PROBABILISTIC FORECASTING MODELS FOR MULTIDIMENSIONAL FINANCIAL TIME-SERIES

With Applications to Systematic Portfolio Management

Dimitri Malandreniotis

A thesis submitted for the part fulfillment of the requirements for the degree of Doctor of Philosophy

Department of Computer Science
Faculty of Engineering
University College London
March – 2023
Dimitri Malandreniotis: Probabilistic Forecasting Models for Multidimensional Financial Time-series With Applications to Systematic Portfolio Management,

SUPERVISOR:
Denise Gorse
Declaration

I, Dimitri Malandreniotis, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

*London, United Kingdom, March*

_________________________
Dimitri Malandreniotis
The light that burns twice as bright burns half as long
— and you have burned so very, very brightly, ...

— Dr Eldon Tyrell

Dedicated to the loving memory of Dimitris Tzamalis.
1979 – 2001
Abstract

In the current era of financial markets, the emergence of autonomous trading systems and systematic strategies has significantly transformed the landscape of asset management. Among these developments, dynamic asset allocation is a strategy that manages a portfolio systematically, adjusting asset positions in response to ever-changing market conditions. At the heart of these strategies is a forecasting model responsible for quantifying the uncertainty of the assets’ future price movements. The quality of continuously updated probabilistic forecasts is of paramount importance for effectively managing risk, generating returns, and is crucial for success. Systematic strategies, such as these, which are designed to require frequent trading, are inherently costly; this can significantly impair the performance of a strategy, making it prohibitively expensive to operate. This thesis addresses these two limiting factors for effective dynamic asset allocation. The primary contribution of the thesis, extending over the first two research chapters, is a novel modelling methodology for forecasting the joint distributions of entire portfolios. Leveraging gradient boosting within a distributional regression framework, this methodology allows all parameters of the multivariate distribution to be time-varying conditioned on high-dimensional sets of exogenous variables, with additional structural time-variation provided by regime-switching. This framework is capable of capturing well-known statistical properties of asset returns, including time-varying covariance, higher-order co-moments, asymmetries in shape and dependence, among other stylised facts. Experiments are undertaken to evaluate the efficacy and feasibility of applying this model in systematic strategies for managing equity portfolios. The second major contribution of this thesis addresses the impact that transaction costs have on the performance of portfolio strategies. It introduces a novel method for portfolio optimisation, one that takes into consideration the potential costs of portfolio rebalancing. The performance improvements resulting from the inclusion of this new method are demonstrated to be substantial when compared to traditional portfolio optimisation techniques.
Impact Statement

Asset management is the practice of investing into financial securities for the purpose of protecting and growing the wealth of individuals, businesses, pension funds, university endowments or, in rare cases government, and is an important part of a capitalist society. The failure of management to adequately assess risk involved in the practice can have devastating consequences to the investors and in some cases, fallout can be systemic. For many years, we have seen these consequences first hand as financial shocks have a tendency to echo through our economy affecting society as a whole. Previous cases of documented mismanagement have been attributed to a misunderstanding of how financial asset prices behave. This primarily has the consequence of underestimating a portfolio’s exposure to different types of risk. Statistical models in the 90s and 00s have, in hindsight, been criticised for making coarse assumptions about the statistical properties of financial price returns, and although these models may appear adequate for most market conditions, they’ve proven not to be robust at times when market conditions fail to comply with the assumptions. Although over the last decade substantial advances have been made, the research I have conducted aims to contribute to the state of the art in statistical modelling of financial assets and to provide models for better understanding the joint behaviour of assets in a managed portfolio;

The task of making optimal decisions under uncertainty is not one that the human brain has been designed to do well. The Nobel prize has been awarded for research which showed that we, as humans, are not capable of correctly assessing the risks we undertake. Asset management is a risky business and therefore, given our human limitations, algorithmic procedures that make decisions to assist, or in place of people, have become more common place over the past decade. Part of my research is in advancements of a sub-type of autonomous trading systems, commonly referred to as systematic trading strategies. These systematic strategies over time may prove to make investment in tradable assets safer by reducing risk to one’s capital by eliminating the human factor from decision making; however, their new ubiquity may also introduce new types of risk to the financial markets and create new problems for asset managers. The overall impact this research will have is to add to the understanding of the financial markets as a whole which is an essential underpinning pillar of our economy.
Finally, the autonomous portfolio management systems developed in this research have commercial value. They are designed and tested under no restrictive assumptions and as will be discussed in detail, can be applied to real world trading with ease. The value to enterprise could be in the retail trading (day trader) space or, as a portfolio management strategy used by an investment fund.
“I never guess. It is a capital mistake to theorise before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts.”

(Sherlock Holmes — “A Scandal in Bohemia”, Doyle (1892))

— Sir Arthur Conan-Doyle, British mystery author & physician (1859 –1930)

Acknowledgments

There are many reasons why I would like to acknowledge the impact that my supervisor, Dr Denise Gorse, has had on my many years at UCL, though I know it will make her embarrassed, and hopefully even make her blush. Her strict, uncompromising demand for the correct use of the English written language has reduced me on many an occasion to feeling like an illiterate buffoon; alas, I very much appreciate that this has been an important, albeit an unforeseen, part of my learning process. I also would like to acknowledge and thank her for the sheer amount of time she has made available for meetings, time that is far above and beyond what could ever be expected from a Ph.D. supervisor, and which also included enjoyable time spent on discussing books and movies and, of course, the ergonomic qualities of german-made office furniture; furthermore, I would like to thank her for the emotional and pastoral support she has provided throughout the plentiful times of woe and despair, and thank her for being my eyes after mine stopped working; I deeply appreciate all you have done for me, Denise, thank you. Additionally, I would like to acknowledge Prof. Philip Treleaven for giving me the opportunity to attend doctoral training at UCL, and for dissuading me from pursuing my life-long dream of launching an online bookshop startup, and acknowledge also all those others who kept the FCA DTC running, along with the financial support from a Department of Computer Science studentship award. Before my time at UCL, I would also like to express my gratitude to Dr. Vinay Nundlall of Royal Holloway for his significant impact on my education. Additionally, I am deeply appreciative of Prof. Marek Capinski at AGH and Prof. Tomasz Zastawniak at the University of York for their faith in my abilities to pursue postgraduate mathematics despite not having an undergraduate degree in math. I would also like to extend my gratitude to my parents, Kerrie and Harry and brother Robbie; I am sure that they are all extremely proud that I, after undergraduate study, three Master’s degrees, a Ph.D., lengthy periods of dithering, and procrastination to rival that of George R. R. Martin’s, has finally completed university; thank you for all your love and support. Speaking of whom, I would also like to thank George R. R. Martin for being so graceful in defeat after losing our bet; George, thanks for being a good sport, I’m sure you too will have your manuscript finished soon. Finally, but most importantly, of course, I would like to thank my wife, Wioletta, for allowing, with loving support and minimal complaint, her husband to be a student when all our friends are grown-ups with proper (paying) jobs and go on nice holidays; I know I have a lot of making up to do, starting now, I promise.
## Contents

### i Preliminaries

1 Introduction
   1.1 Statement of the Problem ........................................... 3
   1.2 Research Aims ......................................................... 4
   1.3 Research Objectives .................................................. 6
      1.3.1 Probabilistic Forecasting ...................................... 6
      1.3.2 Operational Cost Minimisation .................................. 8
   1.4 Scope ........................................................................ 8
   1.5 Significance .............................................................. 9
   1.6 Structure of the Document ............................................. 10

2 Motivation, Background and Related Work
   2.1 Overview of Systematic Trading ...................................... 11
   2.2 Probabilistic Forecasting .............................................. 14
   2.3 Portfolio Management .................................................. 17
   2.4 Modelling Methods ..................................................... 22
   2.5 Distributional Regression ............................................. 29

3 Technical Background
   3.1 Regression Frameworks ............................................... 33
      3.1.1 Linear Models ....................................................... 35
      3.1.2 Generalised Linear Models ...................................... 36
      3.1.3 Generalised Additive Models .................................... 38
      3.1.4 The Generalised Additive Model For Location, Scale and Shape .................. 41
   3.2 Gradient Boosting ...................................................... 48
      3.2.1 Boosting as Functional Gradient Descent ..................... 52
      3.2.2 Component-wise Boosting ...................................... 53
      3.2.3 Gradient Boosting for GAMLSS ................................. 57
   3.3 Markov-switching Regression Models ............................... 60
      3.3.1 Overview of Regime-switching ................................ 60
      3.3.2 Hidden Markov Models .......................................... 64
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.3 Markov-switching Distributional Regression Models</td>
<td>74</td>
</tr>
<tr>
<td>ii Probabilistic Time-series Modelling</td>
<td></td>
</tr>
<tr>
<td>4 Pairs Trading with the Bivariate MS-GAMLSS</td>
<td>82</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>83</td>
</tr>
<tr>
<td>4.2 Model Design and Estimation</td>
<td>85</td>
</tr>
<tr>
<td>4.2.1 Bivariate Extension for a Gaussian GAMLSS Model</td>
<td>85</td>
</tr>
<tr>
<td>4.2.2 Markov-switching Bivariate GAMLSS with Component-wise Gradient Boosting</td>
<td>88</td>
</tr>
<tr>
<td>4.2.3 Base Learners</td>
<td>92</td>
</tr>
<tr>
<td>4.2.4 Model Selection</td>
<td>93</td>
</tr>
<tr>
<td>4.3 Data</td>
<td>96</td>
</tr>
<tr>
<td>4.3.1 U.S. Equities Dataset</td>
<td>96</td>
</tr>
<tr>
<td>4.3.2 Feature Extraction</td>
<td>97</td>
</tr>
<tr>
<td>4.4 Dynamic Portfolio Strategies with MS-GAMLSS</td>
<td>98</td>
</tr>
<tr>
<td>4.5 Results</td>
<td>102</td>
</tr>
<tr>
<td>4.6 Conclusions</td>
<td>109</td>
</tr>
<tr>
<td>5 Dynamic Asset Allocation with the Markov-switching Generalised Additive Model for Location, Scale, Shape and Dependence</td>
<td>111</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>112</td>
</tr>
<tr>
<td>5.2 Background</td>
<td>115</td>
</tr>
<tr>
<td>5.2.1 Overview of Copula Theory</td>
<td>115</td>
</tr>
<tr>
<td>5.2.2 Bivariate Copula</td>
<td>118</td>
</tr>
<tr>
<td>5.2.3 Conditional Copula</td>
<td>120</td>
</tr>
<tr>
<td>5.2.4 Pair Copula Constructions</td>
<td>122</td>
</tr>
<tr>
<td>5.2.5 The Simplifying Assumption</td>
<td>126</td>
</tr>
<tr>
<td>5.3 Model Specification</td>
<td>127</td>
</tr>
<tr>
<td>5.3.1 The Marginal Models</td>
<td>129</td>
</tr>
<tr>
<td>5.3.2 The Dependence Model</td>
<td>132</td>
</tr>
<tr>
<td>5.3.3 The Dynamics Model</td>
<td>134</td>
</tr>
<tr>
<td>5.3.4 The Joint Predictive Distribution</td>
<td>135</td>
</tr>
<tr>
<td>5.3.5 Model Estimation</td>
<td>138</td>
</tr>
<tr>
<td>5.3.6 Model Selection</td>
<td>143</td>
</tr>
<tr>
<td>5.4 Dynamic Portfolio Strategies</td>
<td>147</td>
</tr>
<tr>
<td>5.4.1 Probabilistic Forecasts</td>
<td>149</td>
</tr>
<tr>
<td>5.4.2 Higher Moment Dynamic Portfolio Strategy</td>
<td>151</td>
</tr>
<tr>
<td>5.4.3 Data</td>
<td>154</td>
</tr>
<tr>
<td>5.4.4 Backtesting Methodology</td>
<td>155</td>
</tr>
<tr>
<td>5.5 Results</td>
<td>157</td>
</tr>
</tbody>
</table>
## Contents

### iii Optimal Execution

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.6 Conclusion</td>
<td>166</td>
</tr>
</tbody>
</table>

### iv Conclusions

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 Discussion</td>
<td>208</td>
</tr>
<tr>
<td>7.1 Modelling Methodology and design</td>
<td>209</td>
</tr>
<tr>
<td>7.2 Systematic Trading</td>
<td>211</td>
</tr>
<tr>
<td>7.3 Concluding Remarks</td>
<td>215</td>
</tr>
</tbody>
</table>

### v Appendix

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Cryptocurrency Pairs Study</td>
<td>218</td>
</tr>
<tr>
<td>a.1 Cryptocurrency Pairs Study</td>
<td>218</td>
</tr>
<tr>
<td>a.1.1 Joint Probabilistic Forecasting of Cryptocurrency Time-series</td>
<td>222</td>
</tr>
<tr>
<td>a.1.2 Empirical Examination of Volatility and Correlation</td>
<td>223</td>
</tr>
<tr>
<td>a.1.3 Forecast Evaluation</td>
<td>224</td>
</tr>
<tr>
<td>B Appendix: Feature Space Generation</td>
<td>228</td>
</tr>
<tr>
<td>b.1 Features</td>
<td>229</td>
</tr>
<tr>
<td>C U.S. Equities Details</td>
<td>235</td>
</tr>
<tr>
<td>c.1 Chapter 4 Companies</td>
<td>235</td>
</tr>
<tr>
<td>c.2 Chapter 5 and 6 Companies</td>
<td>238</td>
</tr>
</tbody>
</table>

Bibliography 241
List of Figures

Figure 3.1 Linear model (top left), generalised linear model with poisson distributed response (top right), generalised additive model (bottom left), generalised additive model for location, scale and shape (bottom right). ......................................................... 33
Figure 3.2 A comparison of a two-state MS-GAM (left) fitted with EM and the RS algorithm, with a MS-GAMLSS (right) fitted with EM and the gamBoostLSS algorithm. The plots show the fitted values for each of the states, plotted alongside the estimated standard deviation. ....................... 75
Figure 4.1 Model selection (boosting rounds) search grid for the first (left) and second (right) stages. .......... 94
Figure 4.2 Model selection results for a model built for a side study of cryptocurrency pairs (described below and in Appendix A.1 of the Appendix A.), using an 18-fold cross-validation. ............... 95
Figure 4.3 Out-of-sample cumulative portfolio returns for the four 40-asset portfolio strategies. ............. 106
Figure 4.4 Out-of-sample portfolio drawdowns for the four 40-asset portfolio strategies. ...................... 107
Figure 4.5 Portfolio weights for the LS and DF strategies for a selection of equity pairs. Based on average daily adjustments to the LS portfolio, the figures are ranked from least dynamic (top) to most dynamic (bottom). The dashed line and solid line of each colour identify the weights assigned to each asset in the pair. ........................................................................ 108
Figure 5.1 Density contours of six of the most common bivariate copula. ................................................. 119
Figure 5.2 Rotations of asymmetric Archimedean copula for both positive and negative dependence. ....... 121
Figure 5.3 An example of a C-vine construction. ......................................................................................... 124
Figure 5.4 An example of a D-vine construction ......................................................................................... 125
Figure 5.5 The skewed Student’s t distribution .......................................................................................... 130
Figure 5.6 Graphical representation of a Markov switching model with exogenous covariates. ............... 135
Figure 5.7 Experiment 1: (35-asset portfolio). Shows the gross wealth curves of the strategies when transaction costs are not considered. ................................................................. 157
Figure 5.8 Experiment 1: (35-asset portfolio). Shows the net wealth curves of the strategies when transaction costs are considered. ................................................................. 159
Figure 5.9 Fan chart of the wealth curves for 100 randomly selected five-asset portfolios without transaction costs. ......................................................................................................... 162
Figure 5.10 Fan chart of the wealth curves for 100 randomly selected five-asset portfolios with transaction costs deducted. ...................................................................................... 163
Figure 5.11 Experiment 2: Average wealth curves for gross (top) and net (bottom). ................................. 164
Figure 6.1 The initial, optimal and target portfolios for varying costs. Each of the rows identified by A1, A2 and A3 has a different cost to trade the two assets. Top row A1 shows the cost of $TSLA = 125bps$ and $DAL = 10bps$; in row A2 $TSLA = 20bps$ and $DAL = 30bps$; and in row A3 $TSLA = 10bps$ and $DAL = 50bps$. The left column displays the surface of the expected utility function which includes transaction costs and can be compared to the right column which shows the same portfolios on the surface of the utility function without transaction costs. ................................................................. 186
Figure 6.2 The unconstrained initial, optimal and target portfolios for varying initial allocations. The left column shows the surface with the inclusion of transaction costs and the right column is without costs. The cost for each asset for all examples is 25bps. ................................................................. 188
Figure 6.3 The unconstrained optimal solutions for varying scaling parameter $\lambda$; the red dot indicates $\lambda=1$. Costs in basis points are $TSLA = 80bps$ and $DAL = 25bps$, Expected returns for this problem were $TSLA = 143.2bps$ and $DAL = 29.4bps$. ................................................................. 189
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.4</td>
<td>Chart of the wealth curves for the 35 asset portfolio strategies using the MS-GAMLSSD model, comparing the gross and net results for the cost minimisation strategy (blue) against the original strategy from Chapter 5 (black).</td>
</tr>
<tr>
<td>6.5</td>
<td>Chart of the wealth curves for the 35 asset portfolio strategies using the SS-GAMLSSD model, comparing the gross and net results for the cost minimisation strategy (blue) against the original strategy from Chapter 5 (black).</td>
</tr>
<tr>
<td>6.6</td>
<td>The top panel shows box plots for the bid-ask spreads (in basis points) for the five stocks in portfolio #5 of Experiment 1. The second panel shows the corresponding box plots of the contribution to daily turnover by each stock when transaction costs are not part of the optimisation problem. The third panel shows the turnover when transaction costs are considered in the optimisation. The bottom panel shows the median reduction in daily turnover resulting from the cost minimisation method alongside the total reduction in turnover for the duration of the strategy.</td>
</tr>
<tr>
<td>6.7</td>
<td>Wealth curves for C5 (black) and C6 (blue) strategies</td>
</tr>
<tr>
<td>6.8</td>
<td>Results from Chapter 5: fan charts of the wealth curves for 100 randomly selected five-asset portfolios.</td>
</tr>
<tr>
<td>6.9</td>
<td>Results from the cost minimisation strategy: fan charts of the wealth curves for 100 randomly selected five-asset-asset portfolios.</td>
</tr>
<tr>
<td>A.1</td>
<td>Hourly price time-series of Bitcoin and Ethereum to U.S. Dollar currency pairs. The shaded area to the right of the figures indicates the test region and the rest is the training data.</td>
</tr>
<tr>
<td>A.2</td>
<td>Histogram plot for BTC/USD and ETH/USD hourly price returns data. The black line indicates a density fit to data observed in the period before the high points of each time series (17/12/2017 for BTC and 14/01/2018 for ETH); the blue line is a fit for the period after these dates and green is a normal curve fitted to the entire series.</td>
</tr>
<tr>
<td>A.3</td>
<td>Figure A.3 2D density plots before and after the cut-off of 07/12/2017.</td>
</tr>
<tr>
<td>A.4</td>
<td>The in- and out-of-sample squared volatility and correlation forecasts for the BTC/USD and ETH/USD time series. The series are smoothed with a five-day simple moving average for readability. The MS-GAMLSS forecasts are compared with the DCC, with the asymmetric extension to the DCC (ADCC), and with sample variance and correlation estimates (calculated using a simple moving average).</td>
</tr>
</tbody>
</table>
List of Tables

Table 4.1  Backtesting results for the 20 equity pairs-trading strategies using the bivariate gaussian MS-GAMLSS model. Results include the two mean-variance strategies (long-short portfolio (LS), long-only (LO)), presented alongside the equal weighted (EW) benchmark portfolio. The strategy returns are presented with and without the inclusion of transaction costs. The final column shows the average daily bid-ask spread in terms of percentage of asset prices. The realised Sharpe ratio presented here is annualised.

Table 4.2  Backtesting results for the 20 equity pairs-trading strategies using the bivariate gaussian MS-GAMLSS model for directional forecasting. Results include the 20 individual directional forecast strategies (DF) with S1 and S2 the contributions of each of the two assets in the portfolio. The additional classification results show the bivariate MS-GAMLSS’s performance forecasting up and down price movements.

Table 4.3  The main strategy results for portfolios containing the 20 equity pairs, each modelled by a bivariate MS-GAMLSS. (LS = long short, LO = long only, DF = directional forecast and EW = equally weighted benchmark portfolio).

Table 4.4  Total (all pairs) classification evaluation metrics for the bivariate MS-GAMLSS directional forecasting strategy.

Table 4.5  Daily average percentage change in portfolio weights (turnover).

Table 5.1  Experiment 1 (thirty five asset portfolios): Strategy performance metrics. Panel A reports the results for the strategies without transaction costs; Panel B the results when transaction costs have been deducted; and Panel C the results for the two benchmark portfolios. The letters MVSK stand for mean, variance, skewness and kurtosis, and denote which of these quantities are forecasted by the particular model.

Table 5.2  Average strategy performance metrics from 100 randomly selected five-stock portfolios. Panel A reports the results for the strategies without transaction costs, Panel B shows the results when transaction costs have been deducted, and Panel C shows results for the two benchmark portfolios. The letters M, V, S, and K, where used, stand for mean, variance, skewness, and kurtosis, and denote which moments are considered in the portfolio optimisation by a particular strategy.

Table 5.3  Experiment 2: Average strategy performance metrics from 100 randomly selected five-stock portfolios.

Table 6.1  Descriptive statistics of the two assets.

Table 6.2  The turnover and costs relating to the portfolio rebalancing problem in Figure 6.1.

Table 6.3  The turnover and costs relating to the portfolio rebalancing problem in Figure 6.2.

Table 6.4  Comparative results for the cost minimisation strategy with the original results from Chapter 5 Experiment 1.

Table 6.5  Comparison of strategy test performance for portfolio #5 using the original Chapter 5 (C5) and cost-minimising (C6) strategies. The top panel shows results without transaction costs deducted (gross). The lower panel shows the true results when transaction costs are included (net).
Table 6.6 Experiment 2: Comparison of the new optimisation method (C6) with the standard method (C5) of Chapter 5, showing the average strategy performance metrics from 100 randomly selected five-stock portfolios. Panel A reports the results for the MS-GAMLSSD based strategies both with and without transaction costs, and similarly, Panel B reports the results for the single-state SS-GAMLSSD based strategies. The blue entries represent the standard errors. .......................... 199

Table 6.7 Experiment 2: The complementary benchmarks for comparison with the results from Panel A and B in Table 6.6. These are the equal-weighted five-asset portfolios (EW) and the Standard and Poors 500 index (S&P 500). The blue entries represent the standard errors. .......................... 200

Table A.1 Summary statistics .................................................... 221

Table A.2 Mincer–Zarnowitz regression summary (in-sample); p-values for the $a = 0, b = 1$ hypothesis are included only in the cases where it cannot be rejected with $> 90\%$ confidence. .......................... 226

Table A.3 Mincer–Zarnowitz regression summary (out-of-sample); p-values for the $a = 0, b = 1$ hypothesis are included only in the cases where it cannot be rejected with $> 90\%$ confidence. .......................... 226
Part I

PRELIMINARIES
Introduction

Since the advent of autonomous trading systems, their presence has become ubiquitous throughout our financial system. Their application spans a broad spectrum of market participants, including investment banks, market makers, hedge funds, and extending to individual retail investors. The widespread adoption of trade automation systems is largely attributed to the substantial advantage they can have over traditional discretionary investment management methods. A prime example of such automation in asset management are systematic trading strategies. These strategies, grounded in algorithmic procedures, leverage quantitative methods to detect patterns in the time-series of financial asset prices. They base trading decisions on anticipated future price behaviours, aiming to manage funds securely while generating profits from trading activities.

This thesis focuses on a particular type of systematic trading strategy known as dynamic asset allocation. It is an automated process that strategically manages a portfolio of assets over time, adapting the portfolio composition in response to emerging new information. For success, the strategy necessitates continuous monitoring and adjustment to an optimal asset allocation, frequently altering the holdings in each asset. Theoretically, this method is designed to enhance an investor’s wealth and manage investment risks simultaneously. However, practically implementing this strategy effectively presents significant challenges.

At the heart of any asset allocation strategy lies a forecasting model, tasked with generating expectations of future price behaviour, with the quality of the forecasts being of paramount importance to the strategy’s success or failure. Forecasting the price movements of financial
assets, however, is extremely difficult to do with any degree of accuracy due to the nature of
the uncertainty in financial markets (described in the section below). Moreover, a strategy that,
by design, involves frequent trading, is inherently costly, and these costs can make systematic
trading prohibitively expensive, to the point where operational expenses can easily outweigh
potential profits, rendering the strategies unviable. This thesis addresses these two critical
factors that can significantly impact the performance and viability of using systematic strategies
to autonomously manage portfolios of financial assets.

1.1 STATEMENT OF THE PROBLEM

Before introducing the aims of this research, a brief, semi-formal, description of the basic
requirements for an optimal asset allocation will be given, as this will be helpful in illustrating the
problem. Fundamentally, for a portfolio containing \( d \) assets, we want to find a set of weights
\( w_{t|t+1} \) (the allocation) that proportionally assigns an investor’s wealth \( W_t \) to each of the assets
in the portfolio; typically, the allocation is decided by the solution to an optimisation, with the
best-known of these optimisation problems being based on expected utility theory. In this case,
we start with a utility function \( U(W_{t+1}) \), that has a form that adequately reflects the risk and
reward preferences of the investor, is a function of end-of-period wealth, and outputs a real
value measure of utility. As \( W_{t+1} \) is an unknown random variable from a future point in time, we
cannot just maximise the utility function; instead, the problem requires finding the
\( w_{t|t+1} \) that
maximise the expected value \( E[U(W_{t+1})] \). This expectation can also be written in terms of the
portfolio return, giving us the asset allocation problem

\[
w^*_{t|t+1} := \arg \max_{w_{t|t+1}} E_{P_{t|t+1}} [U(W_t (1 + R_{t+1}))],
\]

where \( R_{t+1} \) is a random variable representing the one-period return of the portfolio and \( P_{t|t+1} \)
is the probability distribution of the portfolio return. The problem we have with taking this
expectation is that, importantly, \( P_{t|t+1} \) is a quantity that contains information about the future
and, therefore, is unknown, and hence we need a forecast of the portfolio return distribution to
compute this expectation.

