(Mkt$Low) ] )

  # Break out high
  Mkt$Long <- ifelse(Mkt$High>Mkt$prevHigh,Mkt$Close-Mkt$prevHigh,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$High>Mkt$prevHigh,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Break out low
  Mkt$Short <- ifelse(Mkt$Low<Mkt$prevLow,Mkt$prevLow-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$Low<Mkt$prevLow,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/Bout_sys.R

A.1.5.2 90% Quantile

```
BaseSystem3Quant902 <- function(Mkt, SLoss, MktName){
  # Calculates the profit/loss from trading a breakout of a 90% quantile move.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$OH <- Mkt$High - Mkt$Open
  Mkt$OL <- Mkt$Open - Mkt$Low
  Mkt$mn <- ifelse(Mkt$OH>Mkt$OL,Mkt$OL,Mkt$OH)
  qq <- quantile(Mkt$mn, probs=0.90)

  # Trade Long
  Mkt$Long <- ifelse((Mkt$High - Mkt$Open) > qq, Mkt$Close - (Mkt$Open + qq), NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$High - Mkt$Open) > qq,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse((Mkt$Open - Mkt$Low) > qq, (Mkt$Open - qq) - Mkt$Close, NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse((Mkt$Open - Mkt$Low) > qq,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/Quant90_sys.R

A.1.6 Candlestick Systems

A.1.6.1 Hammer and Inverted Hammer Candlestick Pattern

```
candle_hammer <- function(Mkt, SLoss, MktName){
  # Trading system based on the Hammer candelstick pattern.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$prev_Hammer   <- c( NA, Mkt$Hammer[ - length(Mkt$Hammer) ] )
  Mkt$prev_Inv_Hammer   <- c( NA, Mkt$InvertedHammer[ - length(Mkt$InvertedHammer
    ) ] )

  # Trade Long
  Mkt$Long <- ifelse(Mkt$prev_Hammer==TRUE | Mkt$prev_Inv_Hammer==TRUE, Mkt$Close
    -Mkt$Open, NA)
   results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$prev_Hammer==TRUE | Mkt$prev_Inv_Hammer==TRUE) > 0,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  return(results)
}
```

RCode/Candle_Hammer.R

A.1.6.2 Hammer and Inverted Hammer Candlestick Pattern in a Trending Market

```
candle_hammer_aroon <- function(Mkt, SLoss, MktName){
  # Trading system based on the Hammer candelstick pattern occurring in a
    trending market.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

```

```
  results <- createResultsVector(MktName, SLoss)

  #browser()
  Mkt$prev_Aroon_UP <- c( NA, Mkt$aroonUp[ - length(Mkt$aroonUp) ] )
  Mkt$prev_Aroon_DN <- c( NA, Mkt$aroonDn[ - length(Mkt$aroonDn) ] )
  Mkt$prev_Hammer   <- c( NA, Mkt$Hammer[ - length(Mkt$Hammer) ] )
  Mkt$prev_Inv_Hammer   <- c( NA, Mkt$InvertedHammer[ - length(Mkt$InvertedHammer ) ] )

  # Trade Long
  Mkt$Long <- ifelse(Mkt$prev_Aroon_DN >= 70, ifelse(Mkt$prev_Hammer==T | Mkt$prev_Inv_Hammer==T, Mkt$Close-Mkt$Open, NA) ,NA)

  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))

  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$High - Mkt$Open) > 0,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  return(results)
}
```

RCode/Candle_Hammer_aroon.R

A.1.6.3 Engulfing Candlestick Pattern

```
candle_engulf <- function(Mkt, SLoss, MktName){
  # Trading system based on the Engulfing candelstick pattern.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$prev_Bull_Engulf <- c( NA, Mkt$Bull.Engulfing[ - length(Mkt$Bull.Engulfing) ] )
  Mkt$prev_Bear_Engulf <- c( NA, Mkt$Bear.Engulfing[ - length(Mkt$Bear.Engulfing) ] )

  # Trade Long
   Mkt$Long <- ifelse(Mkt$prev_Bull_Engulf==TRUE, Mkt$Close-Mkt$Open, NA)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
#   #Adj for SLoss
```

```
  if (SLoss < 0) {
   Mkt$Long <- ifelse(Mkt$prev_Bull_Engulf == TRUE,
                      ifelse( (Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                      Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$prev_Bear_Engulf == TRUE,Mkt$Open-Mkt$Close,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Short <- ifelse(Mkt$prev_Bear_Engulf == TRUE,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/Candle_Engulf.R

A.1.6.4 Engulfing Candlestick Pattern in a Trending Market

```
candle_engulf_aroon <- function(Mkt, SLoss, MktName){
  # Trading system based on the Engulfing candelstick pattern occurring in a
    trending market.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  #browser()
  Mkt$prev_Aroon_UP <- c( NA, Mkt$aroonUp[ - length(Mkt$aroonUp) ] )
  Mkt$prev_Aroon_DN <- c( NA, Mkt$aroonDn[ - length(Mkt$aroonDn) ] )
  Mkt$prev_Bull_Engulf   <- c( NA, Mkt$Bull.Engulfing[ - length(Mkt$Bull.
    Engulfing) ] )
  Mkt$prev_Bear_Engulf   <- c( NA, Mkt$Bear.Engulfing[ - length(Mkt$Bear.
    Engulfing) ] )

  # Trade Long
```

```
  Mkt$Long <- ifelse(Mkt$prev_Aroon_DN >= 70, ifelse(Mkt$prev_Bull_Engulf==T, Mkt
    $Close-Mkt$Open, NA) ,NA)

  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))

  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$High - Mkt$Open) > 0,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  #Trade Short
  Mkt$Short <- ifelse(Mkt$prev_Aroon_UP >= 70, ifelse(Mkt$prev_Bull_Engulf==T,
    Mkt$Close-Mkt$Open, NA) ,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse((Mkt$Open - Mkt$Low) > 0,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/Candle_Engulf_aroon.R

A.1.6.5 Doji Candlestick Pattern in a Trending Market

```
candle_doji_aroon <- function(Mkt, SLoss, MktName){
  # Trading system based on the Doji candelstick pattern occurring in a trending
    market.
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  #
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  #browser()
  Mkt$prev_Aroon_UP <- c( NA, Mkt$aroonUp[ - length(Mkt$aroonUp) ] )
```

```
  Mkt$prev_Aroon_DN <- c( NA, Mkt$aroonDn[ - length(Mkt$aroonDn) ] )
  Mkt$prev_Doji   <- c( NA, Mkt$Doji[ - length(Mkt$Doji) ] )
  Mkt$prev_Dragonfly   <- c( NA, Mkt$DragonflyDoji[ - length(Mkt$DragonflyDoji) ]
    )
  Mkt$prev_Gravestone   <- c( NA, Mkt$GravestoneDoji[ - length(Mkt$GravestoneDoji
    ) ] )

  # Trade Long
  Mkt$Long <- ifelse(Mkt$prev_Aroon_DN >= 70, ifelse(Mkt$prev_Doji==TRUE | Mkt$
    prev_Dragonfly == TRUE, Mkt$Close-Mkt$Open, NA) ,NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))

  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse((Mkt$High - Mkt$Open) > 0,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  #Trade Short
  Mkt$Short <- ifelse(Mkt$prev_Aroon_UP >= 70, ifelse(Mkt$prev_Doji==TRUE | Mkt$
    prev_Gravestone == TRUE, Mkt$Close-Mkt$Open, NA) ,NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse((Mkt$Open - Mkt$Low) > 0,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/Candle_Doji_aroon.R

A.2 Chapter 5

The R code used to generate the results and tables in Chapter 5 is shown in listing A.2. This is followed by the individual files containing the algorithms used in the chapter.

```
# Chapter 5
setwd("D:/Allan/DropBox/MSc/Dissertation/Thesis/RCode")

# libraries
```

```
library(forecast)
library(xtable)

#source
source("../RCode/Utils.R")
source("../RCode/es_1.R")
source("../RCode/ts_1.R")
source("../RCode/ts_2.R")
source("../RCode/ts_3.R")
source("../RCode/ts_3a.R")
source("../RCode/ts_4.R")
source("../RCode//NaiveReversePrev.R")

fil <- c("../Data/Dax_2000_d.csv",
         "../Data/CAC_2000_d.csv",
         "../Data/F100_2000_d.csv",
         "../Data/Dow_2000_d.csv",
         "../Data/N225_2000_d.csv",
         "../Data/Oz_2000.csv")
nm <- c("DAX", "CAC", "FTSE", "Dow", "Nikkei", "AORD")

# Add Naive reverse prev for comparison purposes
# data frame will be fed into sub_df

df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
NaiveRev <- run_NaiveReversePrev(fil, 0, nm)
std6 <- c(1,3,4,5,7,8,10)
misc_col <- 11

# ------------------------------------------------------------
# ---------- Base Systems
Mkt <- read.csv("../Data/Dax_2000_d.csv")
Mkt$Date[2999]
Mkt_ts <- ts(Mkt$Close)
Mkt_train <- window(Mkt_ts, end=2999.99)
Mkt_test <- window(Mkt_ts, start=3000)

# a.build the  mean model
mean_model <- meanf(Mkt_train, h=5)
a <- accuracy(mean_model, Mkt_test) #out of sample
rownames(a) <- c('Mean Training Set', 'Mean Test Set')

# b. build the naive model
naive_model <- naive(Mkt_train, h=5)
b <- accuracy(naive_model, Mkt_test) #out of sample
rownames(b) <- c('Naive Training Set', 'Naive Test Set')

# c. build the drift model
drift_model <- rwf(Mkt_train,drift=TRUE,h=5)
c <- accuracy(drift_model, Mkt_test) #out of sample
rownames(c) <- c('Drift Training Set', 'Drift Test Set')

# combine results
d <- rbind(a,c)