However, to do this we do not need to forecast \( P_{t|t+1} \) directly. As the value of the univariate
random variable \( R_{t+1} \) represents the total return on investment for the portfolio its value is a
linear combination of the individual returns of the portfolio’s assets, weighted by the particular
allocation. Therefore, if \( Y_{t+1} \) is the multidimensional random vector representing the price
returns of the portfolio’s assets, and $\mathcal{Y}_{t|t+1}$ is its multivariate distribution, then, for a given asset allocation $w_{t|t+1}$, $\mathcal{P}_{t|t+1}$ is known if we know the joint distribution of the asset returns $\mathcal{Y}_{t|t+1}$, though, as in the case of $\mathcal{P}_{t|t+1}$, $\mathcal{Y}_{t|t+1}$ is also not known and needs to be forecasted. In brief, to find an efficient asset allocation from an expected utility maximisation, a multivariate probabilistic forecast of the asset returns distribution is needed. This can be problematic for financial time series, as $\mathcal{Y}_{t|t+1}$ is continuously changing over time.

The asset allocation problem described above outlines what is required for choosing a portfolio over a single investment period, and therefore is, in effect, a static asset allocation problem. The type of autonomous trading system proposed within this thesis is one that manages a portfolio strategy over a period of time by systematically updating the allocation of the portfolio’s assets $w_{t|t+1}$ in response to step-wise updates to the probabilistic forecasts generated by the model. Such strategies are dynamic in nature, and so require frequent rebalancing from one allocation to another as the forecasts of $\mathcal{Y}_{t|t+1}$ change over time.

The amount of buying and selling of assets as a consequence of rebalancing can be measured by the quantity known as *portfolio turnover*, which throughout the course of a strategy of the nature described here can become very high. Portfolio turnover exposes the strategy to transaction costs, meaning that, even though frequent rebalancing can ensure that the optimal allocation can be tracked over time, to do so may not be feasible, as it could possibly cost more in transaction fees than would be generated in revenue; at the very best, transaction costs can significantly impair the performance of a portfolio strategy of this nature.

### 1.2 RESEARCH AIMS

The overall aim that encompasses all areas of research in this thesis was to produce a high quality autonomous trading system that manages a systematic strategy for dynamic asset allocation of portfolios. Management in this respect involves generating sufficiently high profits while effectively managing risk and the operational cost of trading. The goal of the research was to ensure the trading strategies perform sufficiently well that they could be used with confidence in a live trading environment, and, therefore, to ensure that the systematic strategies can offer practical value in real world applications. To achieve this aim required two separate topics of research, described below.
The primary contribution of this thesis is to introduce a novel modelling framework specifically designed for forecasting the price behaviour of an entire portfolio of financial assets, thereby introducing a new class of probabilistic forecasting model for multidimensional time-series. The model framework was tailored to generate one-step-ahead rolling forecasts of the joint probability distribution of a portfolio’s asset price returns, to be used as input signals into a dynamic asset allocation strategy. An aim of the model framework was for it to handle potentially very high dimensional data sets by leveraging methods from machine learning, in this way standing apart from orthodox econometric models. The challenges of achieving this aim, especially for multivariate models, will be discussed in detail in the following chapter; however, in brief, the general consensus has always been that there are high amounts of uncertainty in the behaviour of financial asset prices, and, compounding the difficulty, that there is also uncertainty about this uncertainty. This is evident when uncertainty in financial price movements is quantified in a probability distribution; all the features of the distribution have been shown to change over time. This includes time-varying means, variances, skewnesses, kurtoses and dependences (e.g. correlations). The modelling aims in this research were to develop a framework capable of capturing this ever-changing uncertainty and to therefore provide high quality forecasts for dynamic management of a portfolio over time.

The second major contribution of this thesis addresses the impact that transaction costs have on the performance of asset allocation strategies. As described in the previous section, the optimisation problem for an asset allocation requires an expectation of future price behaviours in order to decide on an efficient allocation of wealth across the assets in the portfolio; however, traditional optimisation methods do not take into consideration the cost involved in adjusting positions to meet the new allocation. To rebalance from one optimal portfolio to the next may not be the best choice when costs are taken into consideration; hence, to improve the performance of dynamic asset allocation strategies, the aim was to develop a novel portfolio optimisation objective function that considers the potential costs of each portfolio rebalancing, and actively seeks to minimise these costs.
1.3 RESEARCH OBJECTIVES

To facilitate the aims of this thesis the following research objectives were identified.

1.3.1 Probabilistic Forecasting

A main objective for the design of the modelling framework was for it to have the capacity to capture all known statistical properties of financial asset returns; these properties are discussed in more detail in the following chapter, but, importantly, include time-variation in covariance and higher order moments, asymmetries in shape and dependence, and structural breaks and regime changes. To ensure the new model’s capacity to capture distributional time-variation, two different mechanisms that allow for time-variation were specified in the model’s design objectives: a local source of time-variation, where all parameters of the multivariate distribution are conditioned on an evolving information set of exogenous variables; and a global source of time-variation which accounts for changes in market conditions.

The main objective for model design was to appropriately extend the distributional regression framework known as generalised additive models for location scale and shape (GAMLSS). GAMLSS is a highly-flexible modelling framework where all parameters of a response variable’s distribution are modelled with a generalised additive regression model. The original GAMLSS framework was previously extended in Mayr et al. (2012) by using a model-based gradient boosting method for estimating model parameters, permitting high-dimensional sets of exogenous variables. However, the standard form of the GAMLSS framework is for univariate distributional regression of cross-sectional data; hence, one objective here was to extend the standard framework to one for modelling multivariate time-series.

To achieve the aims of this thesis, the gradient boosting GAMLSS framework was in fact extended in two main ways:

1. An extension of the univariate GAMLSS modelling framework to one for modelling multivariate distributions, as mentioned above.

2. An extension of the multivariate GAMLSS framework for its use in time-series modelling, with the inclusion of regime-switching dynamics.
These two main objectives are addressed in the first two research chapters. In the first of these chapters, the objective is to extend the univariate GAMLSS framework to the bivariate case, to be applied to the modelling of the price returns time-series of pairs of financial assets. The aim for this initial stage of the research was to keep the specification of the bivariate distribution relatively simple, to identify any issues that might be encountered with the multivariate extension. The objective was therefore to introduce a regime switching bivariate gaussian GAMLSS model, and test its efficacy and feasibility for inclusion as a probabilistic forecasting model in dynamic pairs trading strategies.

The modelling objective of the second of these research chapters was to further extend the model from the previous chapter into a full multivariate modelling framework which, in theory, could jointly model an arbitrary number of timeseries, by introducing the Markov-switching generalised additive model for location, shape, scale and dependence (MS-GAMLSSD). Using this multivariate framework the joint behaviour of entire portfolios of assets could be modelled.

Key features of the model design were as follows:

1. All parameters of the multivariate distribution are conditional on a potentially high dimensional set of exogenous variables.
2. Includes automatic variable selection.
3. Has regime-switching dynamics.
4. Accommodates skewed and leptokurtic distributions.
5. Has a flexible dependence structure that accommodates asymmetric and nonlinear dependence.
6. Scales reasonably well with the size of portfolios.

In addition to the model specification, a further objective of the extension from the earlier work on bivariate distributions to the full multivariate model was to generalise the probability distribution from the well-known bivariate gaussian to a flexible copula representation containing skewed Student’s t marginals and a flexible pair copula construction for the dependence structure. This way of specifying the MS-GAMLSSD makes it as flexible and general as possible, and gives it the potential to be able to capture the statistical properties of multidimensional time-series of asset returns.
1.3.2 Operational Cost Minimisation

The objective of this part of the research was to overcome the impairment to systematic trading strategy performance due to the excessive portfolio turnover permitted by traditional portfolio optimisation methods. This was done by introducing an autonomous means to manage the operational cost without compromising optimality of the asset allocation. Specifically, a cost-minimising adaptation to an expected utility portfolio optimisation problem was introduced into the dynamic asset allocation strategy.

1.4 Scope

The scope of this thesis is limited to trading strategies using portfolios containing U.S. equities listed on the NYSE, ANEX and NASDAQ. Backtesting in all the experiments is performed using the same market data and involved daily time-frames. It would be of interest to investigate using intraday trading strategies; however, this was not considered necessary in order to prove the value of the proposed methods. Additionally, the chosen portfolio optimisation methods were straightforward textbook methods, sufficient again to prove the value of the methodology.

For the novel modelling framework the scope for its validation is limited to proving its efficacy and feasibility as a viable probabilistic forecasting model for the task of signal generation for dynamic asset allocation strategies, the task for which it was designed. Other tests were deemed to be out of scope.

The specifications for the models were necessarily restricted in number, though very many alternatives could in principle have been explored. The models that were picked are able to be modified in many ways, and were chosen to be as general as possible. For example, the regime-switching mechanism for the models was limited to a first-order Markov-switching model that switches between multivariate distributions. This is the simplest way of implementing a latent state-space model. Other more complex specifications are possible, and some were developed, including a novel gradient boosting GAMLSS model of the conditional transition density; however, this was not central to the work of this thesis and its further development was therefore left for future work.
1.5 SIGNIFICANCE

Multivariate distributional regression models, until recently, have been rare in the literature; the complexity of a multivariate distribution compared with a univariate one is substantially greater. A significant result of this thesis was that even with a multidimensional model for a parametric distribution there are ways to address model estimation and hyperparameter selection issues that cause intractability. The modelling contributions are:

• Using the gradient boosting GAMLSS to model a conditional bivariate gaussian.

• Using the gradient boosting GAMLSS to model a conditional pair copula construction.

• For both of the above, using these multivariate models as state-dependent components of a regime-switching model.

The operational cost reduction method introduced in this thesis has significance for retail investing. The past few years have seen the emergence of retail trading platforms that operate on a commission-free model. These new brokerages have democratised financial market investing and drawn in a whole new generation of market participants. On these platforms, the only trading cost to an investor is in the spread between the price at which a security can be bought and sold. This is information known at the time of trading and proportional to the size of the trade; therefore, portfolio strategies run on these platforms can directly benefit from the cost minimisation method introduced in this thesis, independently of the method used for obtaining forecasts.

The contributions to autonomous trading are as follows:

• Application of the models to dynamic asset allocation strategies and validation using backtesting without any limiting assumptions.

• Introduction of a operational cost reduction method to improve performance of systematic strategies.
Chapter 1 has introduced the research aims and scope of the thesis.

Chapter 2, which follows, contains a more detailed description of the motivation for the work of this thesis, together with relevant conceptual background. It also contains a review of relevant academic literature.

Chapter 3 presents a detailed pedagogical presentation of the key technical concepts behind the main aspects of the new modelling framework. This includes a detailed description of univariate distributional regression, including generalised additive models for location, scale and shape (GAMLSS), followed by gradient boosting, the machine learning method used for building these models, including the standard version of the gamBoostLSS algorithm. In the final part of the chapter, regime-switching models are discussed, including technical information about estimation and related inference methods.

Chapter 4 is the first of the three research chapters. In this chapter, gradient boosting for GAMLSS models of univariate distributions is extended from a univariate framework to the bivariate case, introducing a bivariate gaussian regime switching model. The aim of extending to the bivariate case before continuing to a multivariate framework was to identify any challenges that would be encountered in further extension.

Chapter 5 is the second of the three research chapters. This chapter extends the bivariate model of the previous chapter to the full multivariate case, introducing the Markov-switching generalised additive model for location, scale, shape and dependence (MS-GAMLSSD).

Chapter 6 is the final research chapter. In this chapter, the problem of operational cost in dynamic asset allocation strategies is addressed. A novel adaptation to a higher moment objective function is introduced that takes into account the potential transaction cost of portfolio rebalancing.

Chapter 7 concludes the thesis with a discussion of the findings of the three previous experimental chapters and identifies opportunities for future work.
Motivation, Background and Related Work

This chapter provides an introduction to the subject domain of this thesis, motivating the topics addressed in this work with relevant, non-technical, background, and a review of the relevant literature. Starting with a brief overview of systematic trading and probabilistic forecasting, the chapter proceeds with a discussion of modern portfolio methods, followed by a review of modelling methods for financial time-series, and ends with a review of current modelling methods that motivated the development of the models proposed in this thesis.

2.1 OVERVIEW OF SYSTEMATIC TRADING

Systematic trading, sometimes referred to as quantitative trading, is an approach to the buying and selling of financial instruments strategically, in a controlled and systematic manner. The trading strategies are designed to run according to a fixed system or plan, where trading rules are set in advance after having been derived through a rigorous process of research, design and testing. Today’s systematic trading strategies tend to be algorithmic procedures for fully or partially automating the trading process. They are developed through applications of quantitative methods from the fields of mathematics, statistics, physics, artificial intelligence and machine learning. The primary goal is very simple: to create a strategy that aims to increase wealth by entering positions which generate a positive return on an initial investment. Secondary goals that need to be considered include achieving the main goal of increasing wealth while at the same time controlling the investor’s exposure to risk, and undertaking all of this while
considering the costs behind transacting the trades. Despite the simplicity of the above goals, attaining them is non-trivial. Financial prices have in the past been argued to move in ways that are fundamentally unpredictable (see, for example, Cootner (1964), Malkiel (1999), Fama (1965) and Fama (1995)). This perception can in part be explained by the source of the randomness governing price behaviour having a tendency to change over time, which makes it extremely difficult to understand how financial asset prices might behave in the future.

Arguments for the necessity of systematic strategies are primarily based around the advantages that can be gained by removing the human decision-making process from the task of trading financial securities. Trading involves making judgements while faced with uncertain outcomes. An ability to assess the risk associated with a decision is an essential factor for successful performance. However, making optimal decisions while faced with uncertain outcomes is not a task that the human brain has evolved to do well. We are reliant on heuristics to make judgements in complex situations; these heuristics are error-prone and induce biases into the decision-making process. Social psychologists Tversky and Kahneman (Tversky and Kahneman (1974)) were awarded the Nobel prize for their work showing that we, as humans, are not capable of correctly assessing the risks we undertake; as such, our judgement can be affected, and decisions fall short of what is optimal. The field of behavioural economics has evolved based on these advancements in social psychology (see Thaler (2016)). Human decision-making is not only guided by biased beliefs but driven by emotions, gut feelings and, of course, stress, all of which are detrimental to making good choices. The management of financial assets is thus a complex, uncertain endeavour, unsuited to human abilities; as a result of advancements in technology, we have therefore seen a significant rise in the use of algorithmic procedures to make investment decisions that can assist people or operate autonomously in their place.

For systematic strategies to perform well, it is rarely enough to simply forecast the direction that the price of an asset will move. Holding positions in financial assets comes at significant risk to one’s investment. Successful strategies need to be able to manage this exposure to risk; to achieve this, they will usually trade multiple assets simultaneously to control the level of risk that is undertaken. The two main mechanisms available to achieve this are diversification and hedging. In order to use these forms of risk management, practitioners require adequate forecasts of risk for the period the assets will be held, along with forecasts of dependencies between all tradeable assets. The latter are forecasted measures of the strength of the statistical relationships between the asset prices, and contain information about how prices are expected to co-vary over the period the position is held. The goal is then to allocate a position to be held in each asset that is expected to be profitable, while managing the exposure to risk, and then adjusting these positions accordingly as new information is revealed.
Dynamic portfolio strategies, briefly introduced in Chapter 1, are those that, from time to time, either at fixed intervals or as a result of some detected event, systematically adjust positions in the assets held in the portfolio. These types of strategies have the potential to show substantial gains; however, adjusting a position incurs a cost for the transaction, and when this is done regularly the costs can easily exceed gains, making the strategy unfeasible. A complete autonomous trading system will usually contain a mechanism to deal with the impacts of these costs. The way in which systematic trading strategies operate can differ widely; however, the three main constituent layers of a typical strategy, as used in the experiments of this thesis, are:

- **The alpha component**, which infers possible opportunities. This layer will usually be in the form of a type of forecasting model that generates buy or sell signals, or expectations of directional movement, return, variance and dependence, etc.

- **Risk management / portfolio optimisation**, which takes the output of the alpha component as its input and allocates a proportion of funds to the identified positions.

- **The execution strategy**, which is primarily there to minimise the cost of transacting the position; however, it may also be used to override the desired trade.

The alpha component of a systematic strategy is normally considered to be the most important. Its main responsibilities are to provide expectations of future price behaviour or to discover a condition that warrants a position in an asset to be changed. Achieving this will usually require a model of the underlying price time series for the set of tradable securities. However, financial returns modelling is a far from trivial undertaking. Properties of these time-series make forecasting even short distances into the future extremely difficult, with forecasts containing high degrees of error. As noted previously, this has in the past motivated some economists to hypothesise that financial markets follow random walks, and are therefore unpredictable. It is, however, the belief of today’s research community and industry practitioners that a limited predictability exists, and that there are sufficient signals in financial data to make systematic strategies feasible. With so much uncertainty around a prediction, however, it makes sense to have some knowledge of this uncertainty; probabilistic forecasts are a way in which we can get a good understanding of all possible outcomes of entering into a position in a financial asset, and, with this knowledge, implement effective risk management measures to protect against possible adverse outcomes.
2.2 PROBABILISTIC FORECASTING

It is part of human nature to want to understand what may happen in the future. As an everyday part of life, we have to make consequential decisions based on our current understanding of what we believe may or may not occur. If I am told there is an 80% chance that it will rain today, I will choose to take an umbrella with me when leaving the house. If there is a 60% chance the Bank of England will hike the base rate in six months from now, I may wish to use this information to lock in a mortgage before they potentially become more expensive. These are two examples of decisions that we make as part of day-to-day life that are based on information we receive through a probabilistic forecast. In these simple cases, the forecasts have been framed over binary outcomes \{rain, no rain\} and \{hike, don’t hike\}, and the probability distribution that explains the random event is a single parameter Bernoulli trial. In this special case, forecasting the mean of the distribution is sufficient to obtain all information about the uncertain event; however, these examples provide us with very little information on which to base a decision. For instance, rather than basing a decision on knowing only whether or not it may rain tomorrow, a farmer may need to have information on the expected level of precipitation in the air before he can decide to harvest a crop, or the Bank of England’s monetary policy committee may need information on how the rate of inflation could change over the next few months to inform their policy decisions. In situations like these, forecasting expected outcomes provides only partial information about the unknown event and what is lacking is any sort of information about the amount of uncertainty that comes with the prediction.

Traditionally when we consider the term ‘forecast’, it is natural to think of a simple point estimate of a single quantity; this, most commonly, will tend to be the mean of the distribution underlying the uncertain event, e.g. ‘Apple’s stock price is expected to increase by 10% in the next year’. Forecasts of this type are essentially deterministic, and give us just a single value to base decisions upon, with no information about the uncertainty of the prediction; with this type of forecasting, as already pointed out, decision makers have no means of gauging the level of risk of an adverse outcome. With a forecasted probability distribution we get not only a point forecast of the expected value of an uncertain event; we get information about the entire distribution, which gives us an understanding of all the possible outcomes. For instance, Apple’s stock price may be expected to increase 10% in the next year, which sounds like an attractive investment opportunity; however, what if we also know that there is a 20% chance that it will fall more than 40%? This completely changes the attractiveness of the investment.
Over recent years, it has become more and more commonplace for forecasters to use probability distributions to describe their forecasts. This is becoming even more present in fields of research for which uncertainty is pervasive and the consequences of being wrong have the propensity to cause harm. Take earthquake forecasting as an example, a low probability, yet high risk, setting where the consequences of incorrect forecasts can be catastrophic; Jordan et al. (2011) state that probabilistic forecasts are the best means for transmitting scientific information about future earthquake occurrence. Likewise for the assessment of flood risk: Krzysztofowicz (2001) state that probabilistic forecasts are scientifically more honest, enable risk-based warnings of floods, allow for rational decision making, and provide economic benefits (see also Cloke and Pappenberger (2009) for a review of probabilistic forecasts in flood forecasting systems).

For a discussion of the importance of probabilistic forecasting in meteorology and climate science, two fields in which these types of forecasts are commonplace, see Anderson (1996), Collins (2007), Pinson (2012) and Palmer (2012). Furthermore, in epidemiology, short-term projections of HIV prevalence have been studied using probabilistic forecasting models in Alkema, Raftery and Clark (2007); likewise, forecasting of the spread of dengue fever has been studied in Johansson et al. (2019) and, in more recent work, Taylor and Taylor (2021) and Cramer et al. (2021) looked at the probabilistic forecasting of COVID-19 mortality rates in the United States. Daily case rate forecasts of COVID-19 prevalence were the subject of Rostami-Tabar and Rendon-Sanchez (2021) (see also Chen (2022), Gibson, Reich and Sheldon (2020), Papastefanopoulos, Linardatos and Kotsiantis (2020) and Athanasopoulos et al. (2021) for other research on the COVID-19 pandemic that relied on probabilistic forecasting methods).

The financial markets – more specifically, financial asset prices – are notoriously difficult to predict, giving rise, as mentioned above, to past speculation in some quarters as to whether they could be predicted at all. Forecasts of future returns are extremely uncertain, and even with the most sophisticated state-of-the-art models predicted values will tend to correlate very weakly with the ground truth when compared with other problem areas in time series prediction. For example, a recent Kaggle competition challenged over 2000 teams to predict price returns of a set of cryptocurrencies. At the end of the test period, the winning entrant’s predictions had a weighted Pearson’s correlation of 0.033 with actual returns (see Kaggle (2021) for more details). With such a high noise, high uncertainty, and very low signal problem domain, it is very important to have a good understanding of the uncertainty behind each prediction; this can be obtained by employing modelling frameworks that achieve forecasts of the entire predictive distributions. For risk management, especially, knowledge of all possible outcomes is essential to guard against investment losses. Having access to a full forecasted predictive distribution permits the computation of risk measures such as value at risk (VaR) and expected shortfall (ES); it also allows us to take expectations of functions of the price return, as with the maximisation of expected utility functions found in asset allocation. In fact, most problems in quantitative...
finance involve the expectation operator over some function of future price, and to take this expectation we require knowledge of the full predictive distribution.

Probabilistic forecasting as we know it today is, however, not a new concept in finance. For example, portfolio management, since its very inception in Markowitz (1952), has required forecasts of the mean and variance of the price returns distribution, as, at the time, price returns were assumed to be normally distributed, these quantities were deemed sufficient to define the entire predictive joint probability distribution of asset returns. Additionally, in financial econometrics, early auto-regressive models for financial time-series prediction, introduced in Engle (1982) and Bollerslev (1986), were designed for forecasting time-varying means and variances of price returns and were, in essence, early time-varying probabilistic forecasting models. But 40 years have passed since Engle’s ARCH model was introduced, and 70 years since Markowitz published his seminal paper on modern portfolio theory, and over the years a lot has changed in our understanding of how financial asset prices behave, and, moreover, of the risks involved in investing in financial markets that result from this behaviour. Taking these advancements into account, today’s probabilistic models for financial price returns require an additional complexity to capture the statistical properties and empirical regularities we now know as the ‘stylised facts’ of financial price returns (Cont (2001)). In addition, today we have available to us major advancements in statistical tools and machine learning methods, along with capability to leverage the information in large datasets to build better, more advanced, models for financial time-series than have been available in the past.
2.3 PORTFOLIO MANAGEMENT

Portfolio management is a well understood practice making use of quantitative methods for deciding how best to allocate funds across multiple investment sources. In its simplest form, a portfolio consists of a set of risk assets, e.g. equities, and an allocation that determines what proportion of the funds (i.e. total wealth) will be invested in each asset. To find the particular allocation that is expected to offer the greatest benefits to the investor, optimisation methods are used. As was briefly discussed in the previous chapter, the portfolio optimisation problem needs, as an input, certain information that quantifies the uncertainty in asset price changes over the investment period. Therefore, optimal allocation requires a probabilistic forecast of the joint distribution of assets' price returns. With this information, as well as an understanding of an investor's appetite for risk, or lack of it, portfolio methods obtain an optimal allocation by means of a trade-off between the expected reward the portfolio may yield and the amount of risk involved in holding it over the investment period.

In the not too distant past, it was widely considered to be sufficient for the expected behaviour (uncertainty) of asset returns to be represented, probabilistically speaking, by only the first two moments of their joint probability distribution; this was the case for the original contribution to statistical portfolio selection known as modern portfolio theory (MPT), also referred to as mean-variance analysis. MPT is regarded as one of the most important contributions to financial economics; it is the seminal work of Markowitz (1952), and became a Nobel prize winning theory that has provided the blueprint for most research on optimal investment of risky assets. MPT highlights the importance of the effects of risk, return and correlation (dependence) when allocating funds optimally within a portfolio of financial assets. Methods for optimising portfolios rely on a concept known as diversification to control the level of risk a portfolio is exposed to, spreading funds over numerous assets in order to limit exposure to any single asset or risk.

The quality of forecasts of the mean and covariance can be an issue in the implementation of mean-variance portfolio optimisation. Although MPT methods are elegant in their simplicity, there are many challenges to providing good quality forecasts of these quantities for input into the optimisation problem. To illustrate this issue, DeMiguel, Garlappi and Uppal (2007) provide a critical review of 14 mean-variance estimation methods and conclude that, for all tested methods, in out-of-sample testing, the gain from optimal diversification is more than offset by the estimation error of the inputs to the optimisation problem. Moreover, when compared to a naive equally weighted portfolio, the 14 more sophisticated methods performed poorly. This
review is important as it illustrates that MPT, and portfolio optimisation methods in general, are dependent on the quality of estimates that the optimisation problems receive as inputs, i.e. the quality of the forecasts of the means, variances and correlations (as well as other quantities yet to be discussed) of asset returns.

Following on from Chapter 1’s brief introduction to asset allocation, if we are at time \( t \), and need to find an allocation for a portfolio that will be invested in until time \( t + 1 \), then we need to have some notion of the uncertainty of the asset price movements between \( t \) and \( t + 1 \), i.e. a probability distribution for the random vector \( Y_{t+1} \), representing the returns of the portfolio’s assets. Given all the information we have at the current time \( t \), we can denote this distribution as \( Y_{t|t+1} \). For a MPT optimisation problem, the inputs into the optimisation objective are the first two moments of the distribution; therefore, the behaviour of asset returns can be formally defined as a random vector

\[
Y_{t+1} \sim Y_{t|t+1} \left( \mu_{t+1}, \Sigma_{t+1} \right),
\]

where \( \mu_{t+1} \) is the vector of expected returns (means) and \( \Sigma_{t+1} \) its expected covariance. This specification for \( Y_{t|t+1} \) relates to a portfolio return \( R_{t+1} \sim \mathcal{N} \left( \theta_{t+1}, \sigma_{t+1} \right) \) with mean \( \theta_{t+1} = w_{t|t+1}^\top \mu_{t+1} \) and variance \( \sigma^2_{t+1} = w_{t|t+1}^\top \Sigma_{t+1} w_{t|t+1} \); therefore, mean-variance analysis weakly assumes that portfolio returns follow a normal distribution.