```

```
# produce latex table
dat <- d[,c(2,3,4,5,6)]
dig <- 0
cap <- c("Error measures from mean and drift models.",
         "Error measures from mean and drift models")
lab = 'tab:chp_ts:sma'
filname ='../Tables/chp_ts_sma.tex'
inclrnam=TRUE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# --- plot all three base systems on Dow
savepdf("chp_ts_dax1")
Mkt_act <- window(Mkt_ts, start=3020, end=3200)
plot.ts(Mkt_train,
        main="Simple Forecasting Methods",
        xlab="Days since 2000", ylab="DAX Closing Price",
        xlim=c(2, 3200))
lines(meanf(Mkt_train, h=350) $mean, lty=2)
lines(rwf(Mkt_train,drift=TRUE,h=350)$mean,lty=3)
legend("bottomright",lty=c(2,3),
       legend=c("Mean method","Drift method"))
dev.off() #savepdf end

# --- plot all three base systems on Dow PLUS actual data
savepdf("chp_ts_dax1_plus_act_data")
Mkt_act <- window(Mkt_ts, start=3020, end=3200)
plot.ts(Mkt_train,
        main="Simple Forecasting Methods",
        xlab="Days since 2000", ylab="DAX Closing Price",
        xlim=c(2, 3200))
lines(meanf(Mkt_train, h=350) $mean, lty=2)
lines(rwf(Mkt_train,drift=TRUE,h=350)$mean,lty=3)
legend("bottomright",lty=c(2,3),
       legend=c("Mean method","Drift method"))
lines(Mkt_act, lty=4)
dev.off() #savepdf end

# 1. Exp Smoothing - mean model
# a. build data set - window thru and add prediction

# a1 - calculates mean prediction
exp_mean <- function(Mkt_ts, Mkt, strt, mean_flag){
  Mkta <- Mkt
  cc <- Mkta[1,]
  cc$a <- 0
  ln <- nrow(Mkt)
  for(i in strt:ln){
    st <- i-30
    Mkt_slice <- window(Mkt_ts,start=st,end=i)
    if (mean_flag == TRUE) {
      modf <- meanf(Mkt_slice,h=1)
    } else {
      modf <- rwf(Mkt_slice,drift=TRUE,h=1)
    }
    a <- as.numeric(modf$mean)
```

```
    c1 <- Mkta[i,]
    ab <- cbind(c1,a)
    cc <- rbind(cc,ab)
  }
  cc <- cc[-1,]
  return(cc)
}

# a2 -generates data sets with predictions
run_exp_mean <- function(fil,nm){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    Mkt_ts <- ts(Mkt$Close)
    res <- exp_mean(Mkt_ts,Mkt,400,TRUE)
    browser()
    write.csv(res,paste('../Data/ES/',nm[i],'_es_mean.csv',sep=""),row.names=
    FALSE)
  }
}

# a3 - run thru data sets - takes while so need to run just once
# run_exp_mean(fil,nm)

# a4 - use data sets in system
fil_mean <- c("../Data/ES/Dax_es_mean.csv",
              "../Data/ES/CAC_es_mean.csv",
              "../Data/ES/FTSE_es_mean.csv",
              "../Data/ES/Dow_es_mean.csv",
              "../Data/ES/Nikkei_es_mean.csv",
              "../Data/ES/AORD_es_mean.csv")

run_es_1 <- function(fil,SLoss,nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    a <- es_1(Mkt, SLoss, nm[i])
    df10 <- rbind(df10,a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

res_mean <- run_es_1(fil_mean,0,nm)
res_mean[misc_col] <- 'Mean Method'

# for summary results
total_res <- res_mean

dat <- res_mean[,std6]
dig <- 2
cap =  c('Results from trading the predictions generated by a mean exponential
    smoothing system.',
```

```
        'Results from trading the predictions generated by a mean exponential
    smoothing system')
lab = 'tab:es_mean_sys'
filname ='../Tables/chp_ts_es_mean.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -------------------------------------
# 2 - Drift model
run_exp_drift <- function(fil,nm){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    Mkt_ts <- ts(Mkt$Close)
    res <- exp_mean(Mkt_ts,Mkt,400,FALSE)
    write.csv(res,paste('../Data/ES/',nm[i],'_es_drift.csv',sep=""),row.names=
    FALSE)
  }
}

# a3 - run thru data sets - takes while so need to run just once
#run_exp_drift(fil,nm)

fil_drift <- c("../Data/ES/Dax_es_drift.csv",
              "../Data/ES/CAC_es_drift.csv",
              "../Data/ES/FTSE_es_drift.csv",
              "../Data/ES/Dow_es_drift.csv",
              "../Data/ES/Nikkei_es_drift.csv",
              "../Data/ES/AORD_es_drift.csv")

res_drift <- run_es_1(fil_drift,0,nm)
res_drift[misc_col] <- 'Drift Method'

# Add to total results
total_res <- rbind(total_res, res_drift)

dat <- res_drift[,std6]
dig <- 2

cap =  c('Results from trading the predictions generated by a drift exponential
    smoothing system.',
        'Results from trading the predictions generated by a drift exponential
    smoothing system')
lab = 'tab:es_drift_sys'
filname ='../Tables/chp_ts_es_drift.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 3. Exp Smoothing - ets model
# 3a. gen data set
exp_sm <- function(Mkt_ts, Mkt, strt){
  Mkta <- Mkt
  cc <- Mkta[1,]
  cc$a <- 0
  cc$b <- 0
  ln <- nrow(Mkt)
```

```
  for(i in strt:ln){
    st <- i-30
    Mkt_slice <- window(Mkt_ts,start=st,end=i)
    modf <- ets(Mkt_slice)
    fcastf <- forecast.ets(modf,h=1)
    a <- as.numeric(fcastf$mean)
    b <- modf$method
    c1 <- Mkta[i,]
    ab <- cbind(c1,b,a)
    cc <- rbind(cc,ab)
  }
  cc <- cc[-1,]
  return(cc)
}

# 3b - generates data sets with predictions
run_exp_sm <- function(fil,nm){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    Mkt_ts <- ts(Mkt$Close)
    res <- exp_sm(Mkt_ts,Mkt,400)
    browser()
    write.csv(res,paste('../Data/ES/',nm[i],'_es.csv',sep=""),row.names=FALSE)
  }
}

# loop thru data sets
#run_exp_sm(fil,nm)

# 3d Trade ES
fil_es <- c("../Data/ES/Dax_es.csv",
               "../Data/ES/CAC_es.csv",
               "../Data/ES/FTSE_es.csv",
               "../Data/ES/Dow_es.csv",
               "../Data/ES/Nikkei_es.csv",
               "../Data/ES/AORD_es.csv")

# use prev function
res_es <- run_es_1(fil_es,0,nm)
res_es[misc_col] <- 'Exponential Smoothing'

# Add to total results
total_res <- rbind(total_res, res_es)

dat <- res_es[,std6]
dig <- 2
cap =  c('Results from trading the predictions generated by a moving window
    exponential smoothing system.',
         'Results from trading the predictions generated by a moving window
    exponential smoothing system')
lab = 'tab:es_sys'
filname ='../Tables/chp_ts_es.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)
```

```
# -----------------------------------------
# 2. ARIMA ----------------------
Mkt <- read.csv("../Data/F100_2000_d.csv")
Mkt_ts <- ts(Mkt$Close)
Mkt_train <- window(Mkt_ts, end=2999.99)
Mkt_test <- window(Mkt_ts, start=3000)

# -----------------------------
# 2.1. Plot the data. Identify any unusual observations.
savepdf("chp_ts_ftse_2000-13")
plot.ts(Mkt_train,
        main="FTSE 2000 - 2013",
        xlab="Days since 2000",
        ylab="FTSE Closing Price",
        xlim=c(100, 3000))
dev.off()

# 2.2. If necessary, transform the data (using a Box-Cox transformation)
#to stabilize the variance.

# 2.3. If the data are non-stationary: take first differences of the
#data until the data are stationary.
savepdf("chp_ts_ftse_2000-13_diff")
plot(diff(Mkt_train),
          main="First Difference of FTSE 2000 - 2013",
          xlab="Days since 2000",
          ylab="FTSE Daily Price Movement",
          xlim=c(100, 3000))
dev.off()

# -------------------------------------
# 2.4. Examine the ACF/PACF: Is an AR(p) or MA(q) model appropriate?

# all 3 incl diff
savepdf("chp_ts_ftse_2000-13_diff_acf_tsd")
tsdisplay(diff(Mkt_train),main="FTSE 100 between 2000 and 2013",
          xlab="Days since 2000",
          ylab="FTSE Daily Price Movement")
dev.off()

# a ACF
savepdf("chp_ts_ftse_2000-13_diff_acf")
plot(Acf(diff(Mkt_train)),
     main="ACF of FTSE 100 between 2000 and 2013",
     ,ylim=c(-0.08, 0.08))
dev.off()

# a PACF
savepdf("chp_ts_ftse_2000-13_diff_pacf")
plot(Pacf(diff(Mkt_train)),
     main="PACF of FTSE 100 between 2000 and 2013",
     ylim=c(-0.08, 0.08))
dev.off()

# ----------------------------------------------------
```

```
# 2.5. Try your chosen model(s), and use the AICc to search for a better model.

mod_ar <- function(Mkt_ts, ord, nm){
  res <- t(as.data.frame(rep(0,4)))
  mod <- Arima(Mkt_ts, order=ord)
  res[1,1] <- nm
  res[1,2] <- round(mod$aic,1)
  res[1,3] <- round(mod$aicc,1)
  res[1,4] <- round(mod$bic,1)
  return(res)
}

results <- t(as.data.frame(rep(0,4)))
colnames(results) <- c('Model','AIC','AICc','BIC')

r2 <- mod_ar(Mkt_train, c(3,1,1), 'ARIMA(3,1,1)')
results <- rbind(results,r2)
r2 <- mod_ar(Mkt_train, c(3,1,2), 'ARIMA(3,1,2)')
results <- rbind(results,r2)
r2 <- mod_ar(Mkt_train, c(3,1,3), 'ARIMA(3,1,3)')
results <- rbind(results,r2)
r2 <- mod_ar(Mkt_train, c(2,1,1), 'ARIMA(2,1,1)')
results <- rbind(results,r2)
r2 <- mod_ar(Mkt_train, c(2,1,2), 'ARIMA(2,1,2)')
results <- rbind(results,r2)
r2 <- mod_ar(Mkt_train, c(2,1,3), 'ARIMA(2,1,3)')
results <- rbind(results,r2)
results <- results[-1,]

# produce latex table
dat <- results
dig <- c(0,0,2,2,2)
cap <- c("AIC, AICc and BIC results from alternative ARIMA models.",
         "AIC, AICc and BIC results from alternative ARIMA models")
lab = 'tab:chp_ts:arima_res_r'
filname ='../Tables/chp_ts_arima_res_r.tex'
inclrnam=F
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -----------------------------------------------------
# 2.6. Check the residuals from your chosen model by plotting the ACF of the
    residuals,
#and doing a portmanteau test of the residuals.
#If they do not look like white noise, try a modified model.

model_used_for_res <- Arima(Mkt_train, order=c(2,1,3))
model_name <- forecast(model_used_for_res)$method

# a mean of residual
residual <- model_used_for_res$residuals
savepdf("chp_ts_ftse_2000-13_mean_residuals")
plot(residual, main = paste("Residuals from model of", model_name),
     ylab="", xlab="Day")
dev.off()