For a long time now it has been appreciated that the above representation is a relatively simple, albeit a very mathematically friendly, approximation of actual price behaviour, see Scott and Horvath (1980) and Samuelson (1975). Financial economists have long questioned whether the mean-variance approximation is a good enough approximation to account for all the associated risks in an investment, arguing that many of the risks involved in holding invested positions are overlooked by ignoring the information located in the higher-order moments of the portfolio returns distribution.

It is nowadays well understood that the first four moments of the portfolio returns distribution, with the obvious exception of the mean, all play an important role in quantifying the different types of risk an investor undertakes when holding a portfolio of risk assets. As already seen, the first and second moments represent actual mean and variance, i.e. the standard reward-to-risk trade-off in traditional portfolio methods. The higher-order moments, for the purposes of asset allocation, are those of the third and fourth order, and are directly related to the skewness and kurtosis of the portfolio return distribution, denoted as \( P_{t|t+1} \). These higher portfolio moments, as with the mean \( \theta_{t+1} \) and variance \( \sigma^2_{t+1} \), are dependent on the particular allocation \( w_{t|t+1}^\top \), and also the higher co-moments of the joint distribution of asset returns, \( Y_{t|t+1} \). As previously mentioned, these additional moments contain important information about risks beyond that
which can be explained by the variance; in summary, the additional risks and their relation to the standardised higher moments can be described as

- **Skewness** (3rd): *Asymmetric risk* - is present when price moves in one direction tend to differ in magnitude and / or frequency to price moves in the opposite direction,

- **Kurtosis** (4th): *Tail risk* - is present when there is an increased risk of extreme price movements; typically, financial market prices experience rare events (price shocks) much more frequently than can be explained by the normal distribution.

A rational investor, for example, if asked to choose between two portfolios with the same expected mean and variance, would have a preference to invest in the portfolio with the highest skewness and the lowest kurtosis Scott and Horvath (1980) and Dittmar (2002). Therefore, to put it simply, the higher-moment optimisation problem is one that aims to maximise the effects of odd moments while minimising even ones.

To compute these higher-order moments of the portfolio returns distribution, additional information about the behaviour of the asset returns $Y_{t+1}$ is required. For this we need a more detailed way for representing the joint distribution, and, therefore, a more descriptive specification. Hence, we can redefine

$$Y_{t+1} \sim Y_{t|t+1} \left( \mu_{t+1}, \Sigma_{t+1}, S_{t+1}, K_{t+1} \right),$$

where the additional features $S_{t+1}$ and $K_{t+1}$ represent the third and fourth co-moments of the joint distribution of $Y_{t+1}$.

As described above, for mean-variance portfolio methods, the first two moments that characterise the portfolio return are deemed to be sufficient to fully quantify its uncertainty, and, under this assumption, it would be acceptable that $Y_{t+1}$ be assumed to follow a multivariate normal distribution. However, this would be problematic for any modern day research on financial modelling due to the overwhelming amount of empirical evidence to the contrary; assuming a normal distribution to describe asset returns is an antiquated idea we thankfully now dismiss.

The statistical properties of price returns, known as the *stylised facts*, are discussed by Cont (2001) in great detail. In brief, the stylised facts describe a set of empirical regularities found in the distributions of asset returns that appear in all classes of financial assets and across all timeframes. The stylised facts relevant to the work in this thesis will be discussed as needed in later sections and chapters, but, to motivate the argument for the use of higher-order risks in portfolio selection, Cont (2001) evidences that the probability distributions of asset returns are characterised by asymmetries in shape and asymmetries in dependencies, as well as
by fat tails, meaning extreme events are more likely to happen than a gaussian distribution suggests. This means that asset returns have been demonstrated to have a tendency to display features of skewed and leptokurtic distributions rather than those of a normal distribution. These observations strongly challenge the legitimacy of the mean-variance approximation, supporting the need for higher-order moments to be considered in portfolio choice problems.

For one commonly used type of asset allocation method (expected utility asset allocation), there have, in the past, been arguments made against the need to consider higher moment risks; these go back as far as Samuelson (1975), Levy and Markowitz (1979) and Reid and Tew (1986). These arguments, however, were made under the assumption that features of the distribution are fixed and so do not change over time, which is an assumption we now firmly reject. There have been more recent contributions providing empirical evidence to support the claim that an investor’s risk preferences can be sufficiently well defined by only the mean and covariance of the joint distribution. In Das and Uppal (2004), the argument used to back this claim is that risk contained in higher moments is generally small, and only becomes relevant for portfolio choice in the extremes, where price returns depart substantially from normality. The same conclusion was reached by Jondeau and Rockinger (2009), which provided empirical evidence that the mean-variance criterion fails to approximate the constant relative risk aversion (CRRA) expected utility only in situations when there are extreme asymmetries, or fat-tailed distributions. But, as before, the major caveat to be considered is that these more recent arguments, also, consider the higher-order moments of asset return distributions to be unchanging in time.

The assumption of distributional time-homogeneity has time and time again been shown to be false. Not only does it contradict certain stylised facts but it is even contrary to the most basic ideas behind some of the most important econometric modelling methods, such as the seminal works of Engle (1982) and Bollerslev (1986), whose ARCH / GARCH framework has become the benchmark method for modelling time-varying variance. Moreover, further modelling methods influenced by these aforementioned contributions have emerged that show there are benefits to permitting parameters that affect higher moments to vary over time, see Hansen (1994), Jondeau and Rockinger (2003b) and Harvey (2013). For multivariate time-series, Engle (2002), Patton (2004), Cappiello, Engle and Sheppard (2006), Pelletier (2006) and Jondeau and Rockinger (2006b) all have made significant contributions showing that dependence structures also are time-varying. It is now generally agreed upon that all moments and co-moments of financial price return distributions are continuously changing over time.

As suggested earlier, a probabilistic model of a multivariate gaussian is unable to model the higher-order behaviour needed for higher order portfolio optimisation methods. However, when used in combination, the resulting mixture model can capture non-linearities and asymmetries of
a price returns distribution, and, therefore, can forecast the moments required for higher-order portfolio optimisation. Similarly, by embedding gaussian models as state-dependent error distributions within a regime-switching model, the predictive distribution becomes a gaussian mixture that can vary over time; this method forms the basis of the framework explored in the early work on the implicit use of higher moments in dynamic portfolio strategies, as in Ang and Bekaert (2002), Pelletier (2006), Guidolin and Timmermann (2007), Guidolin and Hyde (2008), Guidolin and Nicodano (2009) and Tu (2010). This early work accounted for the problem of time-varying higher moments implicitly, through the evolving dynamics of regime-switching between multivariate gaussian models. The last two studies in the above list show particularly well how regime-switching models provide economic benefits over non-switching (static) models when used for dynamic asset allocation.

An alternative to the gaussian mixture approach described above is a probabilistic model of a distribution that comes from a family of parametric distributions which includes, in addition to location and scale parameters, one or more shape parameters. The parameters can then be made to vary over time by conditioning on an information set. Shape parameters, as the name suggests, directly affect the shape of the distribution, allowing for deviations away from normality that are consistent with asset returns; typically there will be one shape parameter that affects the skewness of the distribution and possibly another that affects kurtosis. Time-series models with conditional shape parameters were first seen in Patton (2004), where the effects of skewness and asymmetric dependence on portfolio performance were investigated; he found evidence that including a time-varying third moment in the optimisation was preferable to its exclusion. A different method in Martellini and Ziemann (2010), where the moments themselves were estimated, found that whether or not the higher-order portfolio method is superior to mean-variance is dependent on the quality of the estimator, with poor quality estimations of the higher-order moments having no economic impact on higher-order portfolio selection problems. This last finding is of particular interest, as it offers a plausible explanation as to why the research mentioned earlier on the importance of higher moments in portfolio optimisation did not succeed in demonstrating their value, under the assumptions made in those papers, which would have led to poor estimators. Finally, the results in C. R. Harvey et al. (2010) suggest that it is important to incorporate higher order moments in portfolio selection problems, but did not present results for out of sample performance.

The most influential study on the value of higher-order moments in portfolio optimisation was that of Jondeau and Rockinger (2012); this study investigated the use of modelling methods for forecasting entire multivariate distributions, with the time-varying forecasts used to test the importance of considering higher moment risk in dynamic asset allocation strategies. Their method modelled parameters related to the first four moments of the joint asset returns
distribution, and tested the efficacy of higher-order portfolio strategies, in a dynamic setting, against strategies that did not include the third and fourth moments. They showed that when models of distributions permit time-variability, higher moment optimisation is superior to mean-variance out of sample, thus supporting earlier findings in Patton (2004). To describe this type of investment strategy, Jondeau and Rockinger (2012) introduce the term distribution-timing, which is a term that can be considered analogous to better known terms such as market-timing and volatility-timing; distribution-timing describes a strategy in which managers dynamically re-allocate portfolios given evolving knowledge of the entire joint probability distribution. The dynamic asset allocation method used by these authors for testing portfolio performance has been duplicated in a number of more recent contributions, e.g. Bernardi and Catania (2018), and is also the dynamic asset allocation method used in the experiments in Chapters 5 and 6 of this thesis.

Taken together, the studies mentioned above, and others not included for brevity, provide good evidence to suggest that, when high quality time-varying forecasts are available, portfolio optimisation strategies based on the first four moments provide a better approximation of the expected utility than mean and variance alone, and that it is of economic benefit for dynamic portfolio strategies to make use of predictive models that have the capacity to produce time-varying forecasts of the first four moments of the asset returns distribution.

### 2.4 MODELLING METHODS

The time-series of financial asset returns have some particular, unique, statistical qualities which make it challenging to forecast the inputs for portfolio optimisation methods. As mentioned in the previous section, for portfolio methods to work properly, especially as part of a dynamic allocation strategy, the quality of the forecasts is the most important factor that will influence the investment outcomes. This section takes a look at the most influential modelling methods introduced to model the distinct properties of financial time-series, covering both univariate and multivariate time-series models.

Arguably, the main reason why financial time-series forecasting is so tricky, as was briefly discussed towards the end of the previous section, is that when we consider price returns as random variables, the distribution associated with the random variables will tend to change over time. In other words, price returns time-series have different statistical properties at different points of the time-series; this is commonly referred to as data non-stationarity (or
simply non-stationarity), and is well-known to be a prevalent property of financial time-series data. Some of the more observable effects of non-stationarity in financial time-series include: the periodic emergence and passing of trends and momentum, where means change over time; heteroskedastic effects in variance (Engle (1982) and Bollerslev (1986)) and volatility clustering (Mandelbrot (1997)), a term describing the irregular back-and-forth between episodes of high volatility and low volatility; time-varying correlations (Cappiello, Engle, and Sheppard (2006)); and evidence for distinct market regimes (Hamilton (1989)). There are many more examples of how non-stationarity manifests as observable phenomena in financial time-series. In fact, it is possible that non-stationarity can be used to explain many of the stylised facts of asset returns; this claim, however, will be left as conjecture.

Financial econometric models, and, more generally, time-series models, attempt to address non-stationarity in two main ways, hence can be considered to fall into two main categories, with the commonality that the assumed distribution will, in some way, be permitted to vary over time. One of these two ways will be discussed later in this section; this methodology facilitates time-variation in forecasts via a mechanism to vary the structure of the model. With the other methodology, discussed next, time-variation in the forecasts is driven by conditioning certain features of a time-series distribution on an evolving information set; this can include past observations of the time-series, or come from exogenous sources.

The original, and most uncomplicated, econometric modelling framework for time-series contains classes of auto-regressive (AR) models; these are collectively known as auto-regressive integrated moving average (ARIMA) models. Their function is to forecast the future value of a time-series given past observed values. The effective probability distribution (also referred to as the error distribution) is assumed to be normal with a time-varying conditional mean and constant variance. Choosing to model the time-series of financial price returns using these methods makes the assumption that variance is constant over time; as mentioned above, this is not what is observed empirically, and hence to rely on this assumption can cause these models to be untrustworthy. In order to rectify the problem, variance also needs to vary, to account for heteroskedastic effects in the residuals. This was first achieved in the seminal works of Engle (1982), Bollerslev (1986), and Engle and Bollerslev (1986). These combined works introduced the autoregressive conditional heteroskedasticity (ARCH) model and its generalisation, the GARCH framework. These models were introduced to account for time-variation in asset price volatility by correcting for the variations in variance in the standardised errors. Extensions and alternatives to the GARCH framework include the GARCH-X model of Wang and Wang (2022), the HEAVY model of Shephard and Sheppard (2010) and the realised GARCH model of Hansen, Huang, and Shek (2012). Early phases in the gradual development of the GARCH framework saw the relaxation of the assumptions on the form of the error distribution; for example, Engle
and Bollerslev (1986) used the Student’s t distribution, this relaxation being further extended to the generalised error distribution (GED) in Nelson (1991). The conditional estimation of the variance has two main purposes: first, and the main reason why GARCH was so important at the time, is that it allowed for efficient econometric inference of the conditional mean in the presence of time-varying variances; second, and the reason why it is still important today, is that it is a method of estimating volatility of asset prices, which is required in many fields of risk management and quantitative analysis of financial markets (see Bollerslev, Chou, and Kroner (1992) for a review).

In addition to models of the conditional mean and variance, as mentioned above, conditional dependence structures have also been a well-researched area in financial econometrics. Early models in this area, which remain some of the most successful and well-known models, include the multivariate GARCH (MGARCH) of Y. K. Tse and A. K. C. Tsui (2002), and the dynamic conditional correlation (DCC) model of Engle (2002); both of these have become benchmark methodologies for scale and dependence estimation and are still used widely today. Extensions to the MGARCH and DCC models have included the ADCC model of Cappiello, Engle, and Sheppard (2006), which allows asymmetric correlations, which, as already mentioned, are an important stylised fact of financial returns data. Another important advancement is the generalised auto-regressive score (GAS) model of Creal, Koopman, and Lucas (2013) (see also Creal et al. (2014) and Creal and Tsay (2015), Salvatierra and Patton (2015)). The GAS framework is a flexible way to model the dynamic dependencies of asset returns. In fact, it extends to estimation of all time-varying parameters of a conditional distribution; the GAS model will be discussed more later in this section.

One type of time-series model where the model structure varies over time is based on the idea of market regimes; this encompasses an entire class of time-series models known as regime-switching models (RSM). The most commonly seen type of RSM is one whose forecasts are, primarily, conditioned on the state of an unobservable Markov chain, where states represent distinct regimes, and each regime is modelled with a distinct state-dependent model; each regime therefore has a model with parameters that characterise parts of the time-series that fall into that specific regime. The Markov chain provides a mechanism to switch between these models, or mix them based on regime uncertainty. This mechanism results in the model structure changing over time as the uncertainty over the states changes. The simplest form of RSM is one with the state-dependent distributions parameterised by constant means and variances, e.g. Garcia, Perron, et al. (1991). However, RSMs are more effective when state-dependent distributions are represented, in some way, by using regression type models. In these cases, the functional relationship between the response variable and the conditioning variables is able to change over time; switching regression models of this type were first introduced by
Goldfeld and Quandt (1973). For financial price returns modelling, such models were followed by Hamilton’s seminal work on switching AR models (Hamilton (1989)), thus establishing the concept of regime-switching models of the conditional mean; in this case, mean forecasts are conditioned on lagged observations. The idea was later extended to regime-switching of both mean and variance in Brunner (1991), and developed further in Hamilton and Susmel (1994); this method used an asymmetric function for the volatility process, so that negative price shocks contribute more than positive shocks, therefore being the first RSM model to allow for time-varying shape in the state-dependent distributions. A further development proposed in Haas, Mittnik, and Paolella (2004a), and in Haas, Mittnik, and Paolella (2004b), was to allow the model to switch between a set of independent GARCH equations evolving in parallel according to different sets of parameters.

Building upon the early auto-regressive modelling methods for models for the estimation of conditional mean, variance and dependence, an important advancement came in Hansen (1994) who introduced a distributional auto-regressive modelling methodology known as conditional density estimation, a method that would come to inspire a wide range of contributions to the literature on higher-order modelling for univariate time-series, see Theodossiou (1998), Harvey and Siddique (1999) and Jondeau and Rockinger (2003b). These contributions, and many more, extended Hansen’s distributional regression method to estimate time-varying parameters which determine the location (mean), scale (variance), and shape (higher-order moments) of a distribution. Similar approaches have been developed for joint probabilistic forecasting for multivariate time-series. Early influential contributions that allowed for conditional time-variation included the dynamic conditional correlation (DCC) model of Engle (2002) and Engle and Sheppard (2001), and the related multivariate GARCH class of models, see Y. Tse and A. K. Tsui (2001) and for a comprehensive survey see Bauwens, Laurent, and Rombouts (2006). For early work on multivariate observation-driven models for time-variation in the shape of a distribution, see Bauwens and Laurent (2005), De Luca, Genton, and Loperfido (2006), Franceschini and Loperfido (2010), Jondeau and Rockinger (2006b), Jondeau and Rockinger (2012). A more recent addition to the class of observation-driven models is the multivariate generalised autoregressive score (GAS) model of Creal, Koopman, and Lucas (2011), Creal, Koopman, and Lucas (2013) and Harvey (2013). The GAS framework allows for the modelling of time-variability within all parameters of any specified distribution, and is an especially flexible framework for multivariate financial time-series modelling. By relying on the full density structure to update the time-varying parameters, GAS models capture additional information in the data that is not exploited by GARCH type models, which makes GAS models powerful tools for forecasting that often lead to important forecasting gains over other classes of observation-driven models. This performance advantage was shown in Koopman, Lucas, and Scharth (2016), who found that the GAS models outperform many of the familiar observation-driven autoregressive conditional
parameter models previously mentioned. In addition, the GAS framework is one that can be easily adapted to more complex multivariate models, and even be augmented to accommodate exogenous information, as shown in Salvatierra and Patton (2015), Bernardi and Catania (2019).

To capture all possible asymmetries and non-linearities in a single, univariate, financial time-series model, it is important to consider the factors that result in deviations in the shape of the distribution from normality; primarily, we observe this as non-zero skewness and excess kurtosis in the returns distribution. Likewise, for models for multiple time-series, it is important to consider the possible deviations from multivariate normality, a prime example of such a deviation being asymmetric dependence (for a thorough discussion of this concept see Patton (2006)). As an example to illustrate this particular type of asymmetry, two returns time-series will tend to exhibit higher correlation during market downturns than in market upturns. Asymmetric dependence is particularly important to consider for asset allocation problems, as it manifests as negative skewness in the portfolio returns distribution, a quality that a rational investor will, all else being equal, want to minimise. For a model to be able to capture these types of asymmetries, it cannot rely on traditional elliptical distributions such as the multivariate gaussian and Student’s t, both of which assume symmetric dependence in the form of linear correlation. Measuring the relationship between the time-series using these assumptions can give misleading results when the shapes of the marginal distributions are not the same, or when variables co-vary in a non-linear way, which is unfortunately the case with financial price returns, see Goodwin and Leech (2006). Copula dependence models, a highly flexible method that can capture the diversity of dependence relationships, are able to address these problems. Copula models are a very effective way to represent dependence structures with different types of non-linear and asymmetric properties that would be missed using traditional multivariate distributions. Copula models consist of a decomposition of the multivariate distribution into a set of marginal distributions for each univariate variable, and a copula function containing a model of the dependence structure. It is the copula functions that provide the flexibility to cover a wide range of relationships, as copula densities can take on many different forms, especially in the bivariate case, and account for many of the types of behaviour we observe between the time-series of different asset returns. Using a copula decomposition also permits a more flexible and better-specified choice for the marginal density functions than is possible by using out-of-the-box multivariate distributions. For example, the multivariate Student’s t is parametrised by a single shape parameter, ‘degrees of freedom’, used to specify the shape (heavy tailedness) for all assets (marginals); this is highly unlikely to allow for sufficient flexibility, as different returns series tend to have different shapes of distribution and therefore exhibit different behaviour in the tails, see Bollerslev (1987). A copula decomposition removes this restrictive assumption and allows each of the univariate marginals to be individually specified, with its own unique parameter set, one that has been tailored for its own specific behaviour. Time-
varying (conditional) copula dependence models were pioneered by Patton (2006), who showed how the parameters of copula functions could be made to vary over time by conditioning on a set of exogenous regressors (covariates). This method was further advanced by the influential work of Jondeau and Rockinger (2006b), who introduced the copula-GARCH model, a true multivariate probabilistic forecasting model that permitted all distributional parameters to vary over time, with GARCH-style evolution equations. In Creal, Koopman, and Lucas (2011) models of time-varying copula were introduced, in which parameters are allowed to vary through time, driven by GAS dynamics. Factor models were used as the conditioning mechanism adopted by Creal and Tsay (2015) in their study of high-dimensional financial time-series modelled with time-varying copula. More recent examples of the use of conditional copula in practical applications can be found in Lang et al. (2019), who create a model of wind location, and scale of wind speed and direction, by using a bivariate gaussian copula with parameters estimated using a set of external covariates; see also Hu (2010) for an analysis of the dependence structures in Chinese and U.S. financial markets using time-varying conditional copula, and Xu and Lien (2020), where the GAS framework is applied to a multivariate copula model in which both the univariate marginal distributions and the dependence parameters vary under GAS dynamics; their model is used for probabilistic forecasting for hedging of crude oil and natural gas futures.

An alternative way of introducing time-variation into copula dependence models is to use an RSM in the way discussed earlier in this section, i.e. having a set of state-dependent models of copula densities, each with a constant parameter set, and allowing switching conditional on the state of a Markov chain, thus creating a regime-switching dependence model, as seen in Jondeau and Rockinger (2006b). For switching of the entire joint distribution, not only the dependence, Rodriguez (2007) introduced a regime-switching bivariate copula model for investigating financial contagion; a method similar in design was used in Okimoto (2008) to examine global equity markets, with findings that reject the use of multivariate regime-switching gaussian models in favour of a copula model for the dependence structure. These two studies considered only the bivariate copula case; for multivariate distributions Chollette, Heinen, and Valdesogo (2009) introduced a multivariate regime-switching model with state dependent dependence structures using canonical vine copula. Canonical vine copula, of Aas et al. (2009), are a particular specification class of a pair copula construction (PCC), as will be discussed in Chapter 5, and are a highly flexible means of capturing rich dependence structures of higher dimensional distributions by using bivariate copula as building blocks. The finding in Chollette, Heinen, and Valdesogo (2009) was that models using canonical vines are generally superior to alternative dependence structures. More recently, extending the regime-switching constant correlation model of Pelletier (2006) to copula densities, Yamaka, Phadkantha, and Sriboonchitta (2019) presented a switching copula model of commodity futures, and based on in-sample statistical performance have suggested that their model is superior to the single
regime copula. Using a time-varying copula mixture distribution to model the joint behaviour of the spot clean spark spread and the daily wind index, Christensen, Pircalabu, et al. (2017) showed that accounting for asymmetry, tail dependence, and time-variation in the dependence structure is especially important in the context of risk management of energy futures.

In their study on high-frequency exchange rate data, Dias, Embrechts, et al. (2004) introduced a time-varying GARCH–type conditional copula model. An important insight from their study was the detection of notable change points, or structural breaks, in the data. This suggests that as well as time-variation through conditioning variables being important, an additional regime-switching mechanism may also be needed to handle certain types of dependence related behaviour. This idea was taken up later in Bernardi and Catania (2019), who introduce a model with both time-varying conditional parameters and regime-switching dynamics. More specifically, in their novel class of model, the dependence parameters evolve according to Markov-switching GAS dynamics. In this model, each state is represented by a GAS model with all copula parameters varying over time. The GAS and GARCH frameworks, which comprise the bulk of research on time-varying parameters, have a dynamics evolution that is primarily driven by past observations in the time-series. It is highly likely, however, that parameters of the marginal distributions, as well as those in the dependence model, are highly affected by exogenous, as well as endogenous, information. However, for the multivariate case especially, there have been relatively few contributions to the econometrics literature which consider time-variation conditioned on exogenous covariate. Notable examples can be found in Salvatierra and Patton (2015), who augment a GAS model to allow exogenous information derived from high-frequency data; Marra and Radice (2017), who apply generalised additive models to estimate the parameters of bivariate copula (see also Wojtyś, Marra, and Radice (2018)); and the use of generalised additive models as predictors for the dependence parameters of multivariate distributions, as first presented in Vatter and Chavez-Demoulin (2015), and later generalised to pair copula constructions in Vatter and Nagler (2018). Finally, in this list of uses of exogenous information in probabilistic forecast models, the switching GAS model of Bernardi and Catania (2019) is in Bernardi and Catania (2018) augmented to exploit a small set of exogenous macroeconomic covariates; the authors apply their model to a portfolio of financial assets, providing forecasts of time-varying conditional co-moments up to the fourth order. These moments are used as inputs to a dynamic portfolio allocation strategy. The authors find firstly that their method vastly improves portfolio performance compared to other methods, and secondly that including the exogenous covariates improves performance over the standard GAS set-up.
2.5 DISTRIBUTIONAL REGRESSION

The past decade has seen the emergence of a new paradigm for regression-type modelling, with models that treat the mean of a response variable’s distribution as just one of many quantities to be estimated, and not necessarily even the most important. As a result, attention is shifting from traditional regression models that focus only on the conditional mean, to probabilistic distributional models that provide a joint framework to estimate all parameters of the response variable’s distribution, as functions of exogenous variables. So far in this chapter, the vast majority of conditional probabilistic models mentioned for financial time series modelling have been of a class that primarily utilises past observations of the time-series to drive the dynamics of the parameter forecasts. Distributional regression models (DRMs) are effective in situations where it is highly likely that other properties of the response distribution, such as the variance, skewness, or kurtosis, will also depend on exogenous variables. As explained in the previous section the non-stationarity of financial time-series data means these properties change over time; as such, distributional regression, in this setting, is an approach whereby we can obtain estimates that allow for this time variation. When applied to time series forecasting, the resulting DRM has the capacity to provide full time-varying probabilistic forecasts, i.e. one step ahead predictions of all parameters of the conditional response distribution. In addition, DRMs have the capacity to provide conditional estimates of quantiles of distributions and other related probabilistic quantities. This is particularly useful in systematic trading, portfolio selection and general risk management problems in finance. As an example, a trading strategy may require an estimate of the probability that a position about to be entered into will beat the bid-ask spread and therefore earn a profit, or require up to date conditional estimates of a portfolio’s value at risk (VaR), computed using estimated conditional higher moments. These features allow for estimates of risk that consider asymmetries and rare events (tail risk). For modern portfolio allocation and risk management, risk measures derived from higher moments are nowadays thought of as being essential to properly assess the risks associated with potential trading positions.