```

```
# b. acf of residual
savepdf("chp_ts_ftse_2000-13_acf_residuals")
Acf(residuals(model_used_for_res),
    main= paste("ACF of Residuals of", model_name))
dev.off()

# d. histogram of residuals - normal distribution
savepdf("chp_ts_ftse_2000-13_hist_residuals")
hist(residual, nclass="FD", main="Histogram of residuals")
dev.off()

# e. portmanteau tests
bb <- Box.test(residuals(model_used_for_res), lag=24, fitdf=4, type="Ljung")
results_bc <- as.data.frame(rep(0,3))
results_bc[1,1] <- round(bb$p.value,4)
results_bc[2,1] <- round(bb$parameter)
results_bc[3,1] <- round(bb$statistic)
colnames(results_bc) <- c(forecast(model_used_for_res)$method)
rownames(results_bc) <- c('p-value','x-squared','df')
results_bc_t <- t(results_bc)

dat <- results_bc_t
dig <- c(0,4,0,0)
cap <- c("Box Ljung test of FTSE 100 ARIMA model residuals.",
         "Box Ljung test of FTSE 100 ARIMA model residuals")
lab = 'tab:chp_ts:arima_res_rbox_l'
filname ='../Tables/chp_ts_arima_res_r_box_l.tex'
inclrnam=TRUE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 2.7 Once the residuals look like white noise, calculate forecasts.
model_used_for_res <- Arima(Mkt_ts, order=c(2,1,3))
model_name <- forecast(model_used_for_res)$method

arima_man_fcast <- forecast.Arima(model_used_for_res,Mkt_test)
fitted.data <- as.data.frame(arima_man_fcast$fitted);
Mkt_test_df <- cbind(Mkt,fitted.data)
colnames(Mkt_test_df) <- c('Date','Open','High','Low','Close','Forecast')

# plot the results
dat <- tail(Mkt_test_df)
dig <- 0
cap <- c("One step ahead forecast for FTSE 100 generated from ARIMA(2,1,3) model.
    ",
         "Forecast for FTSE 100 generated from the ARIMA model")
lab = 'tab:chp_ts:ftse_100_fcast'
filname ='../Tables/chp_ts_ftse_100_fcast.tex'
inclrnam=F
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ------------------------------------------------
# 2.8 auto.arima
arim_mod_fnc <- function(fil,nm){
  dfres <- dfres <- t(c('a','b'))
  for(i in 1:length(fil)){
```

```
    Mkt <- read.csv(fil[i])
    Mkt_train <- ts(Mkt$Close)
    arima_train_mod <- auto.arima(Mkt_train)
    dfres <- rbind(dfres,c(nm[i], forecast(arima_train_mod)$method))
  }
  return(dfres)
}

fg <- arim_mod_fnc(fil,nm)
fg <- fg[-1,]
colnames(fg) <- c('Market','ARIMA Model')

# plot the results
dat <- fg
dig <- 0
cap <- c("ARIMA models chosen to forecast future values in the national indice
    data sets.",
         "ARIMA models chosen for the indice data sets")
lab = 'tab:chp_ts_arima_models'
filname ='../Tables/chp_ts_arima_models.tex'
inclrnam=F
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# plot the results for Chp 6 ...
dat <- fg
dig <- 0
cap <- c("ARIMA models chosen to forecast future values in the national indice
    data sets.",
         "ARIMA models chosen for the indice data sets")
lab = 'tab:chp_ts_arima_models_chp6'
filname ='../Tables/chp_ts_arima_models_chp6.tex'
inclrnam=F
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ----------------------------------------------------
# 3. Trading System
# using the models generated from the auto.arima function
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
ts_1_fnc <- function(fil,nm,ts1){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i])
    Mkt_ts <- ts(Mkt$Close)
    Mkt_train <- window(Mkt_ts, end=2999.99)
    Mkt_test <- window(Mkt_ts, start=3000)
    arima_train_mod <- auto.arima(Mkt_train)
    arima_fcast <- forecast.Arima(arima_train_mod,Mkt_test)
    arima_test_mod <- Arima(Mkt_test, model = arima_train_mod) # 1 step fcast on
    future data ...
    arima_test_fcast <- forecast(arima_test_mod)
    fitted.data <- as.data.frame(arima_test_fcast$fitted);
    ln <- nrow(Mkt)
    lw <- nrow(fitted.data)
    Mkt_test_df <- Mkt[(ln-lw+1):ln,]
    Mkt_test_df <- cbind(Mkt_test_df,fitted.data)
    colnames(Mkt_test_df) <- c("Date","Open", "High","Low","Close","p")
```

```
    if(ts1 == TRUE){
      a <- ts_1(Mkt_test_df, 0, nm[i]) # System 1
    } else {
      a <- ts_2(Mkt_test_df, 0, nm[i]) # System 2
    }
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-c(1),]
  return(df10)
}

# run the fnc ts_1 and apply Sys 1 to the auto.arima data
res1 <- ts_1_fnc(fil,nm,TRUE)
res1[misc_col] <- 'ARIMA - System 1'

# Add to total results
total_res <- rbind(total_res, res1)

# produce latex table from ts_1
dat <- res1[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from trading System 1 using the forecasts generated by the ARIMA models.",
         "Results from trading System 1 using the forecasts generated by the ARIMA models")
lab = 'tab:chp_ts:arima1'
filname ='../Tables/chp_ts_arima1.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# compare to Naive reverse
diff_df1 <- sub_df_av_pl(res1,NaiveRev)
# produce latex table from ts_1
#dat <- diff[,c(1,7,10)]
dat <- diff_df1
dig <- 0
cap <- c("Mean Long/Short PL from system using predictions from ARIMA models with the results from the Naive Reverse system subtracted.",
         "Mean PL from ARIMA models minus mean PL from Naive Reverse system")
lab = 'tab:chp_ts:arima1_diff'
filname ='../Tables/chp_ts_arima1_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ------------------------------------------------------------------
# run the fnc ts_2
# apply system 2 to auto.arima data
res2 <- ts_1_fnc(fil,nm,FALSE) # F = ts_2
res2[misc_col] <- 'ARIMA - System 2'

# Add to total results
total_res <- rbind(total_res,res2)