A standard framework for distributional regression has emerged in the form of the generalized additive model for location shape and scale (GAMLSS) models (R. Rigby and D. Stasinopoulos (2001) and Rigby and D. M. Stasinopoulos (2005), Mayr et al. (2012) and Hofner, Mayr and Schmid (2014)). GAMLSS models are a family of semi-parametric regression models in which all parameters of a response distribution can be modelled concurrently within a flexible framework. The main feature of GAMLSS is that every parameter of the conditional distribution is modelled
by its own prediction function. While traditional regression models are primarily interested in the conditional mean of the response variable, treating other distributional parameters as fixed estimates, the GAMLSS approach allows for the regression of each distribution parameter on its own set of explanatory variables. Thus, in the GAMLSS approach, the full conditional distribution of a multi-parameter model is related to a set of exogenous variables. The backbone of this method is the generalised additive model (GAM) of Hastie and Tibshirani (1990). This method consists of a predictor with an additive structure which allows for a wide range of different types of prediction functions to be combined together. The additive components can be linear, non-linear, or smooth functions of the covariates; even decision trees, random effects or spatial effects may be combined in the additive structure.

This additive structure of the GAM predictor functions is suited to a machine learning technique for building the models. Gradient boosting (see Mayr et al. (2010) and Mayr et al. (2012)) is a learning method which has been shown to build models with superior generalisation, primarily due to a robustness to overfitting. This technique is advantageous for situations where we have a large number of features with most, or all, displaying only weak correlation with the response variable. Gradient boosting also allows for automatic feature selection; moreover, component-wise gradient boosting, as used in the following chapters, can choose from a high number of prespecified base models allowing the data to decide the best model specification for each covariate (Mayr et al. (2012)). In a regime switching setting, one state model may find that a particular set of covariates describes the data best, and another state model find a different set. Therefore, different market conditions can be explained with different model specifications, where each specification is determined automatically from the data. Finally, with gradient boosting, we can still discover informative models even when the number of observations is less than the number of covariates, a situation that can arise when using a Markov-switching framework.

Until recently, DRMs have considered only univariate response distributions. However, over the past few years, researchers have noted value in extending DRMs to multivariate analysis. This is achievable due to the flexibility of the GAMLSS framework, which makes it suitable for conditional dependence modelling. Following the approach of Klein et al. (2015) and Klein and Kneib (2016) who extended generalised additive modelling to multivariate normal and Student’s t distributions, Marra and Radice (2017) introduces a bivariate copula extension to the GAMLSS framework; this was further extended by Vatter (2016) and Vatter and Nagler (2018) to the multivariate setting using conditional pair copula constructions. This thesis extends this earlier work on distributional regression models to multivariate time series with regime-switching dynamics. It builds on the work on switching GAMLSS for univariate responses which were
recently introduced in Hambuckers et al. (2018) and Adam, Mayr and Kneib (2017). The following chapter provides a complete background to these approaches.
This chapter contains detailed technical background for the three main topics that underpin the models that will be introduced in later chapters. The content is composed in a pedagogical manner and is self-contained; it can therefore be used as a complete reference, though a reader familiar with the material may wish to proceed straight to the following chapter. The three topics covered here are distributional regression and the GAMLSS frameworks, followed by gradient boosting methods used for model estimation, and, finally, regime-switching models. Additional technical information specific to an individual chapter can be found in the relevant chapter.
Regression methods are essential tools when performing statistical analysis as they are a powerful way to identify relationships between a response variable and a set of exogenous variables. Many different kinds of regression-type models and associated modelling frameworks have been developed over the past few decades; all of these methods can be thought of as descendants of the classical linear model (LM), where the conditional mean of the response is modelled through a linear function of the covariates. The LM model has been extended in many ways; some of the most important extensions include:

- **Generalised linear models** (GLMs, Nelder and Wedderburn (1972)) extend linear modelling to any type of response variable with a distribution in the exponential family by using a
3.1 REGRESSION FRAMEWORKS

nonlinear link function to transform the linear prediction function to the mean of the response distribution (Figure 1, top right).

- **Generalised additive models** (GAMs, Hastie and Tibshirani (1990)) extended GLMs to allow for smooth nonlinear effects in the covariates (Figure 1, bottom left).

- **Generalised additive models for location, scale, and shape** (GAMLSS, Rigby and D. M. Stasinopoulos (2005)) extended GAMs to a full probabilistic regression model. In GAMLSS, each parameter of a probability distribution can depend on an additive predictor of the covariates, allowing for linear and/or smooth nonlinear effects (Figure 1, bottom right).

More often than not in regression analysis it is the conditional mean that is the main quantity of concern. GLMs and GAMs are examples of regression frameworks that serve the purpose of conditional mean estimation very well. However, for many problems, the conditional mean may not be the only, or even the most important, feature of the response variable that requires estimation, meaning that the problem may require a holistic approach, with other quantities, such as variance, also being dependent on exogenous variables. For these types of problems, distributional regression has emerged as a powerful way to obtain conditional estimates of all desired features of a response variable’s probability distribution.

For example, in financial risk management, conditional forecasts of variance, or higher moments of a distribution, are important quantities that may need to be estimated as functions of known data. Similarly, in other research areas, e.g. weather forecasting, there has been an increased interest in estimation of entire probability distributions, and distributional regression models have emerged as the means to achieve this. GAMLSS models have become one of the standard frameworks for conducting this type of analysis. Building upon the GAMs, GAMLSS models allow all parameters of a response variable’s distribution to be modelled as smooth functions of a set of exogenous variables.

Figure 3.1 shows examples of the four main regression frameworks discussed in this chapter, and is useful to illustrate the differences between them. The bottom row of the figure shows examples of GAM and GAMLSS fits. In these examples the response variable is normally distributed, with a conditional mean which is a nonlinear function of a single covariate. The GAM is suitable when the response variable is distributed with a constant variance (Figure 1, bottom left). However, when the variance of the response variable is also conditional (bottom right) it is more appropriate to model the data with a GAMLSS.

This section provides an overview of the four types of regression frameworks in Figure 3.1, each being an extension of the former, building up to a comprehensive description of the GAMLSS framework, along with methods for estimating these classes of models.
### 3.1.1 Linear Models

Linear regression models are the best known way of estimating the conditional mean of a random response variable as a function of a set of exogenous variables. Given the response variable $Y$, and $r$ exogenous covariates $x_1, x_2, \ldots, x_r$, the linear model has the form

$$y_i = \beta_0 + x_{i,1}\beta_1 + x_{i,2}\beta_2 + \ldots, x_{i,r}\beta_r + \epsilon_i,$$

$$\epsilon_i \overset{i.i.d.}\sim \mathcal{N}(0, \sigma^2), \forall i.$$

Therefore for a sample size of $n$, all $\epsilon_i$, for $i = 1 : n$, are independently and identically distributed (i.i.d.) with normal (Gaussian) distribution. The model can equally be represented by

$$Y_i \overset{i.i.d.}\sim \mathcal{N}(\mu_i, \sigma^2), \forall i,$$

$$\mu_i = \beta_0 + x_{i,1}\beta_1 + x_{i,2}\beta_2 + \ldots, x_{i,r}\beta_r + \epsilon_i.$$ (3.1)

We can rewrite Equation (3.1) in vector form as

$$\mathbf{Y} \overset{i.i.d.}\sim \mathcal{N}(\mathbf{\mu}, \sigma^2),$$

$$\mathbf{\mu} = \mathbf{X}\beta,$$ (3.2)

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ is a vector of response variables, $\beta = (\beta_0, \beta_1, \ldots, \beta_r)^T$, and $\mathbf{X} \in \mathbb{R}^{n \times p}$ is a design matrix. In this matrix, where $p = r + 1$, the first column is a column of ones and the remaining $r$ columns contain the covariates. Additionally $\mathbf{\mu} = (\mu_1, \mu_2, \ldots, \mu_n)^T$ and $\sigma^2 = (\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)^T$ are, respectively, the conditional means and a vector of constant variances, i.e. $\forall_{t=1:n}\sigma_t^2 = \sigma^2$. These two quantities have to be estimated from the data, which can easily be done using the ordinary least squares (OLS) method to find a linear prediction function. The least squares method finds a vector of regression coefficients that minimises the sum of squared residuals $\epsilon^T\epsilon$,

$$\hat{\beta} := \operatorname*{argmin}_{\beta} (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta),$$

which has the solution $\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$. Hence, we get $\mathbf{\mu} = \mathbf{X}\hat{\beta}$ as the prediction function of the conditional mean and $\sigma^2 = \frac{\epsilon^T\epsilon}{n-p}$ as a (constant) unbiased estimator of the variance.
3.1.2 Generalised Linear Models

The above framework is not suitable for many applications as, for it to be effective, there are a set of assumptions that must be met. A linear model of the form Equation (3.2) ideally requires residuals $\epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$, $\forall i$. This may occur naturally when the response variable $Y$ is normally distributed, and will imply that a change in a covariate has a linear effect on the response variable and therefore the response is able to vary indefinitely in both directions. There are many situations for which we would like to use linear regression to estimate the conditional mean, but where the response variable will come from some other distribution and so not vary as a linear function of the covariates. For example, in situations where the response variable would always be expected to have positive values, and also expected to vary over a wide range, changes in the covariates will lead to geometrically (i.e. exponentially), rather than constantly varying, changes in the response variable. The generalised linear model of Nelder and Wedderburn (1972) (GLM) was introduced as an extension of the standard linear modelling framework shown above in order for linear models to be able to accommodate a much wider variety of problems. A GLM model can be defined as

$$Y_i \overset{i.i.d.}{\sim} \mathcal{E}(\mu_i, \phi), \forall i,$$

$$g(\mu_i) = \beta_0 + x_{i,1}\beta_1 + x_{i,2}\beta_2 + ..., x_{i,r}\beta_r + \epsilon_i,$$  \hspace{1cm} (3.3)

where $\phi$ is a dispersion parameter, $g$ is a link function, and $Y$ is no longer restricted to the normal distribution but can be of any type of exponential family distribution. The exponential family of distributions is a collection of common probability distributions; a distribution belongs to this family if it has a probability density function $f(y|\theta, \phi)$ that takes the form

$$f(y|\theta, \phi) = \exp\left((y\theta - b(\theta))/a(\phi) + c(y, \phi)\right),$$  \hspace{1cm} (3.4)

where $\theta$ is the canonical parameter of the distribution, $\phi$ is a scale parameter, and $a(), b(), c()$ are functions that characterise the particular family of distribution. The form of Equation (3.4) includes many important types of distribution, including the exponential, gamma, gaussian, Poisson, inverse gaussian, and binomial. The mean and variance of $Y$ can be written in the form

$$\mathbb{E}[Y] = \mu$$

$$= b'(\hat{\theta}),$$  \hspace{1cm} (3.5)

$$\text{Var}[Y] = a(\phi)V(\mu)$$

$$= a(\phi)b''(\hat{\theta});$$  \hspace{1cm} (3.6)
therefore, the mean is a function of the canonical parameter, and $V(\mu)$, known as the variance function, illustrates the second important feature of the GLM, i.e. the variance is a function of the conditional mean.

Within the exponential family, the choice of function linking the output of a linear predictor to any canonical parameter of a probability distribution is made easy, as for every such distribution there exists a canonical link such that $g(\mu) = \theta$, which can be obtained from Equation (3.5). However, it is not necessary to use this choice and, in practice, simply mapping the output of the predictor to the range of the mean is sufficient. For example, if we have a $\theta$ restricted to the range $0 < \theta < \infty$ then the linear prediction function $\eta(x) = x\beta$, is mapped to

$$\eta(x_i) = \log(\theta_i),$$

which has the inverse

$$\theta_i = \exp(\eta(x_i)).$$

The GLM can be written in vector form as

$$Y \overset{i.i.d.}{\sim} E(\theta, \phi),$$

$$g(\mu) = \eta(X) = X\beta,$$

with $y, X, \beta$ and $\mu$ defined in Equation (3.2), $\phi$ a vector of constants, $\eta(.)$ the linear prediction function and $g(.)$ the link function. To estimate the model an iteratively reweighted least squares algorithm is used. A complete derivation of this algorithm can be found in Stasinopoulos et al. (2017). Algorithm 1 summarises the procedure for learning the predictive function $\eta$.

Algorithm 1 shows that at each round, the procedure involves regressing $X$ on and adjusted dependent variable $z$ with weights $W$ to obtain a revised estimate of $\beta^{[m+1]}$. Then a new $\mu$, and $\eta^{[m+1]}$ are computed, followed by the computation of new adjusted dependent variables $z$, and the process is repeated, until convergence.
Input: $X \in \mathbb{R}^{n \times p}, Y \in \mathbb{R}^n$, link function $g$, and distribution $F = \mathcal{E}(\theta, \phi)$.

Select method by setting the function $\alpha$:

$$
\alpha(\mu_i) = \begin{cases} 
1 & \text{Fisher Scoring} \\
\frac{1}{1 + (y_i - \mu_i)} \left( \frac{V''(\mu_i)}{V'(\mu_i)} + \frac{g''(\mu_i)}{g'(\mu_i)} \right) & \text{Newton–Raphson}
\end{cases}
$$

Initialise: $\mu_i = y_i, \eta = g(y_i), \forall i$

• Do:

  - Compute: $z_i = g'(\mu_i)(y_i - \mu_i)/\alpha(\mu_i) + \eta(X_i), \forall i = 1:n$.
  - Compute: $w_i = \frac{\alpha(\mu_i)}{g'(\mu_i)^2 V(\mu_i)}, \forall i = 1:n$, then set $W = \text{diag}(w)$.
  - Estimate $\hat{\beta}$s by regressing $X$ on $z$, using a weighted least squares with weights $w$:
    $$
    \hat{\beta} = (X^T W X)^{-1} X^T W z
    $$
  - Update $\mu_i = g^{-1}(\eta(X_i)), \forall i = 1:n$.

• until convergence, i.e. parameter estimates no longer change, or a likelihood convergence criterion is met.

Output $\hat{\beta}$s.

**Algorithm 1**: The iteratively reweighted least squares algorithm for fitting GLMs.

### 3.1.3 Generalised Additive Models

Although GLMs offer an attractively simple solution for estimating the conditional mean of any exponential family distribution, they will often fail in real life situations where the effect of a covariate is not linear with respect to $g(\mu)$. Generalised additive models (GAMs) Hastie and Tibshirani (1990) extend the class of GLM models by replacing linear terms $\sum_j x_j \beta_j$ with more general functions $\sum_j f_j(x_j)$. These functions can be non-parametric, semi-parametric, or have a pre-specified parametric form. The type of covariate effect can be individually specified for each covariate $x_i$, by choosing the appropriate form of its corresponding predictor function $f_j$.

As with linear models, the functions are then combined in an additive way to form the GAM, a natural extension to GLMs that could accommodate the non-linear smoothing techniques that became popular in the late 1980s. The flexibility of this framework permits a wide range of different types of non-linear smoothers to be included as model components, but is not restricted to this class; in fact, linear functions, interaction terms, and smooth functions could all be used together to create a more powerful predictor.
Let \( \{(y_i, x_i)\}_{i=1}^n \) be a set of realisations of pairs of random variables \((Y, X)\) where \(Y\) is a univariate response variable and \(x_i \in X\) is a \(r\)-dimensional vector of covariates. \(\mu = \mathbb{E}[Y|X]\) is the conditional mean with \(Y \sim \mathcal{E}(\theta, \phi)\) and \(g\) is a link function as defined for GLMs. For each of the exogenous variables \(x_j, j = 1 : d\) there is a function \(f_j\) that, through the link function \(g\), represents the relationship between the \(j\)th covariate and the response variable. The general form of \(f_j\) provides a flexibility beyond that of GLMs by relaxing any assumptions on the form of this relationship.

The standard form of the GAM can be written as

\[
Y_i \overset{i.i.d.}\sim \mathcal{E}(\mu_i, \phi), \forall i,
\]

\[
g(\mu_i) = \eta(x_i) = \beta_0 + f_1(x_{i,1}) + f_2(x_{i,1}) + \ldots + f_r(x_{i,r}) + \epsilon_i,
\]

where \(\beta_0\) is an intercept term \(f_1 \ldots f_r\) are additive predictors that represent the relationship between covariate \(j\) and \(Y\) via the link function \(g\), and \(\epsilon\) is a disturbance term.

The GAM can also be written in a semi-parametric form

\[
g(\mu) = \eta(X) = X\beta + \sum_{j=1}^r s_j(X_j),
\]

with the first term on the r.h.s. indicating an LM as defined in Section 3.1.1 with \(\beta = (\beta_0, \beta_1, \ldots, \beta_r)^T\) and \(X \in \mathbb{R}^{n \times p}\) a design matrix, where \(p = r + 1\), and with the first column of the matrix being a column of ones. The \(s_j(X_j)\) represent smooth functions of each covariate; these can be parametric, semi-parametric or non-parametric smoothers. The flexibility of GAMs to allow for non-parametric fits, without assumptions or prior knowledge about the different types of relationships between the response variable and covariates, is a very powerful improvement over the limitations of the GLM.

There are several model estimation methods that can be used to fit a GAM. For a standard additive model, the conventional method is the backfitting algorithm of Breiman and Friedman (1985). Backfitting works by iterative smoothing of partial residuals and provides a general estimation method capable of using a wide variety of methods to estimate the individual univariate functions \(f_j(x_j)\) that make up the additive prediction function. In contrast, for a GAM, a weighted form of the backfitting algorithm is nested inside an outer iterative procedure similar to the iteratively re-weighted least squares algorithm used for estimating GLMs Algorithm 1; this weighted form is known as the local scoring algorithm, which is outlined in Algorithm 2.
Input: $X \in \mathbb{R}^{n \times p}, Y \in \mathbb{R}^n$, link function $g$, and distribution family $\mathcal{F} = \mathcal{E}(\mu, \phi)$.
Select the estimation method by setting the function $\alpha$:

$$
\alpha(\mu_i) = \begin{cases} 
1 & \text{Fisher Scoring} \\
1 + (y_i - \mu_i) \left( \frac{V''(\mu_i)}{V'\mu_i} + \frac{g''(\mu_i)}{g'(\mu_i)} \right) & \text{Newton–Raphson}
\end{cases}
$$

Initialise: $\mu = y, \eta = g(y)$.

- do:
  - Compute: $z = g'(\mu)(y - \mu) / \alpha(\mu) + \eta(X)$.
  - Compute: $w = \frac{\alpha(\mu)}{g'(\mu)^2 V(\mu)}$.
  - for $j = 1 : r$ (weighted backfitting algorithm)
    * $q_j = z - \beta_0 - \sum_{k \neq j} f_k(X_k)$ (obtain partial residuals)
    * fit $f_j(X_j) = S(q_j, X_j, w)$ (fits $X_j$ to $q_j$ with weights $w$ and some fitting method $S$)
  - Update $\mu = g^{-1}(\eta(X))$.

- Until convergence, i.e. parameter estimates no longer change or a likelihood convergence criterion is met.

Output: Fitted GAM prediction function $\eta$.

**Algorithm 2:** A local scoring algorithm for fitting GAMs.
An alternative approach that will be introduced in a later section is to use gradient boosting to learn the GAM prediction function. Boosting provides a number of advantages over the conventional method in Algorithm 2, particularly in high-dimensional settings, and although this method tends to be more computationally expensive, GAMs which are fit using boosting have been found to generally outperform GAMs fitted using conventional methods (see, Lou, Caruana, and Gehrke (2012)).

### 3.1.4 The Generalised Additive Model For Location, Scale and Shape

The GLM and GAM regression frameworks provide a simple way to obtain conditional estimates for the mean of the distribution of a univariate response variable, while considering the type of distribution of the data. These methods, however, focus on the expected value of a response variable, and assume its variance to be constant (or the variance to be a function of the conditional mean); they also require its distribution to be one within the exponential family of distributions, therefore ruling out important families such as the Student’s t and mixture distributions. For many modelling situations, however, it may also be the case that it is necessary to condition the scale parameter (variance) on exogenous variables, and the same can be said in the case of parametric distributions with more than two parameters, where the shape of the distribution may also be conditional on exogenous information. To address these modelling problems, there has been a significant amount of research on the importance of conditional estimates of other features—beyond the mean—of the response variable’s distribution, including conditional variance, skewness and kurtosis. Methods of this type have collectively become known as *distributional regression frameworks*.

An early approach to modelling conditional variance, focusing on the normal distribution, can be found in A. C. Harvey et al. (1976). This was extended to the GLM framework and consequently all exponential family distributions in Aitkin (1987), Nelder and Pregibon (1987), Smyth and Verbyla (1999) and Verbyla (1993). Within the GAM framework, Rigby and M. D. Stasinopoulos (1996) introduced the *mean and dispersion model* (MADAM). Thus presenting a model for both the conditional mean and dispersion parameters. The MADAM was to become the forerunner of the *generalised additive model of location, scale and shape* (GAMLSS).

The GAMLSS framework allows for modelling of all the parameters of a univariate response variable’s distribution as functions of explanatory variables. ‘location, scale and shape’ identifies the method as one that can be applied generally; that is, it is applicable to any parametric response distribution, continuous or discrete, and not restricted to membership of the exponential family. The method allows for conditional estimation of parameters that determine the
mean, variance, skewness and kurtosis of the response variable. As an example, for the special case where the response is normally distributed, only the parameters that specify the location and scale, i.e. the mean and standard deviation, would be considered. However, the same method can equally be applied to a five-parameter generalised skewed Student’s t distribution with location, scale and three shape parameters.

The original authors of GAMLSS, R. Rigby and D. Stasinopoulos (2001) and Rigby and D. M. Stasinopoulos (2005), and more recent contributors to the theory Mayr et al. (2012) and Hofner, Mayr, and Schmid (2014) state that the conditional density \( f(y|\Phi) \) may depend on up to four parameters \( \Phi = \{\mu, \sigma, \nu, \tau\} \); these parameters are commonly referred to as the location \( \mu \), scale \( \sigma \), and the two shape parameters \( \nu \) and \( \tau \). However, this parameter set is simply convention and there is no reason to restrict the GAMLSS to a maximum of four parameters; in fact, the framework can be used to model any type of parametric distribution. Nevertheless, for illustrative purposes, the above parameter set is sufficient to provide a detailed description of the framework as will be introduced next.

**DEFINITION OF THE GAMLSS FRAMEWORK**

Using the above-defined notation, and building on the theoretical background behind the GLM and GAM model classes, the GAMLSS model definition starts with the following sets of quantities,

\[
\Phi = \{\hat{\mu}, \hat{\sigma}, \hat{\nu}, \hat{\tau}\}, \\
\Xi = \{\eta_{\mu}, \eta_{\sigma}, \eta_{\nu}, \eta_{\tau}\}, \\
\mathcal{G}^{-1} = \{g_{\mu}^{-1}(\eta_{\mu}), g_{\sigma}^{-1}(\eta_{\sigma}), g_{\nu}^{-1}(\eta_{\nu}), g_{\tau}^{-1}(\eta_{\tau})\},
\]

(3.10)

where \( \Phi \) is the set of parameters of the response distribution, \( \Xi \) is the set of additive predictors associated to the parameters, and \( \mathcal{G}^{-1} \) is the set of inverse link functions. The elements of each set are related in the following way:

\[
\hat{\mu}_i = g_{\mu}^{-1}(\eta_{\mu}(x_i)), \\
\hat{\sigma}_i = g_{\sigma}^{-1}(\eta_{\sigma}(x_i)), \\
\hat{\nu}_i = g_{\nu}^{-1}(\eta_{\nu}(x_i)), \\
\hat{\tau}_i = g_{\tau}^{-1}(\eta_{\tau}(x_i)).
\]
Therefore, each distribution parameter $\theta_k \in \Phi$ is modelled with its own GAM with additive predictor function $\eta_{\theta_k}$, noting that for ease of notation $\theta_1 = \mu$, $\theta_2 = \sigma$, $\theta_3 = \nu$, $\theta_4 = \tau$. The GAMLSS model has the form of a set of equations:

$$g_{\theta_k}(\theta_k) = \beta_{0,\theta_k} + \sum_j f_{j,\theta_k}(x_{j,\theta_k})$$

$$= \eta_{\theta_k}, \text{for } k = 1, \ldots, 4.$$

Here $k$ indexes the parameter from the set $\Phi$. Note that the parameters $\theta_k$ can depend on possibly different sets of covariates. The $\beta_{0,\theta_k}$ are the intercept terms of the four sub-models. The $f_{j,\theta_k}$ for $j = 1, \ldots, J_{\theta_k}$ represent a function of covariate $j$.

---

**TRADITIONAL METHODS FOR ESTIMATING A GAMLSS**

This thesis will predominantly use a machine learning approach to the estimation of GAMLSS (Section 3.2.2); however, this builds on the earlier methodologies that will be described in this current section. To estimate the predictor functions, Rigby and D. M. Stasinopoulos (2005) introduced a penalised likelihood approach based on a modified version of the backfitting algorithm for conventional GAM estimation (Breiman and Friedman (1985)). They proposed two algorithms to estimate the parameters of a GAMLSS. Both these algorithms follow the same basic principle: in each iteration, backfitting steps are successively applied to the four distribution parameters, with the sub-model fits of previous iterations used as offset values for those parameters not involved in the current backfitting step. For more comprehensive details on the two algorithms, see a later paper by the same authors Stasinopoulos, Rigby, et al. (2007).
The model definition in Equation (3.11) allows for any predictor function of a single covariate to form part of the additive predictor. To illustrate the estimation of these models it is convenient to rewrite Equation (3.11) in terms of linear elements and smooth elements:

\[
Y \overset{\text{i.i.d.}}{\sim} D(\mu, \sigma, \nu, \tau),
\]

\[
\eta_\mu = g(\mu) = X_\mu \beta_\mu + \sum_{j=1}^{J_\mu} s_{\mu,j}(X_{\mu,j}),
\]

\[
\eta_\sigma = g(\sigma) = X_\sigma \beta_\sigma + \sum_{j=1}^{J_\sigma} s_{\sigma,j}(X_{\sigma,j}),
\]

\[
\eta_\nu = g(\nu) = X_\nu \beta_\nu + \sum_{j=1}^{J_\nu} s_{\nu,j}(X_{\nu,j}),
\]

\[
\eta_\tau = g(\tau) = X_\tau \beta_\tau + \sum_{j=1}^{J_\tau} s_{\tau,j}(X_{\tau,j}).
\]

Here \( D(\mu, \sigma, \nu, \tau) \) indicates the distribution of the response variable \( Y \), the \( X_\theta \) are design matrices for each sub-model \( \theta \in \{\mu, \sigma, \nu, \tau\} \), and \( \beta_\theta \) are the coefficient vectors for the linear part of the model. The functions \( s_{\theta,j} \) indicate smooth functions of each covariate, where \( J_\theta \) is the number of covariates used to estimate each sub-model.