```

```
# produce latex table from ts_2
dat <- res2[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from trading System 2 using the forecasts generated by the
    ARIMA models.",
         "Results from trading System 2 using the forecasts generated by the
    ARIMA models")
lab = 'tab:chp_ts:arima2'
filname ='../Tables/chp_ts_arima2.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# --------- RM Generated Files ----------------------------------------
# --------- HYBRID ARIMA SYSTEMS --------------------------------------

ts_1_2_fnc_ar <- function(fil,nm,ts1){
  for(i in 1:length(fil)){
    #browser()
    Mkt <- read.csv(fil[i],stringsAsFactors=F)
    Mkt_p <- Mkt[,c(1,2,3,4,5,20)]
    colnames(Mkt_p) <- c("Date","Open", "High","Low","Close","p")
    if(ts1 == TRUE){
      a <- ts_1(Mkt_p, 0, nm[i])
    } else {
      a <- ts_2(Mkt_p, 0, nm[i])
    }
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-c(1),]
  return(df10)
}

# ---------- Predicting Closing Price -----------------
# 1. ------ Arima Ann Predicting Closing Price -----------------
fil <- c("../Data/ARIMA2/Predict_Close/ar334_ann_DAX.csv",
         "../Data/ARIMA2/Predict_Close/ar334_ann_CAC.csv",
         "../Data/ARIMA2/Predict_Close/ar334_ann_FTSE.csv",
         "../Data/ARIMA2/Predict_Close/ar334_ann_Dow.csv",
         "../Data/ARIMA2/Predict_Close/ar334_ann_Nik.csv",
         "../Data/ARIMA2/Predict_Close/ar334_ann_Oz.csv")

#nm <- c("Dax","CAC","FTSE","Dow","Nikkei","AORD")
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))

# a. System 1
res3 <- ts_1_2_fnc_ar(fil,nm,TRUE)
res3[misc_col] <- 'ARIMA/ANN Closing Price System 1'

# Add to total results
total_res <- rbind(total_res, res3)

# produce latex table from ts_1
dat <- res3[,c(1,3,4,5,7,8,10)]
```

```
dig <- 0
cap <- c("Results from passing closing price predictions from hybrid ARIMA/ANN model to System 1.",
         "Results from passing closing price predictions from hybrid ARIMA/ANN model to System 1")
lab = 'tab:chp_ts:arima_ann_sys1'
filname ='../Tables/chp_ts_arima_ann_sys1.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# comp aring to Naive Prev
res_diff3 <- sub_df_av_pl(res3,NaiveRev)

dat <- res_diff3
dig <- 0
cap <- c("Results from a trading system based on forecasts of closing price generated by the ARIMA/ANN model compared to baseline Naive Reversing methodology.",
         "ARIMA/ANN predictions passed to System 1 compared to Naive Reversing methodology")
lab = 'tab:chp_ts:arima_ann_sys1_diff'
filname ='../Tables/chp_ts_arima_ann_sys1_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# a. System 2
res4 <- ts_1_2_fnc_ar(fil,nm,FALSE)
res4[misc_col] <- 'ARIMA/ANN Closing Price System 2'

# Add to total results
total_res <- rbind(total_res, res4)

# produce latex table from ts_1
dat <- res4[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from passing closing price predictions from hybrid ARIMA/ANN model to System 2.",
         "Results from passing closing price predictions from hybrid ARIMA/ANN model to System 2")
lab = 'tab:chp_ts:arima_ann_sys2'
filname ='../Tables/chp_ts_arima_ann_sys2.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 2. ------ Arima knn Predicting Closing Price -----------------
fil <- c("../Data/ARIMA2/Predict_Close/ar334_knn_Dax.csv",
         "../Data/ARIMA2/Predict_Close/ar334_knn_CAC.csv",
         "../Data/ARIMA2/Predict_Close/ar334_knn_F100.csv",
         "../Data/ARIMA2/Predict_Close/ar334_knn_Dow.csv",
         "../Data/ARIMA2/Predict_Close/ar334_knn_Nik.csv",
         "../Data/ARIMA2/Predict_Close/ar334_knn_Oz.csv")

# a. System 1
#nm <- c("Dax","CAC","FTSE","Dow","Nikkei","AORD")
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
```

```
# a. System 1
res5 <- ts_1_2_fnc_ar(fil,nm,TRUE)
res5[misc_col] <- 'ARIMA/k-NN Closing Price System 1'

# Add to total results
total_res <- rbind(total_res, res5)

# produce latex table from ts_1
dat <- res5[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from passing closing price predictions from hybrid ARIMA/k-NN
    model to System 1.",
         "Results from passing closing price predictions from hybrid ARIMA/k-NN
    model to System 1")
lab = 'tab:chp_ts:pred_close_arima_knn_sys1'
filname ='../Tables/chp_ts_pred_close_arima_knn_sys1.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# comp aring to Naive Prev
#res_diff <- sub_df_av_pl(res,NaiveRev)
res_diff5 <- sub_df(res5,NaiveRev)

# produce latex table from ts_1
dat <- res_diff5[,c(1,5,7,8,10)]
dig <- 0
cap <- c("Results from a system using forecasts from a ARIMA/k-NN model with the
    results of the Naive Reversing System subtracted.",
         "Mean PL from hybrid ARIMA/k-NN models minus mean PL from Naive Reverse
    system")
lab = 'tab:chp_ts:pred_close_arima_knn_sys1_diff'
filname ='../Tables/chp_ts_pred_close_arima_knn_sys1_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# a. System 2
res6 <- ts_1_2_fnc_ar(fil,nm,FALSE)
res6[misc_col] <- 'ARIMA/k-NN Closing Price System 2'

# Add to total results
total_res <- rbind(total_res, res6)

# produce latex table from ts_1
dat <- res6[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from passing closing price predictions from hybrid ARIMA/k-NN
    model to System 2.",
         "Results from passing closing price predictions from hybrid ARIMA/k-NN
    model to System 2")
lab = 'tab:chp_ts:pred_close_arima_knn_sys2'
filname ='../Tables/chp_ts_pred_close_arima_knn_sys2.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# ---------------------------------------------------------------
```

```
#  ------ Arima Ann Predicting Up/Dn - Categorical -----------------
# a. Categorical

# 1. ARMA / ANN (Predicting Up/Dn - Categorical)
fil <- c("../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_ann_Dax.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_ann_CAC.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_ann_F100.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_ann_Dow.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_ann_N225.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_ann_Oz.csv")

#nm <- c("Dax","CAC","FTSE","Dow","Nik","AORD")
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))

res7 <- ts_4_fnc_ar(fil,0, nm)
res7[misc_col] <- 'ARIMA/ANN Up/Down'

# Add to total results
total_res <- rbind(total_res, res7)

# produce latex table from ts_1
dat <- res7[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from a trading system using the forecast of categorical label \
    "U/D\" from hybrid ARIMA/ANN model.",
         "Results from a trading system using the forecast of categorical label \
    "U/D\" from hybrid ARIMA/ANN model")
lab = 'tab:chp_ts:pUD_CAT_arima_ann_sys'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_ann_sys.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -----------------------
# 2. ARMA / knn (Predicting Up/Dn - Categorical)
fil <- c("../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_knn_Dax.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_knn_CAC.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_knn_F100.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_knn_Dow.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_knn_N225.csv",
         "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_knn_Oz.csv")

#nm <- c("Dax","CAC","FTSE","Dow","Nikkei","AORD")
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))

res8 <- ts_4_fnc_ar(fil, 0, nm)
res8[misc_col] <- 'ARIMA/k-NN Up/Down'

# Add to total results
total_res <- rbind(total_res, res8)

# produce latex table from ts_1
dat <- res8[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from a trading system using the forecast of categorical label \
    "U/D\" from hybrid ARIMA/k-NN model.",
```

```
        "Results from a trading system using the forecast of categorical label \"U/D\" from hybrid ARIMA/k-NN model")
lab = 'tab:chp_ts:pUD_CAT_arima_knn_sys'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_knn_sys.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 2. ARMA / knn (Predicting Up/Dn - Categorical) - SLoss
res8a <- ts_4_fnc_ar(fil, -100, nm)
res8a[misc_col] <- 'ARIMA/ANN Up/Down Stop Loss'

# Add to total results
total_res <- rbind(total_res, res8a)

# produce latex table from ts_1
dat <- res8a[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from a trading system with a stop loss using the forecast of categorical label \"U/D\" from hybrid ARIMA/k-NN model.",
        "Results from a trading system with a stop loss using the forecast of categorical label \"U/D\" from hybrid ARIMA/k-NN model")
lab = 'tab:chp_ts:pUD_CAT_arima_knn_sys_SL'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_knn_sys_SL.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# comp aring to Naive Prev
res_diff8 <- sub_df(res8,NaiveRev)

# produce latex table from ts_1
dat <- res_diff8[,c(1,5,7,8,10)]
dig <- 0
cap <- c("Results from Naive Reversing System subtracted from results generated from predicting Up/Down categorical label using ARIMA/k-NN.",
        "Naive Reversing System subtracted from ARIMA/k-NN predictions")
lab = 'tab:chp_ts:pUD_CAT_arima_knn_sys_diff'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_knn_sys_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 4. ARMA / SVM (Predicting Up/Dn - Categorical)
fil <- c("../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_svm_Dax.csv",
        "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_svm_CAC.csv",
        "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_svm_F100.csv",
        "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_svm_Dow.csv",
        "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_svm_N225.csv",
        "../Data/ARIMA2/PredUpDn_CAT/ar_334_UD_svm_Oz.csv")

#nm <- c("Dax","CAC","FTSE","Dow","Nikkei","AORD")
df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))

res9 <- ts_4_fnc_ar(fil,0, nm)
res9[misc_col] <- 'ARIMA/SVM Up/Down'

# Add to total results
```

```
total_res <- rbind(total_res, res9)

# produce latex table from ts_1
dat <- res9[,c(1,3,4,5,7,8,10)]
dig <- 0
cap <- c("Results from a trading system using the forecast of categorical label \
    "U/D\" from hybrid ARIMA/SVM model.",
         "Results from a trading system using the forecast of categorical label \
    "U/D\" from hybrid ARIMA/SVM model")
lab = 'tab:chp_ts:pUD_CAT_arima_svm_sys'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_svm_sys.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# comp aring to Naive Prev
res_diff9 <- sub_df(res9,NaiveRev)

# produce latex table from ts_1
dat <- res_diff9[,c(1,5,7,8,10)]
dig <- 0
cap <- c("Results from Naive Reversing System subtracted from results generated
    from predicting Up/Down categorical label using ARIMA/SVM.",
         "Naive Reversing System subtracted from ARIMA/SVM predictions")
lab = 'tab:chp_ts:pUD_CAT_arima_svm_sys_diff'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_svm_sys_diff.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# Chp6
lab = 'tab:chp_ts:pUD_CAT_arima_svm_sys_ch6'
filname ='../Tables/chp_ts_predUpDn_CAT_arima_svm_sys_chp6.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -----------------------------------------------
#  ------ Arima Ann Predicting Up/Dn - 01 ---------
# 1. ARMA / ANN - (Predicting Up/Dn - 01)
# fil <- c("../Data/ARIMA/PredUpDn_01/ar_334_01_ANN_Dax.csv",
#           "../Data/ARIMA/PredUpDn_01/ar_334_01_ANN_CAC.csv",
#           "../Data/ARIMA/PredUpDn_01/ar_334_01_ANN_FTSE.csv",
#           "../Data/ARIMA/PredUpDn_01/ar_334_01_ANN_Dow.csv",
#           "../Data/ARIMA/PredUpDn_01/ar_334_01_ANN_N225.csv",
#           "../Data/ARIMA/PredUpDn_01/ar_334_01_ANN_Oz.csv")
#
# #nm <- c("Dax","CAC","FTSE","Dow","Nikkei","AORD")
# df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
#
# res10 <- ts_3a_fnc_ar(fil, nm)
#
# # produce latex table from ts_1
# dat <- res10[,c(1,3,4,5,7,8,10)]
# dig <- 0
# cap <- c("Results from a trading system using the forecast of a continous label
     from a hybrid ARIMA/ANN model.",
```

```
#          "Results from a trading system using the forecast of a continous label
     from a hybrid ARIMA/ANN model")
# lab = 'tab:chp_ts:pUD_01_arima_ann_sys'
# filname ='../Tables/chp_ts_predUpDn_01_arima_ann_sys.tex'
# inclrnam=FALSE
# print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# -- Generate Summary tables for Chp6
# 1. Dax
colnames(total_res)[11] <- 'Methodology'
Dx <- total_res[total_res$Mkt == 'DAX',]
Dx2 <- Dx[c(11,3,4,7,10)]

# latex table
dat <- Dx2
dig <- 2
cap = c('Chapter 5 DAX Results',
        'Chapter 5 DAX Results')
lab = 'tab:chp6:dax2_summary'
filname ='../Tables/chp_6_dax2_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 2. CAC
Cc <- total_res[total_res$Mkt == 'CAC',]
Cc2 <- Cc[c(11,3,4,7,10)]

# latex table
dat <- Cc2
dig <- 2
cap = c('Chapter 5 CAC Results',
        'Chapter 5 CAC Results')
lab = 'tab:chp6:cac2_summary'
filname ='../Tables/chp_6_cac2_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 3. FTSE
Ft <- total_res[total_res$Mkt == 'FTSE',]
Ft2 <- Ft[c(11,3,4,7,10)]

# latex table
dat <- Ft2
dig <- 2
cap = c('Chapter 5 FTSE Results',
        'Chapter 5 FTSE Results')
lab = 'tab:chp6:ftse2_summary'
filname ='../Tables/chp_6_ftse2_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 4. Dow
Dw <- total_res[total_res$Mkt == 'Dow',]
Dw2 <- Dw[c(11,3,4,7,10)]

```

```
# latex table
dat <- Dw2
dig <- 2
cap = c('Chapter 5 Dow Results',
        'Chapter 5 Dow Results')
lab = 'tab:chp6:dow2_summary'
filname ='../Tables/chp_6_dow2_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 5. Nikkei
Nk <- total_res[total_res$Mkt == 'Nikkei',]
Nk2 <- Nk[c(11,3,4,7,10)]

# latex table
dat <- Nk2
dig <- 2
cap = c('Chapter 5 Nikkei Results',
        'Chapter 5 Nikkei Results')
lab = 'tab:chp6:nik2_summary'
filname ='../Tables/chp_6_nik2_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# 6. Oz
Oz <- total_res[total_res$Mkt == 'AORD',]
Oz2 <- Oz[c(11,3,4,7,10)]

# latex table
dat <- Oz2
dig <- 2
cap = c('Chapter 5 AORD Results',
        'Chapter 5 AORD Results')
lab = 'tab:chp6:aord2_summary'
filname ='../Tables/chp_6_aord2_summary.tex'
inclrnam=FALSE
print_xt(dat,dig,cap,lab,al,filname,inclrnam)

# END
```

RCode/Chapter5.R

A.2.1 Exponential Smoothing

```
es_1 <- function(Mkt, SLoss, MktName){
  # Trading system using predictions from exponential smoothing models.
  #
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.
```

```

  results <- createResultsVector(MktName, SLoss)

  Mkt$pred_d <- ifelse(Mkt$a > Mkt$Close, 'U','D')
  Mkt$pu <-c( NA, Mkt$pred_d[ - length(Mkt$pred_d) ] )

  # Trade Long
  Mkt$Long <- ifelse(Mkt$pu == 'U', Mkt$Close - Mkt$Open, NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$p > Mkt$p_c,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$pu == 'D', Mkt$Open - Mkt$Close, NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse(Mkt$p < Mkt$p_c,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats2(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats2(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/es_1.R

A.2.2 System 1