Most of the smooth function types that can be used in this framework can be written as \( s(X) = Z\gamma \) where \( Z \) is another design matrix containing the basis of the smoother and dependent on \( X \). \( \gamma \) is another parameter vector, whose parameters are estimated with a quadratic penalty of the form \( \lambda \gamma^\top G \gamma \), where \( G = D^\top D \) and \( \lambda \) represents a hyperparameter determining the amount of regularisation (smoothing) to be applied. The form of the basis matrix \( Z \) and penalty \( D \) is determined by the type of smoother chosen. Functions such as \( s(X) \) are known as penalised smoothers.
The model can be written in the following form:

\[ Y \overset{\text{i.i.d.}}{\sim} D(\mu, \sigma, \nu, \tau), \]

\[ \eta_{\mu} = g(\mu) = X_{\mu} \beta_{\mu} + \sum_{j=1}^{J_{\mu}} Z_{\mu,j}(X_{\mu,j}) \gamma_{\mu,j}(\lambda_{\mu,j}). \]

\[ \eta_{\sigma} = g(\sigma) = X_{\sigma} \beta_{\sigma} + \sum_{j=1}^{J_{\sigma}} Z_{\sigma,j}(X_{\sigma,j}) \gamma_{\sigma,j}(\lambda_{\sigma,j}). \]

\[ \eta_{\nu} = g(\nu) = X_{\nu} \beta_{\nu} + \sum_{j=1}^{J_{\nu}} Z_{\nu,j}(X_{\nu,j}) \gamma_{\nu,j}(\lambda_{\nu,j}). \]

\[ \eta_{\tau} = g(\tau) = X_{\tau} \beta_{\tau} + \sum_{j=1}^{J_{\tau}} Z_{\tau,j}(X_{\tau,j}) \gamma_{\tau,j}(\lambda_{\tau,j}). \]

This formulation is known as the *random effects model*, or, more appropriately, the *semiparametric GAMLSS model* Rigby and D. M. Stasinopoulos (2005). In this model the \( \beta \)s are fixed effect parameters and the \( \gamma \)s are a set of smooth effect parameters. Therefore the parameters required to be estimated are

\[
\beta = (\beta_{\mu}, \beta_{\sigma}, \beta_{\nu}, \beta_{\tau})^T,
\]

\[
\gamma = (\gamma_{\mu,1}, \ldots, \gamma_{\mu,J_{\mu}}, \gamma_{\sigma,1}, \ldots, \gamma_{\sigma,J_{\sigma}}, \gamma_{\nu,1}, \ldots, \gamma_{\nu,J_{\nu}}, \gamma_{\tau,1}, \ldots, \gamma_{\tau,J_{\tau}})^T. \tag{3.12}
\]

The elements of \( \gamma \) are assumed to be independent of each other and to each have a prior distribution \( \gamma_{\theta,j} \sim \mathcal{N}(0, [G_{\theta,j}(\lambda_{\theta,j})]^{-1}) \). To estimate the \( \beta \)s, MLE estimation is sufficient; however, because of the penalty term \( \lambda \), the \( \gamma \)s require MAP estimation. Therefore the GAMLSS can be estimated by maximising the penalised log-likelihood.

First, the log-likelihood is

\[
\ell(\beta, \gamma) = \sum_{i} \log f_Y(y_i \mid \{x_{\theta,i}\}_{\theta \in \Phi_l}; \Phi_i)
\]

As the model contains smooth components,

\[
\ell(\beta, \gamma | Y, \lambda) = \ell(\beta, \gamma) + \sum_{\theta} \sum_{j} \log f(\gamma_{\theta,j} | \lambda) + c(Y, \lambda)
\]

\[
= \ell(\beta, \gamma | \lambda) + c(Y, \lambda).
\]
So with the normal prior distribution on the $\gamma$ s as defined above, the penalised log-likelihood is proportional to

$$\ell_P(\beta, \gamma | \lambda) \propto \ell(\beta, \gamma | Y, \lambda) - \frac{1}{2} \sum_{\theta \in \Phi} \sum_{j=1}^{J_\theta} \gamma_{\theta,j}^T G(\lambda_{\theta,j}) \gamma_{\theta,j} \tag{3.13}$$

As mentioned previously, to maximise Equation (3.13), Rigby and D. M. Stasinopoulos (2005) introduced two algorithms, the RS and the CG algorithm. In their basic form the algorithms maximise Equation (3.13) with respect to the sets of $\beta$ s and $\gamma$ s, Equation (3.12) keeping the regularisation parameter $\lambda$ fixed. Using the theory of GLMs and GAMs presented earlier, the RS algorithm is given below, as it forms the foundation for algorithms in later sections. For a description of the CG algorithm see Rigby and D. M. Stasinopoulos (2005):

The RS algorithm Rigby and D. M. Stasinopoulos (2005) has the following three nested components:

• the outer iteration, which cycles through $\Phi = \{\mu, \sigma, \nu, \tau\}$, calling,
• the inner iteration: the local scoring algorithm Algorithm 2, calling,
• a modified backfitting algorithm.

The RS algorithm cycles through the steps above until all three sub-algorithms have converged.

Let $m$ denote the current iteration; the outer iteration cycles through the set of distribution parameters $\Phi = \{\mu, \sigma, \nu, \tau\}$, updating them one at a time while holding the other parameters fixed at their most recent updated value. The outer cycle is illustrated in the following scheme:

$$(\hat{\mu}[m], \hat{\sigma}[m], \hat{\nu}[m], \hat{\tau}[m]) \quad \text{update} \quad \eta_{\mu}[m+1] \quad \rightarrow \quad \hat{\mu}[m+1]$$

$$(\hat{\mu}[m+1], \hat{\sigma}[m], \hat{\nu}[m], \hat{\tau}[m]) \quad \text{update} \quad \eta_{\sigma}[m+1] \quad \rightarrow \quad \hat{\sigma}[m+1]$$

$$(\hat{\mu}[m+1], \hat{\sigma}[m+1], \hat{\nu}[m], \hat{\tau}[m]) \quad \text{update} \quad \eta_{\nu}[m+1] \quad \rightarrow \quad \hat{\nu}[m+1]$$

$$(\hat{\mu}[m+1], \hat{\sigma}[m+1], \hat{\nu}[m+1], \hat{\tau}[m]) \quad \text{update} \quad \eta_{\tau}[m+1] \quad \rightarrow \quad \hat{\tau}[m+1] \tag{3.14}$$

The update above calls the inner iteration, which is a local scoring algorithm similar to the one used to fit generalised linear models, see (Algorithm 1).

For the current parameter to be updated, denoted $\theta$, the score function with respect to the predictor of $\theta$, $g(\theta) = \eta_{\theta}$ is

$$\frac{\partial \ell}{\partial \eta_{\theta}} = \frac{\partial \ell}{\partial \theta} \circ \frac{\partial \theta}{\partial \eta_{\theta}} \tag{3.15}$$
where $\circ$ is used to note elementwise multiplication and $\frac{\partial \ell}{\partial \eta_0}$ is a vector of gradients evaluated at each data point. The RS algorithm does not compute cross-derivatives of the information matrix; instead all second order derivatives are with respect to the same variable. The information matrix can be computed in three different ways, with the observed Hessian, expected Hessian (Fisher Scoring) or a quasi-Newton method:

$$H_\theta = \begin{cases} 
E \left[ \frac{\partial^2 \ell}{\partial \eta^2} \right] & \text{FisherScoring} \\
\frac{\partial^2 \ell}{\partial \eta^2} & \text{Newton–Raphson} \\
- \left( \frac{\partial \ell}{\partial \eta} \right)^2 & \text{QuasiNewton–Raphson}.
\end{cases}$$

The local scoring algorithm consists of repeated weighted fits to a modified dependent variable, using modified weights, which are repeated until convergence of the log-likelihood. The weights are computed as

$$w_\theta = -H_\theta \circ \frac{\partial \theta}{\partial \eta_0} \circ \frac{\partial \theta}{\partial \eta_0}$$

and the modified dependent variable for fitting parameter $\theta$ is given by

$$z_\theta = \eta_0(X_\theta) + w^{-1} \circ \frac{\partial \ell}{\partial \eta_0}.$$  

Inputs: $z_\theta$, from Equation (3.17); $w_\theta$ from Equation (3.16); $X_\theta$, $\{Z_{\theta,j}\}_{j=1}^{J_\theta}$

- do:
  - Compute: $q \leftarrow (z_\theta - \sum_{j=1}^{J_\theta} Z_{\theta,j} \hat{\gamma}_{\theta,j}).$
  - Estimate $\hat{\beta}$s by regressing $q$ on $X_\theta$ using weighted least squares with weights $w_\theta$.
  - for $j = 1 : J_\theta$
    - Compute: $q \leftarrow (z_\theta - X_\theta \hat{\beta}_\theta - \sum_{i \neq j} Z_{\theta,i} \hat{\gamma}_{\theta,i}).$
    - Estimate new $\hat{\gamma}_j$ by regressing $Z_{\theta,j}$ on $q$ with weights $w_\theta$, using penalised weighted least squares.
- until convergence, i.e. parameter estimates no longer change.

Outputs: $\hat{\beta}_\theta$, $\{\hat{\gamma}_{\theta,j}\}_{j=1}^{J_\theta}$ as defined in Equation (3.12)

**Algorithm 3:** Modified backfitting algorithm

The final sub-algorithm called by an iteration of the RS algorithm is responsible for estimation of the $\beta_\theta$ and $\gamma_\theta$ parameters Equation (3.12). This is a modified backfitting algorithm, a version of the Gauss–Seidel algorithm Hastie and Tibshirani (1990). The backfitting algorithm works by
fitting linear explanatory variables and smoothers to a vector of modified dependent variables $z_\theta$ with weights $w_\theta$, which have been obtained from the inner cycle of the RS algorithm, as shown in Equation (3.17) and Equation (3.16) above, using the local scoring procedure. Algorithm Algorithm 3 describes the steps involved.

The algorithm fits each covariate in a step-wise iterative fashion. In situations where there are a large number of covariates, especially if many of these covariates have a particularly weak relationship with the response parameter, forcing them to be included in the model, as is done here, is not ideal. With a low signal, high noise modelling environment, such as with financial time-series data, a different approach is preferable. This approach takes as given that all covariates are only very weakly, if at all, correlated with the response variable, and uses this fact to build a much stronger model that is more robust to noise and, therefore, overfitting. This approach is called boosting and is described in the following sections.

3.2 GRADIENT BOOSTING

Boosting is the name given to a general class of machine learning meta-algorithms categorised by the particular way in which they learn from data. The idea behind boosting is for the algorithm to build a 'strong' predictive model from an ensemble, or combination, of many weaker models. The weaker models are building blocks that form the ensemble for the strong boosted predictor, and are commonly known as base learners. The characteristics that qualify a base learner as 'weak' can be easily explained in the context of a classification problem, where a weak learner would be one that, on unseen data, would perform only slightly better than a random guess. The boosting algorithm works by combining a set of these base learners together in such a way so that the combined performance of the ensemble is greater than that of any individual base learner. In practice, however, the overall predictor will tend to substantially out-perform the best base learner.

The key aspect to boosting algorithms is that at each round, a new weak (base) learner is combined with the base learners that were added in previous rounds. The new ensemble component is taught to correct mistakes of the previous rounds or trained so that its addition to the ensemble makes a slight improvement to the value of the loss function. Ensemble methods such as these have many practical benefits over many other classes of model, primarily in the bias-variance trade-off. For this class of model, variance can be substantially reduced at the
cost of only a slight increase in bias; as a result predictive models built with boosting have been shown to be more robust to overfitting despite their complexity.

The idea of boosting was first suggested by Kearns (1988) and Kearns and Valiant (1994), who questioned whether a high bias 'weak' model, that can only perform slightly better than a random guess, can be boosted into an arbitrarily accurate predictive model. This was followed by Schapire (1990) and Freund (1995) and Freund (2001) who developed ADABoost, the first well-known boosting algorithms. Further experiments from Freund, Schapire, et al. (1996) and subsequent work Freund et al. (2003), Freund and Mason (1999), Freund and Schapire (1997), and Freund and Schapire (1996) led to the emergence of boosting as an effective new machine learning methodology.

The original boosting algorithms worked by iteratively reweighting the observations after each boosting round, based on a performance metric obtained at the end of the previous round. This metric could be a loss function, such as the classification margin or squared error, for regression. In such a case, after a boosting round, each data point is evaluated with respect to the loss function, the observations with greater loss then being given more weight in the next round, so that the efforts of the next base learner to be added will be concentrated on getting these observations correct.

Classification problems with binary outcomes are best suited to illustrate the workings of a boosting algorithm. AdaBoost (Freund (1995) and Freund (2001)) is arguably one of the best known boosting algorithms and is presented below in Algorithm 4. In each boosting round of the AdaBoost algorithm there is a step-wise addition of a base learner to the prediction function, as well as an optimally assigned weight for the contribution of the new base learner. This results in a weighted majority predictor of the form

\[ \hat{y} = \text{sign} \left( \sum_{m=1}^{M} \alpha^*[m] h^*[m](x) \right). \]

For a set of training examples \( \{ (x_1, y_1), \ldots, (x_n, y_n) \} \), where \( x \in \mathbb{R}^d \), \( y \in \{-1,1\} \), and a vector of initial observation weights, at each iteration of the training process, the algorithm proceeds by fitting a weak model to the data using weights \( W^*[m] \). This gives a hypothesis \( h^*[m] \) that takes
in an object $x$ and outputs a binary prediction $\hat{y} \in \{-1, 1\}$ for each observation. This is then added to the additive predictor function from the previous round to get

$$\eta^{[m]}(x) = \eta^{[m-1]}(x) + \alpha^{[m]} h^{[m]}(x)$$

$$= \sum_{l=1}^{m} \alpha^{[l]} h^{[l]}(x),$$

which is a weighted majority prediction over the weak hypotheses generated by the base learners. Simple decision stumps are often used alongside the AdaBoost algorithm; however, any type of classifier can be used to generate the weak hypotheses $h$.

The weight $\alpha^{[m]}$ assigned to the output of base learner $h^{[m]}$ is computed by minimising the following loss function,

$$L(y, x) = \sum_{i=1}^{n} W^{[m]}_i \exp \left( -\alpha^{[m]} h^{[m]}(x_i)y_i \right),$$

$$\alpha^{[m]} := \operatorname{argmin}_{\alpha^{[m]}} \sum_{i=1}^{n} W^{[m]}_i \exp \left( -\alpha^{[m]} h^{[m]}(x_i)y_i \right),$$

which has the solution

$$\alpha^{[m]} = \frac{1}{2} \log \left( \frac{\sum_{y = h^{[m]}} W^{[m]}_i}{\sum_{y \neq h^{[m]}} W^{[m]}_i} \right).$$

The initial values for the observation weights are $W^{[1]}_i = 1 \forall i = 1:n$, and the weights for the subsequent boosting rounds are computed by $W^{[m]}_i = \exp \left( -\eta^{[m-1]}(x_i) \right)$, thus giving more weight to the observations that have, in previous rounds, been more difficult to classify. The full AdaBoost algorithm is presented in Algorithm 4.

In the next subsection the work of Breiman et al. (1998), Breiman (1999) Friedman (2001) is outlined; these authors discovered that boosting could be interpreted as a functional gradient descent algorithm, and Friedman, Hastie, Tibshirani, et al. (2000) and Bühlmann and Yu (2003) further showed a direct link to forward stage-wise additive modelling. From these findings boosting, therefore, becomes a natural and theoretically valid method for constructing GAMs.
Input: dataset \( \{(x_1, y_1), \ldots, (x_n, y_n)\} \), where \( x \in \mathbb{R}^d \), \( y = \{-1, 1\} \), \( M \) is the number of boosting rounds.

Initialize: \( W_i^{[1]} = 1/n \) for \( i = 1 : n \).

- For \( m = 1 : M \): (boosting round)
  - Train a weak learner using weighted observations \( W^{[m]} \).
  - Get weak hypothesis \( h^{[m]} : \mathbb{R}^d \rightarrow \{-1, +1\} \).
  - Given \( W^{[m]} \) and \( h^{[m]} \) compute the weighted classification error
    \[
    \epsilon^{[m]} = P \left( h^{[m]}(x_i) \neq y_i \right).
    \]
  - Compute \( \alpha^{[m]} = \frac{1}{2} \log \left( \frac{1-\epsilon^{[m]}}{\epsilon^{[m]}} \right) \) and update weights
    \[
    W_i^{[m+1]} = \frac{W_i^{[m]}}{Z} \exp \left( -\alpha^{[m]} h^{[m]}(x_i) y_i \right),
    \]
    \[
    Z = \sum_{i=1}^{n} W_i^{[m]} \exp \left( -\alpha^{[m]} h^{[m]}(x_i) y_i \right).
    \]

- Output: The final predictor function \( \eta(x) \) as the weighted sum of all base learner predictions
  \[
  \eta(x) = \text{sign} \left( \sum_{m=1}^{M} \alpha^{[m]} h^{[m]}(x) \right).
  \]

**Algorithm 4: AdaBoost**
3.2.1 Boosting as Functional Gradient Descent

Following boosting’s early advances established predominantly by AdaBoost and its variations, Breiman et al. (1998), Breiman (1999), Friedman (2001) showed that boosting can be interpreted as gradient descent in function space, which today we know as gradient boosting. Just like the modus operandi of the AdaBoost algorithm, gradient boosting learns an ensemble of base models, in a sequentially additive manner. The set of base learners comprising the additive predictor function is typically large, and, as a result, the influence of any individual base learner on the overall prediction is marginal. The major difference between AdaBoost-type algorithms and those that use gradient boosting is the way the two methods determine what the new to be added base learners should be trained to fit to. As shown in Algorithm 4 AdaBoost seeks to improve on overall performance by manipulating the observation weights, giving hard to predict observations more weight when new learners are fitted; gradient boosting improves on performance by fitting base models to the gradient of the loss function; therefore, focusing more on fitting to the observations with larger evaluated gradients, and so are further away from the current estimated values.

A major drawback of AdaBoost, and other early algorithms such as LogitBoost, is that the type of loss function is fixed. The contrasting flexibility of gradient boosting is a major advantage as it allows one to optimise an arbitrary loss function, making gradient boosting methods more suited to a wide range of real world applications. In a general gradient boosting framework, the aim is to find an optimal prediction function \( \eta \) defined as

\[
\eta^* := \arg \min_{\eta} \mathbb{E} [\rho(Y, \eta(X))],
\]

where \( Y \) and \( X \) are the random variables for response and covariates respectively, \( \rho \) is a loss function that is required to be differentiable with respect to \( \eta \). The data set \( \{(y_i, x_i)\}_{i=1}^n \) are the real valued univariate response variables and the vectors of covariates for each observation \( i \). The expectation \( \mathbb{E} [\cdot] \) can be approximated by the empirical loss, defined as

\[
\mathbb{E} [\rho(Y, \eta(X))] \approx \frac{1}{n} \sum_{i=1}^n \rho(y_i, \eta(x_i)). \quad (3.18)
\]

Algorithm 5 presents a gradient boosting procedure for an arbitrary loss function.

Typically the class of base learner (we denote as \( \mathcal{H} \)) used in regression or classification problems are classification and regression trees (CART); however, any supervised learning model can—in theory—be used. The resulting learned predictor function is an ensemble containing many of
3.2 GRADIENT BOOSTING

Input: loss function $\rho$, training set $\{(y_i, x_i)\}_{i=1}^n$, max iterations $M$, learning rate $0 < \delta \leq 1$, base model class $\mathcal{H}$
initialize: $\eta[0] = \frac{1}{n} \sum_{i=1}^n y_i$ (or any choice of offset value)

- For $m = 1 : M$ (Boosting rounds)
  
  - Compute the negative gradient and evaluate at the current estimate:
    \[
    \nabla_i = - \left. \frac{\partial \rho(y_i, \eta_i)}{\partial \eta} \right|_{\eta_i = \eta[m-1](x_i)}
    \]

  - Fit a base learner $h \in \mathcal{H}$, from any class of model $\mathcal{H}$ to the negative gradient. For example, using the quadratic loss,
    \[
    h^* := \arg\min_h \sum_{i=1}^n (\nabla_i - h(x_i))^2
    \]

  - Add base learner to the ensemble:
    \[
    \eta[m](x) = \eta[m](x) + \delta h^*(x)
    \]

Output: $\eta[M]$

Algorithm 5: A general gradient boosting algorithm

these base models, combined additively and weighted by a shrinkage factor $\delta$, also commonly referred to as the learning rate. The learning rate is a user specified parameter set to be between 0 and 1 and usually is chosen to control the amount of computation time taken to build the model. In fact, along with the number of boosting rounds (denoted as $M$) it also is used to control the overall complexity of the model. As a means to avoid overfitting in training, only one of these two hyperparameters needs to be selected with the other fixed ex ante; usually it is the learning rate $\delta$ that is set at a low value, $0 < \delta < 0.3$, leaving the number of boosting rounds to be chosen by cross-validation.

3.2.2 Component-wise Boosting

When the covariate vector is multi-dimensional, $X \in \mathbb{R}^{n \times d}$, a gradient boosting algorithm can be adapted to fit the set of covariates component-wise. As an example, given a class of base learner such as decision stumps, linear models or smoothing splines, then for each exogenous variable $p \in 1 : d$, $d$ component models are each specified as a function of a single variable $X_p$ (with the function type determined by base learner class), and all are independently fitted to the negative gradient; then at each boosting round, the best fitting of these component models is added to the predictor function.
The introduction of component-wise boosting in Bühlmann and Yu (2003) showed that the method results in an additive model of a similar design to the models described in Section 3.1.3; however, when building additive models with component-wise gradient boosting the number of models included in the additive predictor is dictated by the number of boosting rounds and not, importantly, the size of the input space. As a result a single covariate may be represented multiple times in the set of selected base-learners and also some covariates may be excluded from the model entirely. Therefore, component-wise boosting has a natural form of feature selection built into the learning procedure; this has been shown to be particularly effective in situations with high dimensional data especially when the input space is larger than the number of observations (see, Bühlmann and Yu (2003), Mayr et al. (2010)). Another advantage of this method is that, any prior information about the type of effect a covariate has on the response can be encoded in the choice of the class of base learner associated to that covariate. Hofner et al. (2014) illustrate the variety of different types of base learners that can be combined in a single additive model and the type of effects these base learners can capture; these include linear effects, smooth effects, random effects, spacial effects, interaction terms and many more.

When pre-specifying a model, there is no restriction to only assign one class of base learner to a particular covariate, i.e. one can specify multiple candidate models for a single covariate but with different types of effect (base learner class); as an example, a linear model and a P-spline as functions of the same covariate can both be pre-specified as candidate components and therefore considered in the selection process. The method offers even greater flexibility, as components do not necessarily need to be functions of a single covariate; when a component is thought of as a pre-specified pair, that is a base learner class and covariate, it can easily be extended to the case of a base learner class and a subset of covariates. This specification is inclusive of base learners that are functionals of more than one covariate such as regression trees or bivariate smoothing splines. In the former case, a regression tree may be specified with all covariates in the input space. During the fitting procedure, this class of base model has its own methods for selecting the most relevant combination of variables to include, and the outcome of such a specification is a gradient boosting regression tree (GBRT), just one of many special cases of models that can be learned with this method.

For a simple problem, to estimate the conditional mean of a univariate response variable $Y$, with a set of covariates $X$ using some prediction function $\eta : \mathbb{R}^d \to \mathbb{R}$,

$$
E[Y|X=x] = \eta(x),
$$
where $x \in \mathbb{R}^d$, the prediction function is structured in the following, general, way

$$\eta(x) = \beta_0 + \sum_{m=1}^{M} f[m](x), \quad (3.19)$$

where $\eta[0](x) = \beta_0$ is the initial (offset) value of the estimate; this can be set to the maximum likelihood estimate of $Y$ in this case, the unconditional mean of the response. $M$ is the total number of boosting rounds and $f[m]$ is the component model added during round $m$. The structure of the predictor is restricted to the additive form in (Equation (3.19)); however, this structure is extremely flexible with arbitrary forms for the functions $f$. This construction has many well known special cases such as a linear model $\eta(x) = \beta_0 + \sum_{m=1}^{d} x_m \beta_m$ or an additive model $\eta(x) = \beta_0 + \sum_{m=1}^{d} f[m](x_m)$ where each $f_m$ is a function of the $m$th covariate $x_m$ (note that, for these examples, the learning rate $\delta = 1$).

Component-wise gradient boosting is a natural method to construct GAMs (Section 3.1.3), yet it offers extra flexibility inasmuch as many more components than covariates can be included in the model, along with the added benefit of the model selection procedure, which disregards features that are deemed to be uninformative.

The gradient boosting algorithm minimises

$$E_{XY} [\rho(Y, \eta(X))] \approx \frac{1}{n} \sum_{i=1}^{n} \rho(y_i, \eta(x_i)),$$

with the loss function $\rho$ as the negative conditional log likelihood function

$$\rho(y_i, \eta(X_i)) = -\log f_Y(y_i|\theta).$$

This can be achieved by a step-wise descent of the gradient. As this is an iterative procedure we use $0 \leq m \leq M$ to denote the current boosting round and $\hat{\eta}[m]_i(x_i)$ the current estimate of data point $y_i$, given covariates $x_i$.

Before the algorithm can be used to learn from data, a set of base learners must be specified; we denote this set $H$ with cardinality $J$. The types of base-learners in $H$ can vary depending on the assumed relationship, or effect, individual covariates have on the response variable. Typical types of base learners that can be used include linear models, penalised p-splines for smooth effects and regression trees to capture non-linear interactions between sets of covariates.
An example would be to specify the components as linear functions of each covariate in the input space, so the set of base learners becomes $H = \{ h(x_1), h_2(x_2), ..., h_d(x_d) \}$, where each $h_j(x_j) = x_j \beta_j$ is a linear model. The flexibility of this setup means one can specify more base learners than there are covariates; as an example, let $S = s_1 \{(x_1), s_2(x_2), ..., s_d(x_d)\}$ be the set of components, where $s_j(x_j)$ is a smooth function, perhaps a p-spline (see, Eilers and Marx (1996)); the set of pre-specified components then becomes $H \cup S$. Additionally one may wish to include a regression tree $t(x)$, which is a function of all covariates, and then the component set becomes $H \cup S \cup \{ t(x) \}$. If the type of effect of a covariate is unknown then different base learners can be specified for a particular covariate and the boosting procedure will select the one it deems more relevant, or perhaps select a different type in a subsequent boosting round.

In the following illustration only the set $H$ is considered, but the procedure is easily extendable to any number of different specifications.

The steps of the algorithm are as follows:

1. At $m = 0$, set an offset value for $\eta^0$, i.e the initial estimate of $\theta$. This can be $\eta^{[0]} = 0$ or usually the sample mean $\eta^{[0]} = \frac{1}{n} \sum_{i=1}^{n} y_i$.

2. Increment $m$.

3. Take derivatives of the loss function and evaluate the negative gradient for all $y_i$ given the most recent estimates of the conditional mean $\hat{\theta}_i = \hat{\eta}^{[m-1]}(x_i)$ to obtain a vector of negative gradients

   $$\nabla \eta_i = - \frac{\partial \rho (y_i, \eta_i)}{\partial \eta} \bigg|_{\eta_i = \hat{\eta}^{[m-1]}(x_i)}. \quad (3.20)$$

4. Fit each of the base learners in $H$ to the negative gradient vector to get a vector of $J$ regression estimators. Typically the fit will be in the form of least squares but any loss function is possible. For $j = \{1, ..., J\}$ then for each $h_j \in H$,

   $$h_j^* := \arg \min_{h_j} \left[ \sum_{i=1}^{n} (\nabla \eta_i - h_j(x_{i,j}))^2 \right], \quad (3.21)$$

   $$H^* = \{ h_1^*(x_1), ..., h_J^*(x_d) \}.$$

5. Select from the set $H^*$ the best fitting base learner denoted as $h^{[m]}$ and update the predictor function,

   $$\eta^{[m]} = \eta^{[m-1]} + \delta h^{[m]}.$$

6. Repeat steps 2 to 5 until $m = M$.

Note that the notation $\eta^{[m]}$ and $\eta^{[m]}(x)$ is used interchangeably and represents the same object.
In step 5 the parameter $0 < \delta < 1$ is a learning parameter that specifies the step length; this is also known as the learning rate. A small learning rate results in larger models that are less prone to overfitting; however, the algorithm takes longer and requires more iterations to effectively learn from the data. If $\delta$ is too high, the algorithm may not converge, and if it is too low, the computation time may be unwelcome. The above procedure shows that at each iteration a proportion $\delta$ of an estimate of the negative gradient is added to $\eta$. As a result, after each round the boosting algorithm takes a small step descending the gradient of the empirical loss function $\mathcal{L}$ (Equation (3.18)), $\mathcal{L}$ is thus gradually minimised, and a functional relationship between $y$ and $x$ will be discovered.

The piecemeal nature of boosting makes it a powerful machine learning method. Inherently, using the contribution of many weak models means that the effect of each individual model is shrunk towards zero as more models are added to the predictor. This has two major learning advantages. First and foremost, shrinking the size of the contribution of each base learner means the model has a natural form of regularisation; thus, boosting models tend to have lower variance of estimates and greater prediction accuracy. The second advantage of using boosting can be seen in step 5 above: at each round the best base learner covariate subset pair $(h_j^*, x_j)$ is selected from a set of candidate models; having only a single feature, or perhaps a subset of features, included in each round means, as a consequence of the boosting procedure, automatic feature selection occurs as part of the learning process; as such, this method is well suited for high dimensional data.

There are clear benefits of using this method for learning additive models. Iteratively applying the steps above gives the predictor function $\eta$ an additive structure in the form similar to that of a GAM Equation (3.9). The next section shows how the above method can be extended to learning full conditional probability distributions.

### 3.2.3 Gradient Boosting for GAMLSS

In this subsection, the gamBoostLSS algorithm of Mayr et al. (2012) is presented in its most general form, and notation and model description for convenience of reference follows that of the original authors. In Rigby and D. M. Stasinopoulos (2005), which introduced GAMLSS models, the conditional response distribution is described as consisting of up to $P = 4$ conditional parameters, that is

$$f_Y(y|\Phi) = f_Y(y|g^{-1}_\mu(\eta_\mu(X_\mu)), g^{-1}_\sigma(\eta_\sigma(X_\sigma)), g^{-1}_\nu(\eta_\nu(X_\nu)), g^{-1}_\tau(\eta_\tau(X_\tau)))$$
3.2 GRADIENT BOOSTING

However, it is important to note that this four-parameter restriction, as mentioned in Mayr et al. (2012), is not in fact necessary, and therefore there is no actual limit to the number of parameters that can be considered by the method; despite this, as in Mayr et al. (2012), we follow notation and convention established by the original authors and limit this illustration to the \( P = 4 \) parameter case. As in (Section 3.1.4) the following sets are defined:

\[
\Phi = \{ \hat{\mu}, \hat{\sigma}, \hat{\nu}, \hat{\tau} \},
\Xi = \{ \eta_\mu, \eta_\sigma, \eta_\nu, \eta_\tau \},
\]

\[
G^{-1} = \{ g_\mu^{-1}(\eta_\mu), g_\sigma^{-1}(\eta_\sigma), g_\nu^{-1}(\eta_\nu), g_\tau^{-1}(\eta_\tau) \} .
\] (3.22)

A gradient boosting GAMLSS is defined by the set of \( P = 4 \) equations

\[
g_{\theta_k}(\theta_k) = \eta_{\theta_k}(X_{\theta_k}) = \eta_{\theta_k}^{[0]} + \sum_{m=1}^{M} \delta h_{\theta_k}^{[m]}(X_{\theta_k}),
\forall \theta_k \in \Phi , \text{ for } k = 1 : P.
\]

Here \( \eta_{\theta_k}^{[0]} \) is the initial offset value or intercept for parameter \( \theta_k \), \( \delta \) is a shrinkage factor, \( h_{\theta_k}^{[m]}(X_{\theta_k}) \) is the base-learner / covariate component selected at round \( m \) for parameter \( \theta_k \). For notational brevity, the elements of the set \( \Phi \) are indexed by \( \theta_k \) for \( k = 1 : P \) and \( h_{\theta_k}^{[m]} \) is implied to include subsetting of the input space such that the component chosen during round \( m \) is a functional of only the subset of covariates that were pre-specified for that particular component in \( H \). As before, \( X_{\theta_k} \) identifies that there may be different sets of exogenous variables used to estimate each of the parameters in \( \Phi \).

To learn a GAMLSS, the loss function we want to minimise is the negative conditional log likelihood,

\[
L(Y, X) = - \sum_{t=1}^{T} \log f_Y(y_t|\mu_t, \sigma_t, \nu_t, \tau_t)
\] (3.23)

\[
= - \sum_{t=1}^{T} \log f_Y(y_t|g_\mu^{-1}(\eta_\mu(X_{\mu,t})), g_\sigma^{-1}(\eta_\sigma(X_{\sigma,t})), g_\nu^{-1}(\eta_\nu(X_{\nu,t})), g_\tau^{-1}(\eta_\tau(X_{\tau,t})))
\] (3.24)

\[
= \sum_{t=1}^{T} \rho(y_t, \Xi),
\] (3.25)

where, in the final line above, the loss function is written as a function of the set of predictors \( \Xi \), as defined in (Equation (3.22)), \( g_{\theta_k}^{-1} \) is the inverse monotonic link function mapping the output from a predictor to the parameter, i.e., \( g_{\theta_k}^{-1} : \eta_{\theta_k} \mapsto \theta_k \), see (Section 3.1.2) for more details.
The loss function is specified by $P$ parameter estimates and so an updating scheme with this in mind is required. In Mayr et al. (2010) a parameter updating strategy is used which has been accredited to Schmid et al. (2010) and already seen in the outer cycle of the RS Algorithm (Section 3.1.4.2). This strategy is a cyclical updating scheme, where at each boosting round the algorithm cycles through the parameter set, updating the predictor function for each parameter in turn while holding the other parameters at their most up-to-date estimates. The scheme is as follows:

$$
\Phi^{[m+1]}(\theta) = (\hat{\mu}^{[m]}, \hat{\sigma}^{[m]}, \hat{\nu}^{[m]}, \hat{\tau}^{[m]})
$$

In general the quantity $\Phi^{[m+1]}(\theta)$ is the current set of parameter estimates, where

$$
\hat{\theta}_k = \begin{cases} 
\hat{\theta}_k^{[m+1]} & \text{if } l < k \\
\hat{\theta}_k^{[m]} & \text{otherwise}.
\end{cases}
$$

That is, notwithstanding the current position in the cycle, $\Phi^{[m+1]}(\theta)$ returns the set with the most up-to-date parameter estimates for when $\theta$ is the current parameter to be updated. This definition is extended to the set of predictor functions $\Xi$, where $\Xi^{[m+1]}(\theta)$ is the set of the most up to date predictors at the point where $\theta$ is being updated, indexing a row of (Equation (3.26)).

Due to the marginal changes that occur during a single boosting round, the order in which the parameters are updated has not been thought to be of importance; however, during experiments in later chapters, certain permutations of the parameter set were shown to be more likely to lead to stability issues. Therefore, if issues are encountered it may prove beneficial to change the order in which the parameters are updated to see if the problem is resolved.

An alternative update scheme can be found in [Adam, Mayr, and Kneib (2017)]; this is known as inline updating and differs from the cyclical scheme by only updating one of the parameters per round. Inline updating works by adding a base-learner to each of the predictor functions based on gradients evaluated using estimates from the previous round. The scheme then proceeds to select only the parameter whose update reduces the loss the most and then adds that base learner to the model. The inline updating scheme has both advantages and disadvantages over the cyclical scheme; the major advantage is becomes obvious in model selection. Rather than selecting a number of boosting rounds for each parameter, when using the inline scheme only
one hyperparameter requires selecting. Disadvantages include a large amount of redundant computation and the algorithm’s tendency to get stuck, only updating a subset of the model’s parameters.

The optimisation problem solved by using the gamboostLSS algorithm can be written as

$$\hat{\mu}, \hat{\sigma}, \hat{\nu}, \hat{\tau} := \arg\min_{\eta_{\mu}, \eta_{\sigma}, \eta_{\nu}, \eta_{\tau}} \mathcal{L}(Y, X),$$

where the empirical loss $\mathcal{L}$ is defined in (Equation (3.23)). The gradient boosting approach for learning GAMLSS models is presented in (Algorithm 6). The algorithm requires a set $H$ of pre-specified base-learner / covariate subset components as described in (Section 3.2.2), and a parametric distribution with probability density $f_Y$. As in the section on GAMLSS, different sets of covariates can be included for the different sub-models therefore the conditioning set can be denoted as $\{X_{\theta_k}\}_{k=1}^P$.

The offset values can be any initial guess; some suggested values include link function transformations of maximum likelihood estimates of the parameters; setting the $\eta_{\mu}^{[0]} = 0$ is also an option. For the conditional mean estimator $\eta_{\mu}^{[0]}$ the offset may even be vector-valued and set to the actual value of the response $\eta_{\mu,t}^{[0]} = y_t$.

3.3 MARKOV-SWITCHING REGRESSION MODELS

3.3.1 Overview of Regime-switching

At different points in time, financial and economic time-series data can often display very different types of behaviour. It is well known, however, that behaviour of a particular type can endure for a period of time only to be then replaced by another distinct type. For the duration of time where the time-series is behaving in a self-similar manner, we can consider the series as operating within a particular regime. By analogy, in politics the term regime can be used to describe a form of government with a distinct set of rules and laws. Similarly, a regime in time-series analysis can be thought of as distinct rules and laws that govern the data-generating process of the observable time-series.
Inputs: response vector and sets of covariates \( Y, \{ X_{\theta_k} \}_{k=1}^P \), pre-specified base-learners / covariate subset components \( \{ H_{\theta_k} \}_{k=1}^P \), response density function \( f_Y \) and corresponding loss function \( \rho \), shrinkage factor \( 0 < \epsilon \leq 1 \), number of boosting rounds \( M \).

Initialise: Initialise the additive predictors \( \Xi[0] = \{ \eta^{[0]}_{\mu}, \eta^{[0]}_{\nu}, \eta^{[0]}_{\tau} \} \) with offset values, \( \eta^{[0]}_{\theta_k} = 0 \) or \( \eta^{[0]}_{\theta_k} = g_{\theta_k}(\hat{\theta}_{k\text{MLE}}) \) for \( k = 1, \ldots, P \).

* For \( m = 0 : M \)
  
  - For \( p = 1 : P \) (cycle through parameter set)
    
    * Compute negative gradient vector, evaluated with the most up-to-date set of predictor functions,
      \[
      \nabla_{\theta_p}^{[m+1]} = - \frac{\partial \rho(y, \Xi(\theta_p))}{\partial \eta_{\theta_p}} = \begin{cases} 
      \Xi^{[m]}(\theta_u) & \text{if } u \geq p \\
      \Xi^{[m+1]}(\theta_u) & \text{if } u < p 
      \end{cases}
      \]

    * Fit each of the base-learners in \( H_{\theta_k} \) to the negative gradient vector to get a vector of \( J \) regression estimators. Typically the fit is in the least squares sense, but any regression loss function is possible. For \( j = \{1, \ldots, J\} \) and for each \( h_j \in H_{\theta_k} \),
      \[
      h_j^* := \arg \min_{h_j} \left[ \sum_{t=1}^{T} \left( \nabla_{\eta_{\theta_k},t} - h_j(X_{\theta_k},t) \right)^2 \right],
      \]
      \[
      H_{\theta_k}^* = \{ h_1^*, h_2^*, \ldots, h_J^* \}. \tag{3.27}
      \]

    * Select from \( H_{\theta_k}^* \) the \( h_j^* \) which had the smallest error in (Equation (3.27)) designated \( h^{**} \).

    * Update predictor with new component \( h^{[m+1]} = h^{**} \)
      \[
      \eta^{[m+1]}_{\theta_k} = \eta^{[m]}_{\theta_k} + \epsilon h^{[m+1]}(X_{\theta_k}), \\
      \hat{\theta}_k = g_{\theta}^{-1}(\eta^{[m]}_{\theta_k}(X_{\theta_k})).
      \]

  - end parameter cycle

* end boosting round

Output: set of prediction functions \( \Xi^{[M]} \) for parameters of \( f_Y \).

**Algorithm 6:** Gradient boosting algorithm for GAMLSS (gamBoostLSS) using the cyclical update scheme.
For financial and economic time-series in particular, which are well known to be susceptible to changing regimes, it is useful to have a model that allows for changes in its parameters to account for the change in behaviour between the different regimes. A failure to account for this type of time-variation will mean that inevitably, at some point in time when the data generating process changes, this change is highly likely going to render a once-working model uninformative. Models that are adaptable to these changes by being able to infer the underlying data generating process from the observable data are known as regime-switching models and have been shown to be particularly useful for modelling of financial and economic time-series data.

There can be many different reasons for one regime to change to another. Obvious factors may include changes in the macroeconomic or political environments, such as a change in monetary policy; a recession transitioning to a recovery; outbreak of war; and even government response to a global pandemic. However, most events that can cause time-series behaviour to change are much more subtle than these examples and it is not immediately obvious when they have occurred. For this reason, it is helpful to have a probabilistic representation for the entire data generating process, one that includes the observable time-series as well as a probability law for the dynamics of regime change. This final section of the technical background provides a detailed background on Markov-switching (MS) regression models, a well known framework for modelling time-series with regime changes when the source of the change is not observable. The basis of this framework is to jointly model the observable time-series along with the unobservable process that represents the changes in regime over time. This presentation will begin below with an introduction to regime switching, before moving on in the following subsection to introduce the backbone architecture of latent state-space models by discussing the inference and estimation methods of the hidden Markov model (HMM). This is followed by linking the Markov-switching dynamics of the HMM to the mechanism that governs the switching of regimes in RSMs, introducing a Markov-switching regression model, and, finally, this section concludes with a description of the extension of MS regression models to the case of distributional regression, substituting the unconditional state-dependent distributions of the HMM with fully conditional GAMLSS models estimated by using the gradient boosting methods introduced in Section 3.2.3.

To begin, some necessary concepts first need to be established. The observed process is defined as $Y = \{Y_t\}_{t=1}^T$, and has a realisation $y_{1:T} = (y_1, y_2, ..., y_T)$. This process is known as the state-dependent process and can be either continuous or discrete-valued, or even vector-valued. The hidden process, $S = \{S_t\}_{t=1}^T$, is not directly observable and must be inferred from the data; at each time point, $S_t$ takes one of a set of $K$ discrete values $S_t \in \{1, 2, ..., K\}$, with the notation identifying by capitalisation that this process is unobservable. Finally, we define
the exogenous variables $X = \{X_t\}_{t=1}^T$, with realisations $X_{0:T-1} = (x_0, x_1, ..., x_{T-1})$, where each datapoint is a $n-$dimensional vector.

Before moving on to considering how a model might switch between regimes, we introduce the following representation of an individual regime, using a simple regression model

$$ y_t = x_{t-1} \beta_k + \sigma_k \epsilon_t, \quad (3.28) $$

with model parameters for each state $\Phi_k = \{\beta_k, \sigma_k\}$; under this specification the state-dependent distributions are gaussian. If the state sequence is known, $S_t = s_t \forall t$, then to estimate the model we would need to maximise

$$ L(M) = \sum_{t=1}^T \log f(y_t | S_t = s_t, \psi_{t-1}) $$

with respect to the model’s parameters $M = \{\Phi_k\}_{k=1}^K$, given the known information set $\psi_{t-1} = \{x_{t-1}\}$.

However, as $S$ is unknown, we instead require the joint density. For each time step this is expressed as $f(y_t, S_t | \psi_{t-1}) = f(y_t | S_t, \psi_{t-1}) f(S_t | \psi_{t-1})$. The marginal $f(y_t | \psi_{t-1})$ can then be obtained by integrating over the hidden states,

$$ f(y_t | \psi_{t-1}) = \sum_{s_t=1}^K f(y_t | S_t = s_t, \psi_{t-1}) f(S_t = s_t | \psi_{t-1}), \quad (3.29) $$

which gives the log-likelihood of the state-dependent sequence as $L(M) = \sum_{t=1}^T \log f(y_t | \psi_{t-1})$, which is the objective function required to be maximised with respect to the parameters of the regression models, and any other parameters that govern regime-switching. So far no assumptions have been made about the law that governs the evolution of $S$, i.e. $P(S_t | \psi_{t-1})$.

A Markov-switching model is a particular type of regime-switching model in which the latent process $S$ follows a Markov chain. This assumes that the state sequence at time $t$ is dependent on the state of the sequence at the previous $r$ time steps, i.e. $P(S_t | S_{t-1}, S_{t-2}, ..., S_{t-r})$. The order $r$ of the Markov chain determines the memory of the latent process. As with the vast majority of analysis performed within this framework, only first-order Markov processes are considered here. These are the simplest variant, for which $r = 1$, so the state of the latent sequence $S$ at time $t$ is assumed to be only dependent on $S_{t-1}$, and, as shown in the above paragraph, the value of the observable sequence at point $y_t$ is conditional only on $S_t$ and $x_{t-1}$.
Under these assumptions, we have a probability law governing the state-sequence $P(S_t|S_{t-1})$ and the state-dependent conditional distributions $P(y_t|S_t, x_{t-1})$.

There are a number of different ways we can estimate Markov switching regression models. One of these and the method used in the research of later chapters, is based on the expectation maximisation (EM) algorithm (Dempster, Laird, and Rubin (1977)). A and well-known and helpful framework to introduce as a preliminary explanation for the more involved algorithms to come later is that of the hidden Markov model (HMM). Considering Markov-switching regression models as HMMs with conditional state density functions, as in Langrock et al. (2017), makes it easy to generalise the framework to many different types of state-dependent models. The following section therefore contains a detailed presentation of the HMM architecture, with definitions, types of inference that can be achieved, and the Baum-Welch algorithm.

### 3.3.2 Hidden Markov Models

The hidden Markov model (HMM) of Rabiner (1989) is a generative probabilistic model widely used for modelling of sequential data. It is a highly effective way in which to represent probability distributions over sequences of observations where the source of the observations is unknown. This is achieved by the architecture of the HMM, which jointly models two stochastic processes, an unobservable state sequence $S$ and a sequence we can observe, $Y$. The only two assumptions required are that the underlying stochastic process governing the evolution of the state sequence follows a Markov chain, and that the observations are conditional on the current state of the underlying latent process.

The first of the two processes, $S = \{S_t : t \in 1 : T\}$, known as the hidden state process, is a stochastic process over a finite state space $k = \{1, 2, \ldots, K\}$, where $K$ is the number of possible states the process can be in. We define an ordered set of random variables $S_{1:T}$ to be a simple process where $S_{1:T} = \{S_1, S_2, \ldots, S_T\}$. As $S_t$ is assumed to be Markovian, it obeys the Markov property and as such the conditional independence $S_{t+1} \mid S_u \mid S_t$ holds for all $u < t$. Therefore, all information about the history of the process is obtained by knowing the state of the process at the previous time step.

The second process, $Y = \{Y_t : t \in 1 : T\}$, is a stochastic process of observable quantities that are assumed to be generated as a probabilistic function of the process $S$. Defining an ordered set of random variables $Y_{1:T} = \{Y_1, Y_2, \ldots, Y_T\}$, the HMM framework assumes two things, first that $Y_t$ is only dependent on $S_t$, therefore, $Y_t \mid S_{1:t} \mid S_t$, and second that $\forall t < \tau \ Y_t$ is independent of $Y_\tau$. These conditional independence relationships make the HMM tractable by having a
probability distribution that can be easily used for modelling, with the assistance of a handful of recursive algorithms.

The flexibility of the HMM framework allows for the hidden states to have a wide range of different types of state-dependent distributions. These can be continuous, discrete, or even a hybrid, which makes the HMM a powerful tool for a diverse group of sequential modelling problems. Fields of research that have benefitted from HMM technology include data compression, computational molecular biology, and most notably speech recognition (Rabiner, 1989).

**DEFINITION**

Considering the conditional independence assumptions of an HMM, the joint distribution of the latent sequence $S_{1:T}$ and observations $y_{1:T}$ can be written as

$$P(y_{1:T}, S_{1:T}) = P(y_1|S_1)P(S_1)\prod_{t=2}^{T} P(y_t|S_t)P(S_t|S_{t-1}).$$ (3.30)

The above distribution is assumed to be stationary so that the $P(S_t|S_{t-1}) \forall t$ do not change over time; therefore, Equation (3.30) can be represented by a parametric model by substituting the probabilities $P(S_t|S_{t-1})$ and $P(S_1)$ with fixed parameters. This means that all $P(S_t|S_{t-1})$ can be represented in the form of a matrix of transition probabilities and $P(S_1)$ by a vector of initial probabilities. For an HMM with $K$ discrete latent states, we define $A \in \mathbb{R}^{K \times K}$ as the transition matrix, where for all $i, j \in \{1, 2, ..., K\}$ we have

$$A_{i,j} = P(S_t = j|S_{t-1} = i).$$ (3.31)

With the vector $\pi$ below representing the initial state probabilities,

$$\pi_i = P(S_1 = i),$$ (3.32)

and with the state-dependent emission densities $f_Y(y_t|\Phi_k) = P(y_t|S_t = k)$, in which $\Phi_k$ is the parameter set of state-dependent distribution $k$, we can now define an HMM by the set of parameters below,

$$\mathcal{M} \in \left\{ \pi, A, \{f_Y(\cdot|\Phi_k)\}_{k=1}^{K} \right\},$$ (3.33)

where $P(S_1) = \text{multinomial}(\pi_1, ..., \pi_K)$, and $P(S_t|S_{t-1} = i) = \text{multinomial}(A_{i,1}, ..., A_{i,K}) \forall j=1:K$, this definition being dependent on the type of distribution used for the emission densities $f_Y$. 
There are a number of different inference tasks that are easily achieved within the HMM framework. The first to be discussed below is filtering, which is a recursive procedure used to infer a distribution over the hidden states at each point in time. The filtering process infers the posterior probability $P(S_t | y_{1:t}) \forall t = 1: T$, known as the filtered posterior. The filtering procedure gives us the probability of being in a certain state at time $t$ given all the observations up to and including time $t$. To obtain the posterior probabilities we first require the forward probabilities; for this we use the recursive scheme from the forward algorithm, defined below.

**Definition 1. (Forward Algorithm):** Assuming the conditional independence assumptions of an HMM, and letting $\alpha(t) = P(y_{1:t}, S_t)$, then for $t > 1$ the recursive formula for the forward probabilities is

$$P(S_t, y_{1:t}) = \sum_{s_{t-1}} P(y_t, y_{1:t-1}, S_t, S_{t-1})$$

$$= \sum_{s_{t-1}} P(y_t | S_t) P(S_t | S_{t-1}) P(S_{t-1}, y_{1:t-1})$$

$$= \left[ \begin{array}{c} \sum_{s_{t-1}} \text{correction} \sum_{s_{t-1}} \text{prediction} \end{array} \right] P(S_t | S_{t-1}) \alpha(t-1)$$

$$\alpha(t) = P(y_t | S_t) \sum_{s_{t-1}} P(S_t | S_{t-1}) \alpha(t-1)$$

with for $t = 1$, $\alpha(1) = P(S_1, y_1) = P(y_1 | S_1) P(S_1)$.