```
ts_1 <- function(Mkt, SLoss, MktName){
  # Trading system using predictions from ARIMA models. Uses relative
  # value of the forecast with the previous close
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

```

```
  Mkt$p_c <- c( NA, Mkt$Close[ - length(Mkt$Close) ] ) # prev close
  Mkt$p_p <- c( NA, Mkt$p[ - length(Mkt$p) ] ) # prev pred

  # Trade Long
  Mkt$Long <- ifelse(Mkt$p_p > Mkt$p_c, Mkt$Close - Mkt$Open, NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$p_p > Mkt$p_c,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$p_p < Mkt$p_c, Mkt$Open - Mkt$Close, NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse(Mkt$p_p < Mkt$p_c,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats2(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats2(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/ts_1.R

A.2.3 System 2

```
ts_2 <- function(Mkt, SLoss, MktName){
  # Trading system using predictions from ARIMA models. Uses
  # relative value of the forecast and the previous forecast
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)

  Mkt$p_p <- c( NA, Mkt$p[ - length(Mkt$p) ] ) # prev prediction
  Mkt$p_p2 <- c( NA, Mkt$p_p[ - length(Mkt$p_p) ] ) # prev prediction

```

```r
  # Trade Long
  Mkt$Long <- ifelse(Mkt$p_p > Mkt$p_p2, Mkt$Close - Mkt$Open, NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$p_p > Mkt$p_p2,
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$p_p < Mkt$p_p2, Mkt$Open - Mkt$Close, NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse(Mkt$p_p < Mkt$p_p2,
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats2(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats2(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/ts_2.R

A.2.4 Categorical Label

```r
ts_4 <- function(Mkt, SLoss, MktName){
  #   trading system based on prediction fron ANN working with categorical
  #   label with valued U or D
  #
  #   Mkt: market data
  #   SLoss: stop loss
  #   MktName: market's name for print out
  #
  # Returns:
  #   results vector.

  results <- createResultsVector(MktName, SLoss)
  Mkt$p_p <- c( NA, Mkt$pred[ - length(Mkt$pred) ] ) # prev pred

  # Trade Long
  Mkt$Long <- ifelse(Mkt$p_p == "U", Mkt$Close - Mkt$Open, NA)
  results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  #Adj for SLoss
```

```
  if (SLoss < 0) {
    Mkt$Long <- ifelse(Mkt$p_p == "U",
                       ifelse((Mkt$Low-Mkt$Open) < SLoss, SLoss, Mkt$Long),
                       Mkt$Long)
    results["LongPL"] <- round(sum(Mkt$Long, na.rm=TRUE))
  }

  # Trade Short
  Mkt$Short <- ifelse(Mkt$p_p == "D", Mkt$Open - Mkt$Close, NA)
  results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  #Adj for SLoss
  if (SLoss < 0){
    Mkt$Short <- ifelse(Mkt$p_p == "D",
                        ifelse((Mkt$Open-Mkt$High) < SLoss, SLoss, Mkt$Short),
                        Mkt$Short)
    results["ShortPL"] <- round(sum(Mkt$Short, na.rm=TRUE))
  }

  Stats <- calcStats2(Mkt$Long)
  results[5:7] <- Stats

  Stats <- calcStats2(Mkt$Short)
  results[8:10] <- Stats

  return(results)
}
```

RCode/ts_4.R

A.3 Utility Code

```
nm <- c("DAX", "CAC", "FTSE", "Dow", "Nikkei", "AORD")

createResultsVector <- function(MktName, SLossValue){
  # Function to create results vector
  #
  # Args:
  #   SLoss: stop loss value
  #   MktName: market's name for print out
  #
  # Returns:
  #   results vector.

  results <- rep(0,11)
  nam <- c("Mkt",          # 1. Name of Mkt
           "S Loss",       # 1. Name of Mkt
           "LongPL",       # 1. Name of Mkt
           "ShortPL",      # 1. Name of Mkt
           "L Win %",      # 1. Name of Mkt
           "L Trades",    # 1. Name of Mkt
           "Av L PL",      # 1. Name of Mkt
           "S Win %",      # 1. Name of Mkt
```

```
          "S Trades",    # 1. Name of Mkt
          "Av S PL",
          "misc")      # 1. Name of Mkt
  names(results) <- nam
  results["Mkt"] <- MktName
  results["S Loss"] <- SLossValue
  return(results)
}

calcStats <- function(x){
  # Function to calculate trade stats
  #
  # Args:
  #   x - data set
  #
  # Returns:
  #   results vector.

  results <- 1:3
  v <- na.omit(x)

  # Win %
  wins <- length(v[v>0])
  losses <- length(v[v<0])
  results[1] <- round(wins/(wins+losses)*100)

  # Num Trades
  results[2] <- length(v)

  # Av Long PL
  results[3] <- round(sum(v) / length(v))

  return(results)
}

calcStats2 <- function(x){
  # Function to calculate trade stats
  #
  # Args:
  #   x - data set
  #
  # Returns:
  #   results vector.
  #browser()
  results <- 1:3
  #v <- na.omit(x)
  v <- x

  # Win %
  wins <- sum(v>0,na.rm=T)
  losses <- sum(v<0,na.rm=T)
  results[1] <- round(wins/(wins+losses)*100)

  # Num Trades
  results[2] <- wins+losses
```

```

  # Av Long PL
  results[3] <- round(sum(v,na.rm=T) / (wins+losses))

  return(results)
}

calcWinPer <- function(x){
  wins <- length(x[x>0])
  losses <- length(x[x<0])
  return(wins/(wins+losses)*100)
}

calcAverageWin <- function(x){
  wins <- length(x)
  winpl <- sum(x, na.rm=T)
  return((winpl/wins))
}

calcNumTrades <- function(x){
  return(length(na.omit(x)))
}

savepdf <- function(file, width=16, height=10)
{
  fname <- paste("../Figures/",file,".pdf",sep="")
  pdf(fname, width=width/2.54, height=height/2.54,
      pointsize=10)
  par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(3.3,3.6,1.1,1.1))
}


print_xt <- function(dat,dig,cap,lab,al,filname,inclrnam){
  xt <- xtable(
    dat,
    digits = dig,
    caption = cap,
    label = lab
  )
  al <- c('l','l')
  al <- c(al, rep('c',ncol(dat)-1))
  align(xt) <- al
  print(xt,
        file=filname,
        include.rownames=inclrnam,
        caption.placement = "top",
        hline.after=NULL,
        add.to.row=list(pos=list(-1,0, nrow(xt)),
                        command=c('\\toprule ', '\\midrule ', '\\bottomrule ')))

}


# subtract 2 data frames
# df2 from df1
```

```
sub_df <- function(df1, df2){

  nc <- ncol(df1)
  ln <- nrow(df1)
  dfres <- df1

  for(i in 1:ln){
    for(j in 2:nc){
      dfres[i,j] <- as.numeric(df1[i,j]) - as.numeric(df2[i,j])
    }
  }
  return(dfres)
}

# subtract 2 data frames - rtn fewer cols
# df2 from df1
sub_df_av_pl <- function(df1, df2){

  nc <- ncol(df1)
  ln <- nrow(df1)
  dfres <- df1
  for(i in 1:ln){
    for(j in 2:nc){
      dfres[i,j] <- as.numeric(df1[i,j]) - as.numeric(df2[i,j])
    }
  }
  dfres <- dfres[,c(1,7,10)]
  colnames(dfres) <- c('Mkt','Diff in Mean Long PL','Diff in Mean Short PL')
  return(dfres)
}


# ----------------------------------------------------------------
# ------------ CHAPTER 4 -----------------------------------------
# ----------------------------------------------------------------

# ----------- Follow Previous -------------
run_NaiveReversePrev <- function(fil,SLoss, nm){
  df10 <- as.data.frame(matrix(seq(11),nrow=1,ncol=11))
  for(i in 1:length(fil)){
    Dax <- read.csv(fil[i],stringsAsFactors=F)
    a <- NaiveReversePrev(Dax, SLoss, nm[i])
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-1,]
  return(df10)
}

# ----------------------------------------------------------------
# ------------ CHAPTER 5 -----------------------------------------
# ----------------------------------------------------------------
#  ------ Arima Ann Predicting Up/Dn - Categorical -----------------
# a. Categorical
```

```
ts_4_fnc_ar <- function(fil,SLoss,nm){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors=F)
    Mkt_p <- Mkt[,c(1,2,3,4,5)]
    Mkt_p$pred <- Mkt$pred
    colnames(Mkt_p) <- c("Date","Open", "High","Low","Close","pred")
    a <- ts_4(Mkt_p, SLoss,nm[i])
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-c(1),]
  return(df10)
}


# -----------------------------------------------
#  ------ Arima Ann Predicting Up/Dn - 01 ---------
ts_3_fnc_ar <- function(fil,nm,ts1){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors=F)
    Mkt_p <- Mkt[,c(1,2,3,4,5,18)]
    colnames(Mkt_p) <- c("Date","Open", "High","Low","Close","p")
    a <- ts_3(Mkt_p, 0, nm[i])
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-c(1),]
  return(df10)
}