The forward algorithm can be computed efficiently with a recursive matrix calculation. For each time-step, we construct a diagonal matrix $P_t = \text{diag}(f(y_t | \Phi_1), f_2(y_t | \Phi_2), ..., f_K(y_t | \Phi_K))$ with off-diagonals equal to zero; this is known as the soft evidence and contains the information about the next observation $y_t$. Then starting at $t = 1$ with initial probability vector $\alpha(1)$,

$$\alpha(1) = \pi' P_1,$$

the recursion from $t = 2 : T$ follows as

$$\alpha(t) = \alpha(t-1) A \underbrace{P_t}_{\text{predictor}} \underbrace{P_t}_{\text{corrector}} .$$

(3.34)
Hence, the filtered distribution at the previous time-step $\alpha(t-1)$ is propagated forwards by the dynamics $A = P(S_t|S_{t-1})$ to obtain a prior distribution $P(S_t, y_{1:t-1})$ which is corrected by the introduction of the next observation $y_t$ through the soft evidence $P_t$ to obtain the new $\alpha(t) = P(S_t, y_{1:t})$. This process continues until the end of the sequence.

This recursion will quickly shrink values to zero causing the usual numerical issues encountered in probabilistic computations. Therefore, it is usually, depending on the sequence length, necessary to normalise the $\alpha(t)$ so they sum to one, thus, by Bayes theorem, obtaining the filtered posterior $\tilde{\alpha}(t) = \frac{\alpha(t)}{\sum_{i=1}^{K} \alpha(t)}$. A scaling factor is needed in order to force $\tilde{\alpha}(t-1)A P_t$ to be a distribution, i.e. to sum to one. This is achieved by summing over the $K$ states; therefore,

$$\tilde{\alpha}(t) = c_t^{-1} \tilde{\alpha}(t-1) A P_t,$$

with $c_t = \tilde{\alpha}(t-1) A P_t I$, and $I$ a vector of ones. As a consequence of scaling we obtain a series of scaling factors $c_t$, with the useful property that $\prod_{u=1}^{t} c_u = P(y_{1:t})$, which is the total probability of the observed sequence up until time $t$.

The forward algorithm gives us the forward probabilities and the filtered posterior; for the next inference task, known as smoothing, we also need backward probabilities $P(y_{t+1:T}|S_t)$. For this we require the backward algorithm.

**Definition 2.** (Backward Algorithm): Assuming the conditional independence assumptions of an HMM, and letting $\beta(t) = P(y_{t+1:T}|S_t)$, then the recursive formula for the backward probability $P(y_{t+1:T}|S_t)$ is

$$P(y_{t:T}|S_{t-1}) = \sum_{s_t} P(y_t, y_{t+1:T}, S_t|S_{t-1})$$

$$= \sum_{s_t} P(y_t|S_t) P(y_{t+1:T}|S_t) P(S_t|S_{t-1})$$

$$= \sum_{s_t} P(y_t|S_t) P(S_t|S_{t-1}) \beta(t)$$

$$= \beta(t-1)$$

$$\beta_i(T) = 1 \forall i=1:K$$

As the name suggests, the backward pass is a recursion that passes messages in the opposite direction from the forward pass of the forward algorithm. It is used to obtain the probabilities $\beta_i(t) = P(y_{t+1:T}|S_t = i)$ of the observed sequence from $t+1 : T$ given the state at time $t$; these are known as the backward probabilities.
As with the forward algorithm, this can be represented as a recursive matrix calculation. Setting the initial value at time $T$ to a vector of ones, we have the recursive scheme

$$
\beta_i(T) = 1,
\beta(t)^\top = AP(t+1)\beta(t+1)^\top.
$$

Unlike the forward pass it is not necessary to compute normalisation constants for each $\beta(t)$ as there is no use for the resulting quantities. Instead the scale factor $c_t$ that has already been computed can be used to avoid any numerical issues. An alternative method to avoid encountering numerical problems with the two algorithms, though one which is problematic to define without resorting to pseudocode, is to perform calculations in log space; for more details see Barber (2012).

From the sequences of forward and backward probabilities $\alpha(t)$ and $\beta(t)$ it is possible to compute the posterior probabilities of being in a particular state at any time period, given the information from the entire observed sequence. This is the process known as smoothing, which is an inference task similar to filtering, except the obtained quantity is the smoothed posterior $P(S_t|y_{1:T})$ for $t < T$, which is a distribution over the hidden states at some point in the past $t$ given all observations up to $T$. This is achieved with the forward-backward algorithm defined below.

**Definition 3.** (Forward-Backward Algorithm): Given $\alpha(\cdot)$ from Definition 1 and $\beta(\cdot)$ from Definition 2 and a sequence of observations $y_{1:T}$, for any $t \leq T$, define $\gamma(t) = P(S_t|y_{1:T})$; then the smoothed posterior at time $t$ is calculated by

$$
\gamma_i(t) = \frac{\alpha_i(t)\beta_i(t)}{\sum_{i=1}^K \alpha_i(t)\beta_i(t)},
$$

for all $i = 1 : K$.

The forward-backward algorithm gives us an inferred posterior distribution over hidden states using all observations in the series. To illustrate, if $t$ denotes the present, then $\gamma(t)$ may suggest changes in state which have not occurred. The smoothed posteriors, by contrast, are more extensively conditioned by using observations from the past and future, which reduces the chance that the model will misinterpret an outlier occurring within a particular state as an actual change in state.
Smoothing and filtering are two inference tasks important for the estimation of Markov-switching models; there are, however, other types available with the HMM architecture that do not need to be discussed here. For a more extensive presentation of inference tasks that can be achieved with a HMM see Zucchini and MacDonald (2009) or Barber (2012).

There is one other important inference task required for model estimation; this is the task that evaluates the marginal likelihood of all observations in the sequence, \( P(y_{1:T}) \), where the summation in this calculation is over all possible state sequences. As discussed earlier, this can easily be efficiently computed with the forward algorithm from Definition 1. The vector of normalisation factors \( c_t = \sum_{i=1}^{K} \alpha_i(t) \), for \( t = 1 : T \), represents \( P(y_t) \); therefore, under the conditional independence assumption, \( P(y_{1:T}) = \prod_{t=1}^{T} c_t \).

**MODEL ESTIMATION**

There are two main approaches to learning HMMs and consequently Markov-switching models more generally. One of these is the direct numerical optimisation of the likelihood; this is, however, more involved than typical applications of numerical optimisation. This section provides background information about the second method, which is an application of the expectation maximisation (EM) algorithm to HMMs; this is known as the **Baum-Welch algorithm** and by far the most popular method for estimating HMMs.

When working with latent variable models, parameter estimation is not a straightforward optimisation. This is due to there being two types of uncertainty to contend with; these are the usual parameter uncertainty and the uncertainty around the latent variables. The method outlined in this section, the EM algorithm, addresses the two forms of uncertainty one at a time by cycling through a two-step procedure, each step concentrating on one form of uncertainty. Thus, during each iteration, a round of EM makes updates of the model parameters \( \mathcal{M} \) and the distribution over the latent variables \( S \), in alternation, to eventually reach a self-coherent solution.

Let \( Y \) be observable, and let \( S \) be an unobservable latent variable; then the objective function for maximising the likelihood of a simple latent variable model is

\[
\mathcal{L}(\mathcal{M}) = \log P(Y|\mathcal{M}) = \log \sum_{S} P(Y, S|\mathcal{M}).
\]

The problem of maximizing \( \mathcal{L}(\mathcal{M}) \) with respect to the model parameters \( \mathcal{M} \) can be simplified by introducing a distribution \( Q(S) \) over the hidden variables. The basis for the expectation
maximisation algorithm is the maximisation of the log-likelihood. The log-likelihood can be expressed as

$$L(M) = \log \int dSP(Y, S|M)$$

$$= \log \int dSQ(S) \frac{P(Y, S|M)}{Q(S)}$$

$$\geq \int dSQ(S) \log \frac{P(Y, S|M)}{Q(S)}$$

$$= \int dSQ(s) \log P(Y, S|M) - \int dSQ(S) \log Q(S)$$

$$= \int dSQ(s) \log P(Y, S|M) + H[Q]$$

$$= F(Q, M), \quad (3.36)$$

where $Q(S)$ can be any probability distribution over the latent variables, and $H[Q]$ is the entropy of Q. In all cases $F(Q, M)$ gives rise to a lower bound of the log-likelihood. The inequality in Section 3.3.2 follows from the concavity of the log function and is known as Jenson’s Inequality. The quantity $F(Q, M) \leq L(M)$ is the expected energy under $Q$ minus the entropy of $Q$, and is known as the free energy. Section 3.3.2 can be rewritten as

$$F(Q, M) = \langle \log P(Y, S|M) \rangle_{Q(S)} + H[Q],$$

$$= \int dSQ(S) \log \frac{P(Y|S,M)P(S|M)}{Q(S)},$$

$$= \int dSQ(S) \log P(Y|M) + \int dSQ(S) \log \frac{P(S|Y,M)}{Q(S)},$$

$$= L(M) - KL[Q(S)||P(S|Y,M)] \quad (3.37)$$

where $\langle \cdot \rangle_Q$ denotes the expectation under $Q$. The second term in the last line above is the Kullback-Leibler (KL) divergence. It can be seen from the above that when the KL divergence is zero the free energy equals the log-likelihood. This happens when we substitute $Q(S)$ with $P(S|Y,M)$.

The EM algorithm is an iterative two-stage procedure which switches back and forth between a first stage (E-step) of maximising $F(Q, M)$ by inferring the latent variable distribution $P(S|Y,M)$, holding $M$ fixed, and a second stage (M-step) which maximises $F(Q, M)$ with respect to $M$, holding $Q$ fixed, using sufficient statistics obtained from the E-step. The algorithm guarantees convergence to a local maximum of the log-likelihood, with a further guarantee that there will be no decrease in likelihood after each iteration.
The Baum–Welch Algorithm is a special case of the EM algorithm (Dempster, Laird, and Rubin, 1977). The algorithm is tailored to the specific structure of a hidden Markov model and will converge to a local maximum of the likelihood. An HMM \( \mathcal{M} \) as specified by the parameter set in Equation (3.33) can be learned from a sequence of observations \( y_{1:T} \). For a given set of initial parameters we seek an optimal parametrisation \( \mathcal{M}^* \) that satisfies

\[
\mathcal{M}^* = \arg \max_{\mathcal{M}} P(y_{1:T}; S_{1:T} | \mathcal{M}). \quad (3.38)
\]

Tailoring the EM algorithm to solve Equation (3.38) requires applying Equation (3.37) while considering the specific conditional independence structure of the HMM. Assuming \( Q \) is optimal, applying Section 3.3.2 to the likelihood Equation (3.30), the log-likelihood can be written in terms of energy as

\[
\mathcal{F}(Q, \mathcal{M}) = \langle \log P(y_{1:T}; S_{1:T} | \mathcal{M}) \rangle_{Q(S_{1:T})},
\]

\[
= \langle \log P(S_1 | \mathcal{M}) \rangle_{Q(S_1)} + \sum_{t=2}^{T} \langle \log P(S_t | S_{t-1}, \mathcal{M}) \rangle_{Q(S_t, S_{t-1})} + \sum_{t=2}^{T} \langle \log P(y_t | S_t, \mathcal{M}) \rangle_{Q(S_t)}.
\]

Therefore, to take the above expectation, the Baum–Welch algorithm requires three separate types of distribution over the latent variables to be inferred. Due to the conditional independence assumptions of the HMM and the exact inference achieved with the forward-backward algorithm Definition 3 the \( Q_s \) can be set to be the conditional distributions for the initial state probability \( Q(S_1) = P(S_1 | y_{1:T}) \), the pair-wise marginal posteriors set to \( Q(S_t, S_{t-1}) = P(S_t, S_{t-1} | y_{1:T}) \), and the marginal posteriors to \( Q(S_t) = P(S_t | y_{1:T}) \); this choice for the \( Q_s \) is equivalent to setting \( \mathcal{F}(Q, \mathcal{M}) = \mathcal{L}(\mathcal{M}) \) and so the lower bound on the log-likelihood becomes the log-likelihood.

**E-STEP:**

The sufficient statistics for the discrete distributions \( P(S_1 | y_{1:T}) \), \( P(S_t | y_{1:T}) \) and \( P(S_t, S_{t-1} | y_{1:T}) \) can be represented in terms of expectations of indicator variables:

\[
z_i(1) = \mathbb{E} [\mathbb{I}_{s_1=i} | y_{1:T}] = P(S_1 = i | y_{1:T}), \quad (3.39)
\]

\[
\phi_{i,j}(t) = \mathbb{E} [\mathbb{I}_{s_t=j} \mathbb{I}_{s_{t-1}=i} | y_{1:T}] = P(S_t = j, S_{t-1} = i | y_{1:T}),
\]

\[
\gamma_i(t) = \mathbb{E} [\mathbb{I}_{s_t=i} | y_{1:T}] = P(S_t = i | y_{1:T}).
\]
Inserting these into the objective function and replacing probabilistic terms with their corresponding model parameters, i.e. $\pi_i = P(S_1 = i)$ and $A_{i,j} = P(S_t = i | S_{t-1} = j)$, gives the complete data log-likelihood function (CDLL),

$$CDLL = \log \left< p(s_1) P(y_1 | s_1) \prod_{t=2}^{T} p(s_t | s_{t-1}) P(y_t | s_t) \right>$$

$$= \sum_{i=1}^{K} z_i(1) \log (\pi_i) + \sum_{i=1}^{K} \sum_{j=1}^{K} \sum_{t=2}^{T} \phi_{i,j}(t) \log (A_{j,i}) + \sum_{i=1}^{K} \sum_{t=1}^{T} \gamma_i(t) \log f_i(y_t | \Phi_i). \quad (3.40)$$

These sufficient statistics can be obtained efficiently with the forward-backward algorithm. From the definition of the functions $\{z_i, \{\phi_{i,j}(t)\}_{t=2}^{T}, \{\gamma_i(t)\}_{t=1}^{T}\}_{i,j=1}^{K}$ in Equation (3.39) and with $\alpha$ from the forward algorithm Definition 1 and $\beta$ defined in Definition 2, it follows immediately from an application of Bayes Theorem that

$$\gamma_i(t) = \frac{\alpha_i(t) \beta_i(t)}{\sum_{j=1}^{K} \alpha_j(t) \beta_j(t)}, \quad (3.41)$$

$$\phi_{i,j}(t) = \frac{\alpha_i(t-1) A_{i,j} \beta_j(t) f(y_t | \Phi_j)}{\sum_{j=1}^{K} \alpha_j(t) \beta_j(t)}, \quad (3.42)$$

$$z_i(1) = \gamma_i(1).$$

**M-STEP**

With the sufficient statistics computed in the E-step above, the next step involves inserting them into the complete data log-likelihood and maximising it given these estimates. This includes solving for the HMM parameters $\pi$ and $A$ and the state dependent emission parameters $\{\Phi_k\}_{k=1}^{K}$.

The solutions for $\pi$ and $A$ are as follows:

$$\pi_i = \gamma_i(t), \quad (3.43)$$

$$A_{i,j} = \frac{\sum_{t=2}^{T} \phi_{i,j}(t)}{\sum_{j=1}^{K} \sum_{t=2}^{T} \phi_{i,j}(t)}. \quad (3.44)$$

The emission parameters $\{\Phi_k\}_{k=1}^{K}$ are estimated using maximum-likelihood estimation, weighting the observations when solving for each state density by the vector $\gamma_k(t)$ for $k = 1 : K$.

Algorithm 7 shows the procedure to estimate a simple HMM with arbitrary emission densities, as has been described in this section.
Inputs: a sequence of observations \( \{ y_t \}_{t=1}^T \), a choice of emission distribution \( f_{Y}(y_t|\Phi) \), a number of states \( K \).

Initialise: \( \{ \Phi_k \}_{k=1}^K \) with some initial values for the emission parameters, and \( \pi[1] \) and \( A[1] \).

- For \( m = 1 : M \)

  - (E-step)
    - \( \alpha[m], \beta[m] \leftarrow \text{From forward-backward algorithm Definition 3, given } \{ \Phi_k \}_{k=1}^K, \pi[m] \text{ and } A[m] \),
    - \( \ell[m] \leftarrow \begin{cases} 
      \sum_k K \alpha_k[m](T) & \text{un-normalized } \alpha \\
      \prod_{t=1}^T c_t & \text{scaled } \alpha 
    \end{cases} \)
    - Estimate sufficient statistics \( \left\{ \left\{ \phi_{i,j}[m](t) \right\}_{t=2}^T, \left\{ \gamma_i[m](t) \right\}_{t=1}^T \right\}_{i,j=1}^K \) from Equation (3.42), and Equation (3.41)

  - (M-step)
    - \( \pi[m+1], A[m+1] \leftarrow \text{from Equation (3.43) and Equation (3.44)} \),
    - \( \{ \Phi_k[m+1] \}_{k=1}^K \leftarrow \text{estimated using maximum likelihood, weighting the observations by } \gamma_k[m](t) \).

- Check convergence: if \( \ell[m] - \ell[m-1] < C \) break

- end loop

Output: \( \mathcal{M}^* \in \left\{ \pi^*, A^*, \{ f_{Y}(y|\Phi_k^*) \}_{k=1}^K \right\} \).

Algorithm 7: The Baum–Welch algorithm for hidden Markov models.
The architecture of the HMM forms the probabilistic framework for the models that will be introduced later in this thesis. In this standard form, the HMM has many limitations for modelling financial time-series. The parameters of the state-dependent distributions are fixed weighted averages representative of the entire sequence. They are the simplest unconditional estimates and do not evolve over time. If estimates of these state-dependent parameters need to vary in response to influence from an exogenous source, they can be rendered time-varying through conditioning on a set of exogenous variables. To achieve this, the unconditional estimates of the HMM emission parameters could be replaced by regression-type models to provide conditional estimates of these parameters, while still having the advantage of using most of the HMM architecture. The following section outlines such a type of model.

3.3.3 Markov-switching Distributional Regression Models

The original Markov-switching regression models focused on estimating the means of the state-dependent distributions as functions of exogenous variables, while keeping the other parameters that specify these distributions fixed. This has been shown to be a very effective way of modelling economic and financial time-series data, as the mixture distribution that results from the switching mechanism allows for time-varying variances and higher moments. This time-variation, however, comes from the conditional dependence on the state of the Markov chain, and, for these models, no additional dependence based on exogenous variables are assumed for parameters other than the mean. It is, however, likely that within each state-dependent distribution, the other parameters, such as those that effect variance, skewness, and kurtosis may also be conditional on exogenous variables. In these situations the GAMLSS framework can be used to model the state-dependent conditional distributions.

The GAMLSS framework of Rigby and D. M. Stasinopoulos (2005) has become a very powerful way of modelling fully conditional probability distributions, as discussed in Section 3.1.4.1, and it is natural when looking to extend the GAMLSS to time-series modelling to consider embedding these conditional distributions as state-dependent distributions in a Markov-switching framework. This was first proposed in Langrock et al. (2017) and Hambuckers et al. (2018) and Adam, Mayr, and Kneib (2017) both of whom exploit the HMM architecture to embed GAMLSS into a Markov-switching model for time-series applications. These two examples of what is termed as Markov-switching generalised additive models for location, scale and shape (MS-GAMLSS) were introduced in these papers along with two different model estimation methods. The first uses numerical maximisation of the likelihood, and the second uses the EM algorithm along with component-wise gradient boosting. In this section, in Section 3.3.3.1, the latter method for fitting MS-GAMLSS models is presented; this is the MS-gamBoostLSS algorithm of Adam,
3.3 MARKOV-SWITCHING REGRESSION MODELS

Figure 3.2 illustrates the importance of considering GAMLSS models over more traditional regression models in a regime switching setting. The data is generated by a two-state latent process where for each state, the data is distributed by a conditional Gaussian, where both the mean and standard deviation are functions of a single covariate. So that these relationships could be plotted the covariate is assumed to be monotonically increasing with time, Hence the x-axis explicitly represents the value of the covariate, and can be thought to implicitly represent time. The colour of each data point represents the regime that generated it. Both models estimate the conditional means of each state well; however, it is only the MS-GAMLSS that can capture the changing relationship between the covariate and the standard deviations in each state.

Before presenting this method, some important differences between the MS-GAMLSS model used here and that used elsewhere in the literature need to be noted. Primarily previous forms of the MS-GAMLSS model have not considered the density \( f(y_t | \varphi_{t-1}) \); instead they focus on conditioning on a contemporaneous information set \( \varphi_t \). The MS-GAMLSS models in this thesis are adapted for use as forecasting models, which requires a modest change to the notation found in the literature. As with the notation adopted previously in the section, the forecasting index is \( t \), with only information up and including \( t - 1 \) being used for conditioning.
3.3 MARKOV-SWITCHING REGRESSION MODELS

Consider an HMM with the set of emission densities \( \{ f_Y (y_t; \Phi_{t,k}) \}_{k=1}^K \) with each \( \Phi_k \) for \( k = 1 : K \) representing a state-dependent set of conditional forecasts of the parameters of a pre-specified distribution, which may or may not be in the exponential family. In order to describe the conditional densities more accurately, the notation for the state-dependent densities moving forward is \( \{ f_Y (y_t | X_{t-1}; \Phi_{t,k}) \}_{k=1}^K \), indicating that \( y_t \) is conditioned on \( X_{t-1} \), i.e. \( (y_t | X_{t-1}) \) and that the density \( f_Y \) is parameterised by \( \Phi_{t,k} \), i.e. \( f_Y (y_t; \Phi_{t,k}) \).

Each of the state-dependent GAMLSS models of the conditional densities contains the quantities

\[
\Phi_{t,k} = \{ \hat{\mu}_{t,k}, \hat{\sigma}_{t,k}, \hat{\nu}_{t,k}, \hat{\tau}_{t,k} \},
\Xi_k = \{ \eta_{\mu_k}, \eta_{\sigma_k}, \eta_{\nu_k}, \eta_{\tau_k} \},
G^{-1} = \{ g^{-1}_{\mu}(\eta_{\mu_k}), g^{-1}_{\sigma}(\eta_{\sigma_k}), g^{-1}_{\nu}(\eta_{\nu_k}), g^{-1}_{\tau}(\eta_{\tau_k}) \},
\tag{3.45}
\]

where \( \Xi_k \) is the set of additive predictors associated to the parameters of the density for state \( k \) and \( G^{-1} \) is the set of inverse link functions. Members of these sets are related in the following ways:

\[
\hat{\mu}_{t,k} = g^{-1}_{\mu}(\eta_{\mu_k}(X_{t-1})), \quad \hat{\sigma}_{t,k} = g^{-1}_{\sigma}(\eta_{\sigma_k}(X_{t-1})), \quad \hat{\nu}_{t,k} = g^{-1}_{\nu}(\eta_{\nu_k}(X_{t-1})), \quad \hat{\tau}_{t,k} = g^{-1}_{\tau}(\eta_{\tau_k}(X_{t-1})).
\]

One-step-ahead parameter estimates are produced by the prediction functions

\[
\hat{\theta}_{t,k} = g^{-1}_{\theta}(\eta_{\theta_k}(X_{t-1})),
\eta_{\theta_k}(X_{t-1}) = \beta_0 + \sum_{m=1}^{M} f^{[m]}(X_{t-1})
\]

for all \( \hat{\theta}_{t,k} \in \Phi_{t,k} \), and the fully conditional emission densities can be defined by

\[
f_Y (y_t | X_{t-1}; \Phi_{t,k}) = f_Y (y_t | X_{t-1}; \hat{\mu}_{t,k}, \hat{\sigma}_{t,k}, \hat{\nu}_{t,k}, \hat{\tau}_{t,k})
\quad \text{for } k = 1 : K, t = 1 : T.
\]

In the Markov-switching (MS) models used throughout this study, the switching process is an ergodic first-order Markov chain with a finite number of states, defined by an initial state probability vector and the transition probability matrix. The full set of components that specify an MS model of this type are

\[
\mathcal{M} \in \left\{ \pi, A, \{ f_Y (\cdot; \Phi_k) \}_{k=1}^K \right\}.
\]

\footnote{For clarification of terminology, state dependent distributions / emission densities in a regression setting are more commonly referred to as error distributions; this text uses all three terms interchangeably.}
where \( \pi \) and \( \mathcal{A} \) are defined in Section 3.3.2.

**GRADIENT BOOSTING FOR MARKOV-SWITCHING GAMLLSS**

The procedure for learning the MS-GAMLSS model in this way consists of embedding a weighted form of the component-wise gradient boosting algorithm that is described in Section 3.2.2 as an inner cycle within the EM algorithm. This as the method first proposed in Adam, Mayr, and Kneib (2017) and known as the *MS-gamBoostLSS algorithm*.

The E-step, in a similar way to a standard HMM (Algorithm 7), requires the following expectations, however, in this case we are also conditioning on \( x_{0:T-1} \):

\[
\begin{align*}
z_i(1) &= \mathbb{E}[I_{s_1=1}|y_{1:T}, X_{0:T-1}] = P(S_1 = i|y_{1:T}, X_{0:T-1}), \\
\phi_{i,j}(t) &= \mathbb{E}[I_{s_t=j}I_{s_{t-1}=i}|y_{1:T}, X_{0:T-1}] = P(S_t = j, S_{t-1} = i|y_{1:T}, X_{0:T-1}), \\
\gamma_i(t) &= \mathbb{E}[I_{s_t=i}|y_{1:T}, X_{0:T-1}] = P(S_t = i|y_{1:T}, X_{0:T-1}).
\end{align*}
\]  

(3.46)

In the M-Step, component-wise boosting replaces the weighted maximum-likelihood estimation (MLE) method for estimating state-dependent densities found in the Baum-Welch algorithm. An important difference between learning a switching model and a standard non-switching GAMLSS is that in the switching case each observation is weighted by the smoothed posterior \( P(S_t|y_{i:T}) \). Finally, the HMM parameters can be learned from the related step of the Baum–Welch algorithm Algorithm 7.