# bit of fiddling for ANN
ts_3a_fnc_ar <- function(fil,nm,ts1){
  for(i in 1:length(fil)){
    Mkt <- read.csv(fil[i],stringsAsFactors=F)
    Mkt_p <- Mkt[,c(1,2,3,4,5,18)]
    colnames(Mkt_p) <- c("Date","Open", "High","Low","Close","p")
    a <- ts_3a(Mkt_p, 0, nm[i])
    df10 <- rbind(df10, a)
  }
  df.name <- names(a)
  names(df10) <- df.name
  df10 <- df10[-c(1),]
  return(df10)
}
```

RCode/Utils.R

Appendix B

Technical Indicators

B.1 Moving Average Convergence Divergence (MACD)

MACD is a widely used technical indicator which attempts to detect the early stage of a market trend. It is calculated by subtracting a long exponential moving average (EMA) from a shorter one. The EMA is calculated as follows:

$$EMA(n)_t = \frac{2}{n+1}(P_t - EMA_{t-1}) + EMA_{t-1}$$

Where P_t is the closing price of a market on day t and n is the number of periods used in calculating the moving average. MACD itself is calculated as:

$$MACD_t = EMA(s)_t - EMA(l)_t$$

where $EMA(s)_t$ is the short moving average and $EMA(l)_t$ is the long one. In addition an EMA of the MACD itself is calculated in order to generate trade signals and is often referred to as the "trigger line". Thus a particular MACD trading rule is often expressed in the form $MACD(s, l, k)$ where s is the number of periods of the short EMA, l the number of periods of the long EMA and k the period used to average the MACD for the trigger line.

B.2 Aroon Indicator

The Sanskrit word aroon means "dawn's early light" and the Aroon indicator attempts to show when a new market trend is dawning (Chande and Kroll, 1994). The indicator is made up of two lines (Aroon Up and Aroon Down) that measure how long it has been since the highest high and lowest low has occurred within an n period range, and an oscillator value that is the difference between the two. Aroon Up (or Down) is the elapsed time, expressed as a percentage, between today and the highest (or lowest) price in the last n periods. If the current price is a new high (or low) Aroon Up (or Aroon Down) will be 100. Each subsequent period without another new high (or low) causes Aroon up (down) to decrease by (1 / n) x 100.

$$AroonUp = 100 * \left(\frac{n - PeriodSinceHighestHigh}{n}\right)$$

$$AroonDown = 100 * \left(\frac{n - PeriodSinceLowestLow}{n}\right)$$

When the Aroon Up is between a value of 70 and 100 it indicates an upward trend. When the Aroon Down is staying between 70 and 100 then it indicates an downward trend. A strong upward trend is indicated when the Aroon Up is above 70 while the Aroon Down is below 30. Likewise, a strong downward trend is indicated when the Aroon Down is above 70 while the Aroon Up is below 30. Also the crossing over of the lines is significant. When the Aroon Down crosses above the Aroon Up, it indicates a weakening of the upward trend (and vice versa).

The Aroon Oscillator signals that an upward trend is occurring when it is above zero and a downward trend is occurring when it falls below zero. The farther away the oscillator is from the zero line, the stronger the trend.

B.3 Parabolic Stop-and-Reverse (SAR)

The Parabolic Stop-and-Reverse (SAR) is a quite complex indicator developed by Welles Wilder in 1978 (Wilder, 1978). The calculation for SAR in rising and falling markets are different and are usually presented separately.

If the market is rising SAR is calculated as:

$$\text{Current SAR} = \text{Prior SAR} + \text{Prior AF}(\text{Prior EP - Prior SAR})$$

where:

- Prior SAR is the SAR value for the previous time period, for example the previous day's value.
- Extreme Point (EP) is the highest high of the current trend.
- Acceleration Factor (AF) starts at 0.02, and increases by 0.02 each time the market makes a new high (Extreme Point). The maximum value the AF can reach is 0.20, at which point it is capped.

Note: SAR can never be greater than the value of the previous two periods' lows. Should SAR be above one of those lows, it is set to the lowest of the two.

If the market is falling SAR is calculated as:

$$\text{Current SAR} = \text{Prior SAR - Prior AF(Prior SAR - Prior EP)}$$

Note: SAR can never be less than the value of the previous two periods' highs. Should SAR be less than one of those highs, it is set to the lowest of the two.

B.4 Stochastic

The stochastic oscillator measures where a particular close price is in relation to the highest high and lowest low in the range under study. It is usually drawn on a chart as two lines, one is %K and the other is its moving average usually called %D.

The calculation of the stochastic involves four variables:

1. %K Period - the number of periods used in the calculation (see below).
2. %K Slowing Period - smoothing period applied to %K.
3. %D Period - the number of time periods used in the moving average of %K to generate %D.
4. %D Method - the moving average method used to calculate %D.

%K is calculated as follows:

$$\%K = 100 * \left(\frac{\text{Today's Close} - \text{Lowest Low in n Periods}}{\text{Highest High in n Periods} - \text{Lowest Low in n Periods}} \right)$$

The stochastic is used in a variety of ways. One popular method is to buy when the stochastic falls below a particular level then rises back above that level (and vice versa for a short trade). An alternative technique is to buy when the %K rises above %D and sell when it falls under %K.

B.5 Rate of Change(ROC)

The Rate of Change or ROC indicator highlights the difference between a particular price (e.g. closing price) and the same price a number of periods previously. This value can be expressed in absolute terms or a percentage rise or fall. The calculation is as follows:

$$ROC = 100 * \left(\frac{\text{Today's Close} - \text{Today's Close n Periods Ago}}{\text{Today's Close n Periods Ago}} \right)$$

The ROC can be calculated from a wide range of time periods, with 12 and 25 days being the most common. The ROC is typically used as an over-bought / over-sold indicator to provide evidence for when a market turn maybe expected.

Appendix C

Summary of Results

C.1 Chapter 4 Results

Table C.1: Chapter 5 DAX Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	-1640	-1505	-1	-1
Drift Method	2310	2445	1	2
Exponential Smoothing	-2029	-1894	-1	-1
ARIMA - System 1	-1285	-2522	-5	-9
ARIMA - System 2	733	-505	3	-2
ARIMA/ANN Closing Price System 1	-446	-645	-1	-2
ARIMA/ANN Closing Price System 2	3283	3110	6	7
ARIMA/k-NN Closing Price System 1	12	-174	0	0
ARIMA/k-NN Closing Price System 2	489	731	1	2
ARIMA/ANN Up/Down	199	0	0	NaN
ARIMA/k-NN Up/Down	-1553	-1752	-3	-4
ARIMA/ANN Up/Down Stop Loss	-430	-1444	-1	-4
ARIMA/SVM Up/Down	-123	-322	0	-1

Table C.2: Chapter 4 CAC Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Naive Long	-6725	0	-2	0
Naive Long 2	-1667	0	0	0
Reverse Prev	940	7810	1	4
Reverse Prev Stop Loss	1335	8165	1	5
SMA 5	-3952	2338	-2	1
SMA 25	-5058	1615	-2	1
SMA 50	-5323	1029	-3	1
SMA 100	-2363	3188	-1	2
SMA 200	-1219	3923	-1	3
SMA 100	-172	5178	0	4
SMA 100	-1822	4658	-1	3
MACD	-4153	2188	-2	1
Aroon	-1638	4919	-1	4
Aroon Stop Loss	-1224	6086	-1	5
SAR	-5584	1034	-3	1
MACD Reversal	-545	2657	-1	5
Stoch	-4540	1817	-3	1
Stoch Stop Loss	-3493	2730	-2	2
ROC	952	956	2	2
Daily Breakout	3491	6955	2	4
90% Quantile Breakout	2647	5085	2	3
Hammer Candlestick	-793	0	-5	0
Engulfing Candlestick	-319	228	-2	1
Engulfing Candlestick in Trend	-118	-666	-3	-11
Doji Candlestick	-747	-326	-6	-2

Table C.3: Chapter 4 FTSE Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Naive Long	149	0	0	0
Naive Long 2	86	0	0	0
Reverse Prev	4284	4115	3	2
Reverse Prev Stop Loss	5537	5200	3	3
SMA 5	-4724	-5331	-2	-3
SMA 25	-1013	-1940	0	-1
SMA 50	-2226	-2769	-1	-2
SMA 100	-889	-1692	0	-1
SMA 200	-158	-835	0	-1
SMA 100	1114	6303	1	5
SMA 100	-885	1892	0	1
MACD	63	-839	0	0
Aroon	3042	5715	2	5
Aroon Stop Loss	3091	8015	2	7
SAR	-1141	-1663	-1	-1
MACD Reversal	2080	1649	4	3
Stoch	-73	-744	0	0
Stoch Stop Loss	1640	1424	1	1
ROC	1147	1880	2	4
Daily Breakout	13189	18481	7	12
90% Quantile Breakout	10758	15295	7	10
Hammer Candlestick	834	0	4	0
Engulfing Candlestick	-1721	1185	-4	3
Engulfing Candlestick in Trend	-1217	-782	-8	-3
Doji Candlestick	-697	418	-8	3

Table C.4: Chapter 4 Dow Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Naive Long	9816	0	3	0
Naive Long 2	5219	0	1	0
Reverse Prev	15799	6047	10	3
Reverse Prev Stop Loss	-8571	-14604	-5	-8
SMA 5	408	-9630	0	-6
SMA 25	1138	-9204	1	-7
SMA 50	5478	-5876	3	-4
SMA 100	2576	-8220	1	-6
SMA 200	6378	-4567	3	-4
SMA 100	-18212	-8229	-9	-6
SMA 100	-11771	-14696	-6	-11
MACD	5592	-5190	3	-3
Aroon	12131	3811	7	3
Aroon Stop Loss	-5922	-9341	-3	-8
SAR	-1301	-11112	-1	-7
MACD Reversal	3882	-807	7	-2
Stoch	867	-9414	0	-5
Stoch Stop Loss	-13969	-27388	-8	-16
ROC	8517	3396	16	6
Daily Breakout	-19598	-28337	-11	-17
90% Quantile Breakout	-30262	-34854	-24	-28
Hammer Candlestick	2097	0	24	0
Engulfing Candlestick	-770	-3662	-4	-28
Engulfing Candlestick in Trend	202	-1154	4	-11
Doji Candlestick	-763	-2869	-5	-10

Table C.5: Chapter 4 Nikkei Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Naive Long	-18125	0	-5	0
Naive Long 2	-2712	0	-1	0
Reverse Prev	2324	20486	1	12
Reverse Prev Stop Loss	18137	27909	10	17
SMA 5	3078	20401	2	13
SMA 25	-7878	10770	-4	7
SMA 50	-6054	11408	-4	7
SMA 100	-6235	8381	-4	5
SMA 200	-5928	6836	-4	4
SMA 100	8258	33882	5	20
SMA 100	2550	25582	2	15
MACD	-4078	14064	-2	8
Aroon	-4852	12013	-3	10
Aroon Stop Loss	3153	22177	2	18
SAR	-5767	12424	-3	8
MACD Reversal	199	2828	0	6
Stoch	-10591	7802	-6	5
Stoch Stop Loss	1647	17977	1	10
ROC	2971	2546	6	5
Daily Breakout	31988	43554	19	27
90% Quantile Breakout	23606	31830	16	20
Hammer Candlestick	-2202	0	-15	0
Engulfing Candlestick	-3823	-1166	-39	-11
Engulfing Candlestick in Trend	-1522	-1733	-59	-32
Doji Candlestick	1296	-2944	12	-22

Table C.6: Chapter 4 AORD Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Naive Long	972	0	0	0
Naive Long 2	2229	0	1	0
Reverse Prev	1264	237	1	0
Reverse Prev Stop Loss	2320	1085	1	1
SMA 5	5009	3929	3	3
SMA 25	3701	2674	2	2
SMA 50	2804	1864	1	1
SMA 100	2688	1521	1	1
SMA 200	2574	1616	1	2
SMA 100	4008	3730	2	3
SMA 100	2881	2149	1	2
MACD	2563	1569	1	1
Aroon	3735	3540	2	3
Aroon Stop Loss	3786	4159	2	4
SAR	2071	1097	1	1
MACD Reversal	-319	-584	-1	-1
Stoch	2839	1780	2	1
Stoch Stop Loss	3028	1974	2	1
ROC	271	1325	1	2
Daily Breakout	17225	19184	10	13
90% Quantile Breakout	16730	19357	9	12
Hammer Candlestick	-809	0	-3	0
Engulfing Candlestick	-6	-600	0	-3
Engulfing Candlestick in Trend	-49	-27	-1	0
Doji Candlestick	-115	195	-1	2

C.2 Chapter 5 Results

TABLE C.7: Chapter 5 DAX Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	-1640	-1505	-1	-1
Drift Method	2310	2445	1	2
Exponential Smoothing	-2029	-1894	-1	-1
ARIMA - System 1	-1285	-2522	-5	-9
ARIMA - System 2	733	-505	3	-2
ARIMA/ANN Closing Price System 1	-446	-645	-1	-2
ARIMA/ANN Closing Price System 2	3283	3110	6	7
ARIMA/k-NN Closing Price System 1	12	-174	0	0
ARIMA/k-NN Closing Price System 2	489	731	1	2
ARIMA/ANN Up/Down	199	0	0	NaN
ARIMA/k-NN Up/Down	-1553	-1752	-3	-4
ARIMA/ANN Up/Down Stop Loss	-430	-1444	-1	-4
ARIMA/SVM Up/Down	-123	-322	0	-1

TABLE C.8: Chapter 5 CAC Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	-1086	3553	-1	2
Drift Method	-2422	2217	-1	2
Exponential Smoothing	-266	4048	0	2
ARIMA - System 1	872	167	3	1
ARIMA - System 2	545	-80	2	0
ARIMA/ANN Closing Price System 1	532	1527	2	2
ARIMA/ANN Closing Price System 2	-1832	-816	-3	-2
ARIMA/k-NN Closing Price System 1	-684	312	-1	1
ARIMA/k-NN Closing Price System 2	-1458	-441	-3	-1
ARIMA/ANN Up/Down	0	996	NaN	1
ARIMA/k-NN Up/Down	270	1265	1	2
ARIMA/ANN Up/Down Stop Loss	203	1326	0	2
ARIMA/SVM Up/Down	-140	855	0	1

5

Table C.9: Chapter 5 FTSE Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	1680	345	1	0
Drift Method	-518	-1853	0	-1
Exponential Smoothing	3866	2531	3	2
ARIMA - System 1	990	-249	4	-1
ARIMA - System 2	941	-383	3	-2
ARIMA/ANN Closing Price System 1	625	-624	1	-2
ARIMA/ANN Closing Price System 2	1092	-182	2	0
ARIMA/k-NN Closing Price System 1	699	-550	1	-1
ARIMA/k-NN Closing Price System 2	-388	-1662	-1	-4
ARIMA/ANN Up/Down	1400	151	2	0
ARIMA/k-NN Up/Down	1764	515	3	1
ARIMA/ANN Up/Down Stop Loss	1919	526	4	1
ARIMA/SVM Up/Down	2115	866	5	1

Table C.10: Chapter 5 Dow Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	8356	-2126	7	-1
Drift Method	5416	-5066	3	-4
Exponential Smoothing	12901	2419	8	2
ARIMA - System 1	1539	-3356	7	-11
ARIMA - System 2	2598	-2221	9	-10
ARIMA/ANN Closing Price System 1	2846	-3979	5	-9
ARIMA/ANN Closing Price System 2	3829	-2942	7	-7
ARIMA/k-NN Closing Price System 1	4436	-2389	11	-4
ARIMA/k-NN Closing Price System 2	2969	-3411	5	-9
ARIMA/ANN Up/Down	5218	-1607	6	-10
ARIMA/k-NN Up/Down	801	-6024	1	-14
ARIMA/ANN Up/Down Stop Loss	3076	-2698	6	-6
ARIMA/SVM Up/Down	2138	-4686	10	-6

Table C.11: Chapter 5 Nikkei Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	-32	10646	0	6
Drift Method	-6939	3739	-4	3
Exponential Smoothing	-2741	7937	-2	5
ARIMA - System 1	4268	3071	21	13
ARIMA - System 2	179	-916	1	-4
ARIMA/ANN Closing Price System 1	913	2039	2	4
ARIMA/ANN Closing Price System 2	-4485	-3229	-9	-7
ARIMA/k-NN Closing Price System 1	-66	1060	0	2
ARIMA/k-NN Closing Price System 2	-2916	-1660	-6	-4
ARIMA/ANN Up/Down	234	1360	12	1
ARIMA/k-NN Up/Down	2707	3834	6	8
ARIMA/ANN Up/Down Stop Loss	4021	2804	9	6
ARIMA/SVM Up/Down	9	1135	0	2

Table C.12: Chapter 5 AORD Results

Methodology	LongPL	ShortPL	Av L PL	Av S PL
Mean Method	-1333	-2149	-1	-1
Drift Method	1476	660	1	1
Exponential Smoothing	645	-171	0	0
ARIMA - System 1	635	-247	2	-1
ARIMA - System 2	811	-117	3	0
ARIMA/ANN Closing Price System 1	-3036	-371	-5	-1
ARIMA/ANN Closing Price System 2	-2783	-137	-5	0
ARIMA/k-NN Closing Price System 1	497	3162	1	7
ARIMA/k-NN Closing Price System 2	-3449	-804	-6	-2
ARIMA/ANN Up/Down	-2712	-47	-3	-12
ARIMA/k-NN Up/Down	748	3413	1	7
ARIMA/ANN Up/Down Stop Loss	570	3424	1	7
ARIMA/SVM Up/Down	-2364	301	-4	1

Bibliography

Achelis, S. B. Macd, 2014. URL `http://www.metastock.com/Customer/Resources/TAAZ/default.aspx?c=3&p=66`.

Aladag, C. H., E. Egrioglu, and C. Kadilar. Forecasting nonlinear time series with a hybrid methodology. *Applied Mathematics Letters*, 22(9):1467 – 1470, 2009.

Appel, G. *Technical Analysis: Power Tools for Active Investors : [new Techniques for Active Trading in the Stock Market!]*. Financial Times Prentice Hall books. Financial Times/Prentice Hall, 2005. ISBN 9780131479029.

Appel, G. and E. Dobson. *Understanding Moving Average Convergence-divergence*. Traders Press, 2007. ISBN 9780934380928.

Atsalakis, G. S. and K. P. Valavanis. Surveying stock market forecasting techniques – part ii: Soft computing methods. *Expert Systems with Applications*, 36(3):5932 – 5941, 2009.

Bajgrowicz, Pierre and Olivier Scaillet. Technical trading revisited: False discoveries, persistence tests, and transaction costs. *Journal of Financial Economics*, 106(3):473 – 491, 2012. ISSN 0304-405X. doi: http://dx.doi.org/10.1016/j.jfineco.2012.06.001. URL `http://www.sciencedirect.com/science/article/pii/S0304405X1200116X`.

Beau, C.L. and D.W. Lucas. *Day Trading Systems and Methods*. Marketplace Books, 1999. ISBN 9781883272272.

Bigalow, S.W. *Profitable Candlestick Trading: Pinpointing Market Opportunities to Maximize Profits*. Wiley Trading. Wiley, 2011. ISBN 9781118005804.

Box, G.E.P. and G.M. Jenkins. *Time series analysis: forecasting and control*. Holden-Day, 1970.

Brock, W., J. Lakonishok, and B. LeBaron. Two simple technical trading rules and the stochastic properties of stock returns. *Journal of Finance*, 47(5):1731–64, 1992.

Brown, R.G. *Statistical forecasting for inventory control.* McGraw-Hill, 1959.

Brown, R.G. *Smoothing, forecasting and prediction of discrete time series.* Prentice-Hall, 1963.

Bulkowski, T.N. *Encyclopedia of Chart Patterns.* Wiley Trading. Wiley, 2011. ISBN 9781118045855.

Chande, T.S. and S. Kroll. *The new technical trader: boost your profit by plugging into the latest indicators.* Wiley Finance. Wiley, 1994. ISBN 9780471597803.

Chen, David Y. and Tongzhe Li. Financial crises, asian stock indices, and current accounts: An asian-u.s. comparative study. *Journal of Asian Economics*, 34(0):66 – 78, 2014. ISSN 1049-0078. doi: http://dx.doi.org/10.1016/j.asieco.2014.06.002. URL `http://www.sciencedirect.com/science/article/pii/S1049007814000402`.

Chopra, N., J. Lakonishok, and J. Ritter. Measuring abnormal performance do stocks overreact? *Journal of Financial Economics*, 31:235 – 268, 1992.

Covel, M.W. *Trend Following (Updated Edition): Learn to Make Millions in Up or Down Markets.* Pearson Education, 2009. ISBN 9780135094402.

De Bondt, W. and R. Thaler. Does the stock market overreact? *Journal of Finance*, 40 (0):793 – 805, 1985.

De Gooijer, J. G. and R. J. Hyndman. 25 years of time series forecasting. *International Journal of Forecasting*, 22(3):443 – 473, 2006.

Devcic, J. Weighted moving averages: The basics, 2010. URL `http://www.investopedia.com/articles/technical/060401.asp`. [Online; accessed 1-March-2014].

Dueker, M. and C.J. Neely. Can markov switching rules predict foreign exchange returns? *Journal of Banking and Finance*, 31:297 – 296, 2007.

Efron, B. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 71:1–26, 1979.

Faith, C. *Way of the Turtle: The Secret Methods that Turned Ordinary People into Legendary Traders.* McGraw Hill professional. McGraw-Hill Education, 2007. ISBN 9780071509466.

Fama, Eugene F. and Kenneth R. French. Size, value, and momentum in international stock returns. *Journal of Financial Economics*, 105(3):457 – 472, 2012. ISSN 0304-405X. doi: http://dx.doi.org/10.1016/j.jfineco.2012.05.011. URL `http://www.sciencedirect.com/science/article/pii/S0304405X12000931`.

Fang, Jiali, Ben Jacobsen, and Yafeng Qin. Predictability of the simple technical trading rules: An out-of-sample test. *Review of Financial Economics*, 23(1):30 – 45, 2014. ISSN 1058-3300. doi: http://dx.doi.org/10.1016/j.rfe.2013.05.004. URL http://www.sciencedirect.com/science/article/pii/S1058330013000396.

Fatima, S. and G Hussain. Statistical models of kse100 index using hybrid financial systems. *Neurocomputing*, 71(13–15):2742 – 2746, 2008.

Fiess, N. M. and R. MacDonald. Towards the fundamentals of technical analysis: analysing the information content of high, low and close prices. *Economic Modelling*, 19(3):353 – 374, 2002.

Friesen, G. C., P. A. Weller, and L. M. Dunham. Price trends and patterns in technical analysis - a theoretical and empirical examination. *Journal of Banking and Finance*, 33(6):1089 – 1100, 2009.

Gutierrez Jr, R. and E. Kelley. The long-lasting momentum in weekly returns. *Journal of Finance*, 63:415 – 447, 2008.

Han, J., M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques: Concepts and Techniques*. Elsevier Science, 2011. ISBN 9780123814807.

Holt, C. C. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5 – 10, 2004.

Hsu, C.M. A hybrid procedure for stock price prediction by integrating self-organizing map and genetic programming. *Expert Systems with Applications*, 38(11):14026 – 14036, 2011.

Hsu, Po-Hsuan, Yu-Chin Hsu, and Chung-Ming Kuan. Testing the predictive ability of technical analysis using a new stepwise test without data snooping bias. *Journal of Empirical Finance*, 17(3):471 – 484, 2010. ISSN 0927-5398. doi: http://dx.doi.org/10.1016/j.jempfin.2010.01.001. URL http://www.sciencedirect.com/science/article/pii/S0927539810000022.

Huang, W., Y. Nakamori, and S-Y. Wang. Forecasting stock market movement direction with support vector machine. *Computers and Operations Research*, 32(10):2513 – 2522, 2005.

Hyndman, R. J and K. Yeasmin. Automatic time series forecasting: The forecast package for r. *Journal of Statistical Software*, 2008.

Jegadeesh, N. Evidence of predictable behavior of security returns. *Journal of Finance*, 45:881 – 898, 1990.

Jegadeesh, N. and S. Titman. Returns to buying winners and selling losers: Implications for stock market efficiency. *Journal of Finance*, 48:65 – 91, 1993.

Jegadeesh, N. and S. Titman. Overreaction, delayed reaction and contrarian profits. *Review of Financial Studies*, 8:973 – 993, 1995.

Kaufman, P.J. *Trading Systems and Methods*. Wiley, 2013. ISBN 9781118222249.

Khashei, M and M. Bijari. A new class of hybrid models for time series forecasting. *Expert Systems with Applications*, 39(4):4344 – 4357, 2012.

Khashei, M., M. Bijari, and G. A. R. Ardali. Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (anns). *Neurocomputing*, 72(4–6):956 – 967, 2009.

Kriechbaumer, T., A. Angus, D. Parsons, and M. R. Casado. An improved wavelet–arima approach for forecasting metal prices. *Resources Policy*, 39(0):32 – 41, 2014.

Kuang, P., M. Schroder, and Q. Wang. Illusory profitability of technical analysis in emerging foreign exchange markets. *International Journal of Forecasting*, 30(2):192 – 205, 2014. ISSN 0169-2070. doi: http://dx.doi.org/10.1016/j.ijforecast.2013.07.015. URL `http://www.sciencedirect.com/science/article/pii/S0169207013000964`.

Lane, G.C. *Using Stochastics, Cycles and R.S.I.* 1986.

Lehmann, B. Fads, martingales and market efficiency. *Quarterly Journal of Economics*, 105:1 – 28, 1990.

Levich, R. and L. Thomas. The significance of technical trading rule profits in the foreign exchange market: A bootstrap approach. *Journal of International Money and Finance*, 12:451 – 474, 1993.

Lu, Tsung-Hsun. The profitability of candlestick charting in the taiwan stock market. *Pacific-Basin Finance Journal*, 26(0):65 – 78, 2014. ISSN 0927-538X. doi: http://dx.doi.org/10.1016/j.pacfin.2013.10.006. URL `http://www.sciencedirect.com/science/article/pii/S0927538X13000735`.

Malkiel, B.G. *A Random Walk Down Wall Street: Including a Life-cycle Guide to Personal Investing*. Norton, 1999. ISBN 9780393047813.

Marshall, B. R. and R. H. Cahan. Is technical analysis profitable on a stock market which has characteristics that suggest it may be inefficient? *Research in International Business and Finance*, 19(3):384 – 398, 2005.

Marshall, B. R, M. R. Young, and L. C. Rose. Candlestick technical trading strategies: Can they create value for investors? *Journal of Banking and Finance*, 30(8):2303 – 2323, 2006.

Marshall, B. R., R. H. Cahan, and J. M. Cahan. Does intraday technical analysis in the u.s. equity market have value? *Journal of Empirical Finance*, 15(2):199 – 210, 2008.

Matheson, Troy D. Financial conditions indexes for the united states and euro area. *Economics Letters*, 115(3):441 – 446, 2012. ISSN 0165-1765. doi: http://dx.doi.org/10.1016/j.econlet.2011.12.119. URL `http://www.sciencedirect.com/science/article/pii/S0165176511006422`.

Menkhoff, L. The use of technical analysis by fund managers: International evidence. *Journal of Banking and Finance*, 34(11):2573 – 2586, 2010.

Menkhoff, Lukas, Lucio Sarno, Maik Schmeling, and Andreas Schrimpf. Currency momentum strategies. *Journal of Financial Economics*, 106(3):660 – 684, 2012. ISSN 0304-405X. doi: http://dx.doi.org/10.1016/j.jfineco.2012.06.009. URL `http://www.sciencedirect.com/science/article/pii/S0304405X12001353`.

Mills, T. C. *The Foundations of Modern Time Series Analysis*. Palgrave Macmillan, 2011. ISBN 9780230290181.

Morris, G.L. *Candlestick Charting Explained: Timeless Techniques for Trading stocks and Sutures*. McGraw-Hill Education, 2006. ISBN 9780071632171.

Moskowitz, Tobias J., Yao Hua Ooi, and Lasse Heje Pedersen. Time series momentum. *Journal of Financial Economics*, 104(2):228 – 250, 2012. ISSN 0304-405X. doi: http://dx.doi.org/10.1016/j.jfineco.2011.11.003. URL `http://www.sciencedirect.com/science/article/pii/S0304405X11002613`. Special Issue on Investor Sentiment.

Murphy, J.J. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. Penguin Group US, 1999. ISBN 9781101659199.

Neely, C. J., P. Weller, and R. Dittmar. Is technical analysis in the foreign exchange market profitable? a genetic programming approach. *Journal of Financial and Quantitative Analysis*, 32:405 – 426, 1997.

Nison, S. *Japanese Candlestick Charting Techniques: A Contemporary Guide to the Ancient Investment Techniques of the Far East*. New York Institute of Finance, 2001. ISBN 9780735201811.

Novy-Marx, Robert. Is momentum really momentum? *Journal of Financial Economics*, 103(3):429 – 453, 2012. ISSN 0304-405X. doi: http://dx.doi.org/10.1016/j.

jfineco.2011.05.003. URL `http://www.sciencedirect.com/science/article/pii/S0304405X11001152`.

Ord, K. Charles holt's report on exponentially weighted moving averages: an introduction and appreciation. *International Journal of Forecasting*, 20(1):1 – 3, 2004.

Pai, P-F. and C-S. Lin. A hybrid arima and support vector machines model in stock price forecasting. *Omega*, 33(6):497 – 505, 2005.

Qian, G. and X. Zhao. On time series model selection involving many candidate arma models. *Computational Statistics and Data Analysis*, 51(12):6180 – 6196, 2007.

Reitz, S. On the predictive content of technical analysis. *The North American Journal of Economics and Finance*, 17(2):121 – 137, 2006.

Rojas, I., O. Valenzuela, F. Rojas, A. Guillen, L.J. Herrera, H. Pomares, L. Marquez, and M. Pasadas. Soft-computing techniques and arma model for time series prediction. *Neurocomputing*, 71(4-6):519 – 537, 2008.

Rout, M., B. Majhi, R. Majhi, and G. Panda. Forecasting of currency exchange rates using an adaptive arma model with differential evolution based training. *Journal of King Saud University - Computer and Information Sciences*, 26(1):7 – 18, 2014.

Schulmeister, S. Profitability of technical stock trading: Has it moved from daily to intraday data? *Review of Financial Economics*, 18(4):190 – 201, 2009.

Schwager, J.D. *Market Wizards: Interviews With Top Traders*. HarperCollins, 1988. ISBN 978-0887306105.

Schwager, J.D. *The New Market Wizards: Conversations with America's Top Traders*. HarperBusiness, 1994. ISBN 978-0887306679.

Shumway, R.H. and D.S. Stoffer. *Time Series Analysis and Its Applications: With R Examples*. Springer, 2010. ISBN 9781441978653.

Shynkevich, A. Performance of technical analysis in growth and small cap segments of the us equity market. *Journal of Banking and Finance*, 36(1):193 – 208, 2012.

So, M. K. P. and R. S. W. Chung. Dynamic seasonality in time series. *Computational Statistics and Data Analysis*, 70(0):212 – 226, 2014.

Sweeney, R. Beating the foreign exchange market. *Journal of Finance*, 41:163 – 182, 1986.

Tan, Z., J. Zhang, J. Wang, and J. Xu. Day-ahead electricity price forecasting using wavelet transform combined with arima and garch models. *Applied Energy*, 87(11): 3606 – 3610, 2010.

Tanaka-Yamawaki, M. and S. Tokuoka. Adaptive use of technical indicators for the prediction of intra-day stock prices. *Physica A: Statistical Mechanics and its Applications*, 383(1):125 – 133, 2007.

Taylor, Mark P. and Helen Allen. The use of technical analysis in the foreign exchange market. *Journal of International Money and Finance*, 11(3):304 – 314, 1992. ISSN 0261-5606. doi: http://dx.doi.org/10.1016/0261-5606(92)90048-3. URL `http://www.sciencedirect.com/science/article/pii/0261560692900483`.

Taylor, Nick. The rise and fall of technical trading rule success. *Journal of Banking & Finance*, 40(0):286 – 302, 2014. ISSN 0378-4266. doi: http://dx.doi.org/10.1016/j.jbankfin.2013.12.004. URL `http://www.sciencedirect.com/science/article/pii/S0378426613004664`.

Ulku, N. and E. Prodan. Drivers of technical trend-following rules profitability in world stock markets. *International Review of Financial Analysis*, 30:214 – 229, 2013.

Wang, J-J., J-Z. Wang, Z-G. Zhang, and S-P. Guo. Stock index forecasting based on a hybrid model. *Omega*, 40(6):758 – 766, 2012.

Weissman, R.L. *Mechanical Trading Systems: Pairing Trader Psychology with Technical Analysis*. Wiley, 2005. ISBN 9780471654353.

Wilder, J.W. *New concepts in technical trading systems*. Trend Research, 1978. ISBN 9780894590276.

Williams, L. *Long-Term Secrets to Short-Term Trading*. Wiley Trading. Wiley, 2011. ISBN 9781118184684.

Winters, P. R. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342, 1960.

Zhang, G. P. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50(0):159 – 175, 2003.