For updating the state-dependent models, the loss function is the negative of the complete data log-likelihood, Equation (3.40), and is only partially dependent on \( \Phi \); hence this loss function can be reduced to the weighted negative log-likelihood of the response variable:

\[
-CDDL = -\sum_{t=1}^{T} \rho(y_t, \Xi_{t,k}) = -\sum_{i=1}^{K} \gamma_i(t) \sum_{t=1}^{T} \log f_Y(y_t|X_{t-1}; \Phi_{k,t}^i) .
\]

The gradient vectors required for the fitting procedure can be computed quite easily as only one of the three terms of the *CDDL* Equation (3.40) is dependent on \( \Phi_k \). For the current parameter
to be updated, denoted $\theta$, the score function with respect to the predictor of $\theta$, $g(\theta) = \eta$ is as
for the RS algorithm Equation (3.15):
\[
\frac{\partial CDLL}{\partial \eta_{\theta,k}(x_t)} = -\gamma(t) \frac{\partial \log f_Y(y_t|X_{t-1}; \Phi_{k,t})}{\partial \eta_{\theta,k}(X_{t-1})} \frac{\partial g^{-1}_\theta (\eta_{\theta,k}(X_{t-1}))}{\partial g^{-1}_\theta (\eta_{\theta,k}(X_{t-1}))} = -\gamma(t) \frac{\partial \log f_Y(y_t|X_{t-1}; \Phi_{k,t})}{\partial \eta_{\theta,k}(X_{t-1})}.
\]
These gradients are evaluated with the most up-to-date values for the parameter estimates in accordance with the cyclical update scheme in Equation (3.26),
\[
\nabla_{\eta_{\theta,k},t}^{[m+1]} = -\gamma(t) \frac{\partial \log f_k(y_t|X_{t-1}; \Phi_{k,t})}{\partial \eta_{\theta,k}(X_{t-1})} \bigg|_{\Xi = \begin{cases} \Xi^{[m]}(\theta_u) & \text{if } u \geq p \\ \Xi^{[m+1]}(\theta_u) & \text{if } u < p \end{cases}}.
\]
Algorithm 8 outlines the steps of the MS-gamBoostLSS algorithm.

The framework described above is highly flexible with respect to the choice of response distributions which can be used to represent each state, as well as flexible in the choice of base learners that can be pre-specified as candidate components of the predictor functions for each of the parameters of the response distributions. It is this flexibility that makes the above procedure a valid candidate for multivariate distributions, as will be shown in the next chapter.
3.3 MARKOV-SWITCHING REGRESSION MODELS

Inputs: response vector and sets of covariates $Y, \{X_\theta\}_{k=1}^P$, pre-specified base-learners / covariate subset components $\{H_\theta\}_{k=1}^P$, response distribution density function $f_Y$ and corresponding loss function $\rho$, shrinkage factor $0 < \delta \leq 1$, number of boosting rounds $M$.

Initialise: set initial values for the additive predictors $\Xi^{[0]} = \{\eta_0^{[0]}, \eta_0^{[0]}, \eta_0^{[0]}, \eta_0^{[0]}\}$; for example, $\eta_0^{[0]} = 0$ or $\eta_0^{[0]} = g_{\theta_{\text{MLE}}} (\hat{\theta}_{\text{MLE}})$ for $k = 1, ..., P$.

- For $i = 1 : M$ (EM round)
  - (E-Step)
    - $\alpha^{[m]}, \beta^{[m]} \leftarrow$ from forward-backward algorithm Definition 3, given $\{\Phi^{[m]}_{k}\}_{k=1}^P$, $\pi^{[m]}$ and $A^{[m]}$,
    
    \[ \ell^{[m]} \leftarrow \begin{cases} \sum_k^K \alpha^{[m]}_k (T) & \text{un-normalised } \alpha \\ \prod_{t=1}^T c_t & \text{scaled } \alpha \end{cases} \]

    - estimate sufficient statistics $\{\{\phi^{[m]}_{i,j} (t)\}_{t=2}^T, \{\gamma^{[m]}_i (t)\}_{t=1}^T\}_{i,j=1}^K$ from Equation (3.42).
  
  - (M-step)
    
    \[ \pi^{[m+1]}, A^{[m+1]} \leftarrow \text{from Equation (3.43) and Equation (3.44),} \]

    - For $k = 1 : K$ (cycle through states)
      * For $m = 0 : M$ (boosting round)
        - For $p = 1 : P$ (cycle through parameter set)
          - Compute negative gradient vector, evaluated with estimates from most recent set of predictor functions,
            \[ \nabla^{[m+1]}_{\eta_p} = - \frac{\partial \rho (y, \Xi (\theta_p))}{\partial \eta_p} \bigg|_{\Xi = \begin{cases} \Xi^{[m]} (\theta_u) & \text{if } u \geq p \\ \Xi^{[m+1]} (\theta_u) & \text{if } u < p \end{cases}} \]

          - Fit each of the base learners in $H_{\theta_k}$ to the negative gradient vector to get a set of $J$ regression estimators (typically the fit is in the least squares sense, but any loss function is possible); therefore, for each $h_j \in H_{\theta_k}$ we obtain $H^*_k = \{h^*_1, h^*_2, ..., h^*_J\}$, where
            \[ h^*_j := \min_{h_j} \sum_{t=1}^T \left( \nabla_{t, \eta_{\theta_k}} - h_j (X_t, \theta_k) \right)^2, \]  
            (3.47)

          - Select from $H^*_k$ the $h^*_j$ which had the smallest error in (Equation (3.47)), designated $h^{**}$. 

          - Update predictor with new component $h^{[m+1]} = h^{**}$
            \[ \eta^{[m+1]}_{\theta_k} = \eta^{[m]}_{\theta_k} + \delta h^{[m+1]} (X_{\theta_k}), \]
            \[ \hat{\theta}^{[m+1]}_{\theta_k} = g_\theta^{-1} (\eta^{[m+1]}_{\theta_k} (X_{\theta_k})). \]

        - end boosting round
      * end parameter cycle
    * end state cycle
  - end EM round
Part II

PROBABILISTIC TIME-SERIES MODELLING
In this first of the three research chapters, a bivariate extension to the Markov-switching generalised additive model for location, scale and shape (MS-GAMLSS) is introduced. The model is used to forecast time-varying joint distributions of financial time series, and, specifically, to provide dynamic estimates of mean, variance and correlation for pairs of financial asset returns. The model is implemented as the forecasting model of a systematic trading strategy for dynamic pairs trading.
4.1 INTRODUCTION

In this chapter, the first stage of development of the multivariate modelling framework of this thesis, a regime-switching distributional regression model is introduced for forecasting of the predictive distribution of pairs of financial asset returns. The model is designed specifically to produce rolling probabilistic forecasts of a bivariate distribution that update over time as new information is revealed, thus providing input signals for a dynamic asset allocation strategy involving active trading of two assets. The probabilistic forecasting model provides estimates of the mean, variance and covariance that change over time in two ways, first, by conditioning predictions on high-dimensional sets of exogenous covariates, and, second, by conditioning on the state of an unobservable Markov chain.

For this initial stage, before, in the next chapter, moving on to the full multivariate modelling framework, the choice for the form of the state-dependent densities is made to be as simple as possible, in order to serve as a test of the underlying model-building principles. The number of parameters, and, therefore, the complexity of the state-dependent models, was intentionally limited, with the bivariate gaussian distribution being chosen as the specification for each state model. The generalised additive model of location, scale and shape (GAMLSS) distributional regression framework was chosen for modelling of these state-dependent distributions; more specifically, a variant of the GAMLSS trained by component-wise gradient boosting was the method selected for building the bivariate gaussian models. Within this framework, all distributional parameters are estimated by individual prediction functions, learned via gradient boosting of generalised additive models (GAMs), a method first seen in Mayr et al. (2016) and later extended in Adam, Mayr and Kneib (2017) to univariate switching distributions. Using this gradient boosting method, known as the *gamBoostLSS algorithm*, each prediction function is an ensemble containing a large set of linear, non-linear or smooth functions of explanatory variables. As well as being suited for high dimensional data, in a way that traditional GAMLSS estimation methods are not (see, Mayr et al. (2012) and Vatter and Chavez-Demoulin (2015) for detailed reasoning), gamBoostLSS also has inbuilt variable selection, which means that, in theory, only the most relevant covariates are chosen to be included for each of the state-dependent models, meaning that different regimes can have different conditioning variables driving the conditional forecasts.

For the univariate methods mentioned above to be applicable to bivariate analysis, the topic of this chapter, the chapter will introduce a method to extend the GAMLSS modelling framework,
and related gamBoostLSS algorithm, from a method for univariate response variables to one that models bivariate response vectors, which includes the modelling of dependence; for the bivariate normal distribution this means using a full covariance matrix. For this extension to guarantee that covariances are well defined, a Cholesky decomposition of the covariance matrix is used, where the Cholesky factors are individually estimated by gradient boosting GAMs. The choice of decomposition follows the review of Pourahmadi (2011), who studied the possible decompositions of the covariance matrix most suitable for estimation in a regression setting, finding that the Cholesky decomposition has clear benefits over other methods such as variance-correlation decomposition or spectral decomposition.

To test the efficacy of the model this chapter will backtest a dynamic portfolio strategy in which the Markov-switching bivariate gaussian GAMLSS is used to provide the time-varying forecasts of return, variance and correlation for dynamic pairs trading, using simple mean-variance portfolio selection methods. A set of 40 U.S. equities are paired into 20 components within an equal weighted portfolio. The bivariate gaussian MS-GAMLSS is applied to jointly model each pair of time series. Strategies are run over a year and strategy performance is compared to benchmarks. To ensure that testing conditions are as realistic a reflection of live trading as possible, transaction costs obtained from real market data are used.

The remainder of this chapter is as follows. In Section 4.2 the bivariate extension to the MS-GAMLSS is introduced. For brevity, only information that applies to the model extension is discussed, and the reader is referred to Chapter 3 for a complete background to the univariate MS-GAMLSS model and the gamBoostLSS algorithm, as well as the details of regime-switching with Markov models. Section 4.3 introduces the datasets used in the upcoming experiments, followed in Section 4.4 by the main experiment, with its results in Section 4.5, and, finally concluding remarks in Section 4.6.
4.2 MODEL DESIGN AND ESTIMATION

The bivariate gaussian MS-GAMLSS is a regime-switching distributional regression model for forecasting a switching bivariate gaussian distribution. The model consists of two component mechanisms: a set of bivariate gaussian GAMLSS models, one for each state-dependent density; and a dynamics model that mixes the forecasts emitted from the state models, based on state uncertainty.

This section describes how this model will be built, starting with Section 4.2.1, which looks at the way in which the states of the regime-switching model are represented; this includes the specification for the state-dependent models in the context of the GAMLSS framework, thereby introducing a method for extending the GAMLSS framework to cater for a bivariate gaussian GAMLSS model. This is followed by Section 4.2.2, which introduces the method used to estimate the model, a procedure that combines the expectation maximisation algorithm, as applied to HMMs, with the gamBoostLSS algorithm for component-wise gradient boosting. Most of the background material for the procedure described here can be found in Chapter 3; the description in this section focuses on the adaptations and additions made to existing methods which are required to build the model, with other material referenced where applicable. The types of base learners the gradient boosting method combines to create the additive predictor functions are then discussed in Section 4.2.3, and the section ends with Section 4.2.4, which contains a discussion of the difficulty the new model’s architecture poses for model selection and outlines a heuristic method to address it.

4.2.1 Bivariate Extension for a Gaussian GAMLSS Model

The main challenge when extending the GAMLSS to a framework for multivariate analysis of joint response distributions is to determine the most effective way to model dependencies between response variables. The best choice of method will depend on a number of factors: 1) whether or not an out-of-the-box family of multivariate distributions exists for the response; 2) whether or not all marginals are from the same family of distribution; 3) whether the problem needs to consider non-linear or asymmetrical dependence structures; 4) the number of additional parameters required to model dependence; and 5) the constraints that may have to be placed on estimates to ensure the distribution is well-defined and any consequences from these constraints for the process of estimation.
If for factors 1) or 2) the answer is ‘no’, noting that 2) implies 1), then a copula-based approach will be necessary; if 3) is a consideration then copula are, independently of those concerns, more likely to provide better flexibility than a multivariate gaussian or Student’s t. With a copula approach, each response variable (marginal) is modelled independently, as if it is from a univariate distribution. The joint density can then be represented as the product of the univariate marginal densities and a parametric copula density function for modelling dependence structure. The copula parameters can then be estimated in the usual way, with unconditional maximum likelihood methods or, as will be seen in the following chapter, for the GAMLSS framework, modelled as functions of exogenous variables.

The work on conditional copula is quite extensive; for a review see Patton (2012). For the approach taken in this chapter, conditional copula methods are not deemed necessary, as a simple choice has been made here for state-dependent joint distributions, by using a bivariate gaussian, parameterised by a mean vector and full covariance matrix. With this choice, factors 1) to 4) above are not a concern. However, the fifth factor remains a concern when choosing a method for scale and dependence modelling, especially with distributional regression. Any method used must have a guarantee that the estimated covariance matrix will always be well defined; estimation of the covariance matrix using regression-type models does not provide this guarantee. Therefore, for the model specification of the GAMLSS used here, the covariance matrix is factorised using its Cholesky decomposition, and each Cholesky factor becomes a conditional parameter modelled by a GAM, modelled using gradient boosting of a generalised additive model.

The main methods for estimating covariance / precision matrices are reviewed in Pourahmadi (2011), which investigated a variety of possible decompositions of the covariance matrix that allow for regression methods to be used. Of the main methods reviewed in this study, the Cholesky decomposition was found to have clear benefits over other methods such as variance-correlation decomposition or spectral decomposition. The Cholesky decomposition allows for unconstrained estimation of the individual Cholesky factors as functions of exogenous variables, while always guaranteeing the positive-definiteness of the estimated covariance matrix. Pourahmadi’s review argues that this method does, however, have its own problem, namely that the order in which the Cholesky factors are estimated must be carefully chosen. Due to the nature of the updating scheme of the gamBoostLSS algorithm, which cycles through parameters, only making incremental improvements each time, this should not be of any concern, or, at the very least, any effect of the ordering should be small. Additionally, with this choice of factorisation, analytical derivatives are available; hence, this method was considered to be the best choice for use with gradient boosting for the GAMLSS framework.
### THE STATE DEPENDENT MODELS

The joint distribution of two normally distributed random variables has the density

\[
f_Y(y|\Phi) = (2\pi |\Sigma|^\frac{1}{2})^{-1} \exp \left( -\frac{1}{2} (y - \mu)^\top \Sigma^{-1} (y - \mu) \right). \tag{4.1}
\]

For \( y \in \mathbb{R}^2, \mu \in \mathbb{R}^2 \) and \( \Sigma \in \mathbb{R}^{2 \times 2} \), the conditional parameter set \( \Phi \) has five parameters to estimate. For a more general multivariate normal distribution there are \( |\Phi| = 2d + d(d-1)/2 \) individual regression models; in this bivariate case \( |\Phi| = 5 \). The parameter set for the bivariate distribution is \( \Phi = \{ \hat{\mu}_1, \hat{\mu}_2, \hat{a}, \hat{b}, \hat{c} \} \); following the notation in Equation (3.45), \( \Xi_k \) is the set of additive predictors associated to the parameters on state \( k \) and \( G^{-1} \) is the set of inverse link functions.

The quantities are related in the following way:

\[
\hat{\mu}_{1,t+1} = g_{\mu_1}^{-1}(\eta_{\mu_1}(x_t)), \hat{\mu}_{2,t+1} = g_{\mu_2}^{-1}(\eta_{\mu_2}(x_t)), \hat{a}_{t+1} = g_a^{-1}(\eta_a(x_t)), \hat{b}_{t+1} = g_b^{-1}(\eta_b(x_t)), \hat{c}_{t+1} = g_c^{-1}(\eta_{c,k}(x_t)), \text{ and the conditional density functions are therefore}
\]

\[
f_Y(y_t; \Phi_{t,k}) = f_Y(y_t; \hat{\mu}_{t,k}, \hat{\sigma}_{t,k}, \hat{a}_{t,k}, \hat{b}_{t,k}, \hat{c}_{t,k}),
\]

for \( k = 1 : K, t = 1 : T - 1 \),

where the parameters \( a, b, c \) are the entries in the Cholesky decomposition of a covariance matrix such that \( \Sigma = LL^\top \) and \( L = \begin{bmatrix} a & 0 \\ b & c \end{bmatrix} \). It is required for \( \Sigma \) to be positive semi-definite; as such a unique \( L \) does not exist. Hence to ensure uniqueness a positivity constraint is imposed on the diagonal entries of \( L \). This is achieved by setting the monotonic link functions for parameters \( a \) and \( c \) to \( g_\theta(\theta) = log(\theta) \forall \theta \in \{a,c\} \). Additionally, the link function for all other parameters is simply the identity link function \( g_\theta(\theta) = \theta \forall \theta \in \{\mu_1, \mu_2, b\} \). Therefore, for each point in time the estimated conditional covariance matrix for the bivariate case can be recovered by the matrix product of the decomposition

\[
\hat{\Sigma}(X_t) = \begin{bmatrix} \hat{a}_t^2 & \hat{a}_t \hat{b}_t \\ \hat{a}_t \hat{b}_t & \hat{b}_t^2 + \hat{c}_t^2 \end{bmatrix}
\]

\[
= \begin{bmatrix} \exp(2\eta_a(X_t)) & \exp(\eta_a(X_t)) \eta_b(X_t) \\ \exp(\eta_a(X_t)) \eta_b(X_t) & \eta_b(X_t)^2 + \exp(2\eta_b(X_t)) \end{bmatrix}.
\]

This decomposition ensures that the resulting estimates for the matrix \( \Sigma \) are always symmetric and positive semi-definite.
For this particular model specification, the distributional regression consists of the five following additive regression models

\[ \hat{\mu}_1 = g_{\mu_1}^{-1} \left( \beta_{\mu_1}^{[0]} + \sum_{m=1}^{M_{\mu_1}} f[m](X_{\mu_1}) \right) \]

\[ \hat{\mu}_2 = g_{\mu_2}^{-1} \left( \beta_{\mu_2}^{[0]} + \sum_{m=1}^{M_{\mu_2}} f[m](X_{\mu_2}) \right) \]

\[ \hat{a} = g_{a}^{-1} \left( \beta_{a}^{[0]} + \sum_{m=1}^{M_{a}} f[m](X_a) \right) \]

\[ \hat{b} = g_{b}^{-1} \left( \beta_{b}^{[0]} + \sum_{m=1}^{M_{b}} f[m](X_b) \right) \]

\[ \hat{c} = g_{c}^{-1} \left( \beta_{c}^{[0]} + \sum_{m=1}^{M_{c}} f[m](X_c) \right) \]

where the \( \beta \)'s are the initial offset values for each predictor, the additive terms \( f[m] \) are functions of the (single) exogenous variable in \( X \) that was chosen at boosting round \( m \), and the \( M \)'s are the total number of boosting rounds used to build each additive model. Finally, the different subscripts of \( X \) denote that the sets of covariates may be different for each parameter; these subscripts will be subsequently dropped, in order to simplify the notation.

### 4.2.2 Markov-switching Bivariate GAMLSS with Component-wise Gradient Boosting

The procedure for learning an MS-GAMLSS model consists of two distinct algorithms; these are referred to in Adam, Mayr and Kneib (2017) as the inner and outer cycles. The outer cycle is formed from a variant of the Baum-Welch algorithm for hidden Markov models, as discussed in section Section 3.3.2. The inner cycle is a weighted form of the GAMLSS gradient boosting (GAMBoostLSS) algorithm of Mayr et al. (2012) Hofner, Mayr and Schmid (2014), as previously described in Section 3.2.3.

For each round of the outer cycle, a new set of GAMLSS models are built, i.e. one model for each of the \( K \) states. Both the outer and inner algorithms are iterative procedures; the outer procedure is that of the expectation maximisation (EM) algorithm. In the first step of EM the uncertainty over the hidden state sequence is inferred (E-step). These inferred posterior distributions over the hidden states translate into \( K \) weightings of the dataset; the inner cycle is then called, with new sets of observation weights, re-estimating the GAMLSS models as part of the M-step. Analogous to the Baum–Welch algorithm for HMMs, the M-step also updates the Markov transition density matrix and initial state probabilities vector. This process is repeated
with the updated models and the new Markov parameters until some convergence criterion is met.

The full MS-GAMLSS is defined in Section 3.3.3 and the GAMBoostLSS gradient boosting method for learning state dependent predictors can be found in Section 3.2.3. The description presented in Chapter 3 is for general state-dependent distributions of a univariate response variable; the reader is referred to Section 3.1.4 and Section 3.2.3 for a comprehensive guide. In this section, only a brief review of the problem is provided; however, the details of the modifications to the MS-GAMBoostLSS learning procedure for a bivariate normal extension, and the specifics required to achieve the extension, are covered in detail.

The general goal of the algorithm is to maximise the complete data log-likelihood (CDLL)

\[
CDLL = \left\{ \log p(S_1 = s_1) + \sum_{t=1}^{T-1} \log p(S_{t+1} = s_{t+1}|S_t = s_t) + \sum_{t=1}^{T-1} \log P(y_{t+1}|X_t, S_{t+1} = s_{t+1}) \right\}
\]

\[
= \sum_{i=1}^{K} z_i(1) \log (\pi_i) + \sum_{i=1}^{K} \sum_{j=1}^{K} \sum_{t=1}^{T-1} \phi_{i,j}(t+1) \log (A_{i,j}) + \sum_{i=1}^{K} \sum_{t=1}^{T-1} \gamma_i(t+1) \log f_Y(y_{t+1}|X_t; \Phi_{t+1,i}),
\]

(4.3)

where \(f_Y(y_{t+1}|X_t; \Phi_{t+1,k})\)\(^1\) is the joint density function from Equation (4.1), conditioned on current observations \(X_t\) through the set of generalised regression models as defined in Equation (4.2). The time- and state-indexed quantities \(\{\{\phi_{i,j}^{[m]}(t)\}_{t=2}^{T}\}_{i=1}^{K}\) and \(\{\gamma_{i}^{[m]}(t)\}_{t=1}^{T}\) are the sufficient statistics which, during each round, are inferred by Equation (3.42). The Markov-switching (HMM) parameters \(\pi, A\) are the initial state probability vector and transition density matrix; these are updated during each round of the EM using Equation (3.43) and Equation (3.44).

The GamBoostLSS learning algorithm presented in Algorithm 6 of the previous chapter requires additional information for a bivariate extension. The additional steps only apply to the inner cycle of the algorithm, which would be equivalent to that part of the M-step of the Baum-Welch algorithm for HMMs that updates the emission densities. This part of the M-step only concerns the maximisation of the section of the CDLL that is dependent on \(\Phi_{t,k}\) for \(K = 1 : K\). That is, the problem reduces to the maximisation of

\[
\sum_{i=1}^{K} \sum_{t=1}^{T-1} \gamma_i(t) \log f_Y(y_{t+1}|X_t; \Phi_{t+1,i}),
\]

(4.4)

\(^1\) The conditional parameter estimates \(\Phi_{t+1,k}\) are forecast using the information set at time \(t\) containing exogenous variables \(X_t\).
Once the inner cycle is called, the E-step has inferred a distribution over the hidden states. This is in the form of a set of observation weights for each state $\gamma_k$. The maximisation of Equation (4.4), i.e. the $K$ state-densities, is then achieved with a weighted form of the gradient boosting algorithm for GAMLSS (GAMBoostLSS), as first introduced in Hofner, Mayr and Schmid (2014) and described in Section 3.2.3.

Prior to introducing the modified MS-GAMLSS for bivariate normal state densities, the component-wise boosting procedure requires the specification of the candidate component set $H$. Recall from Section 3.2.2 that a component is a pair $h_j = (x_j, h)$, where $x_j$ is a single covariate (it may be a subset of covariates), and $h$ is a base learner class. The subscript $j$ identifies the particular covariate. The component models (base learners) used throughout the experiments in this chapter are described in Section 4.2.3.

The modified inner cycle for learning the state dependent distributional regression models follows the steps in the box below.

Call the inner algorithm for each $k = 1 : K$

1. For state-dependent model $k$, set $m = 0$, and initialise the $P = 5$ parameter predictors

$$\Phi_t = \{\eta_{\mu_1}^{[0]}, \eta_{\mu_2}^{[0]}, \eta_{\eta}^{[0]}, \eta_{a}^{[0]}, \eta_{b}^{[0]}, \eta_{c}^{[0]}\}$$

with offset values. We use the weighted sample mean for the mean vector, and the Cholesky factorisation of the weighted sample covariance for the scale and dependence parameters

$$\eta_{\mu_1}^{[0]}, \eta_{\mu_2}^{[0]} \leftarrow \text{weighted\_mean}(y_{1:T}, \gamma_{1:T,k})$$
$$\eta_{a}^{[0]}, \eta_{b}^{[0]}, \eta_{c}^{[0]} \leftarrow \text{Chol}(\text{weighted\_covariance}(y_{1:T}, \gamma_{1:T,k})).$$

a) For $m = 1 : m_{\text{stop}}$ (mth boosting round), where

$$m_{\text{stop}} = \max(M_{\mu_1}, M_{\mu_2}, M_a, M_b, M_c)$$

i. Compute the negative gradients of the complete data log-likelihood with respect to $\Phi_t = \{\mu_{1,t}, \mu_{2,t}, a_t, b_t, c_t\};$ that is, for each $t = 1 : T$

$$-\frac{\partial CDL}{\partial \eta_{\theta,z}(X_t)} = -\frac{1}{k} \frac{\partial \log f_k(y_{t+1}|X_t; \Phi_{t+1,k})}{\partial g^{-1}(\eta_{\theta,z}(X_t))} \frac{\partial g^{-1}(\eta_{\theta,z}(X_t))}{\partial \eta_{\theta,z}(X_t)}$$
$$= -\frac{1}{k} \frac{\partial \log f_k(y_{t+1}|X_t; \Phi_{t+1,k})}{\partial \eta_{\theta,z}(X_t)}.$$
In the standard parameterisation, the loss function is the bivariate normal negative log-likelihood

\[- \log f_Y(y_{t+1}; \Phi_{t+1}) = \log(2\pi) - \log |\Sigma_{t+1}| + (y_{t+1} - \mu_{t+1})^T \Sigma_{t+1}^{-1} (y_{t+1} - \mu_{t+1}),\]

and the covariance matrix is re-parameterised as \(\hat{\Sigma}_{t+1} = \hat{L}_{t+1} \hat{L}_{t+1}^T\), where the entries of \(\hat{L}_{t+1}\) are the conditional estimates of the Cholesky factors

\[
\hat{L}_{t+1} = \begin{bmatrix}
\exp(\eta_{a}^{[m-1]}(X_t)) & 0 \\
\eta_{b}^{[m-1]}(X_t) & \exp(\eta_{c}^{[m-1]}(X_t))
\end{bmatrix},
\]

obtained from the predictor functions \(\eta_{\theta}^{[m-1]}(X_t)\) for \(\theta \in \{a, b, c\}\). Likewise, \(\hat{\mu}_{t+1} = \begin{bmatrix}
\eta_{a}^{[m-1]}(X_t) \\
\eta_{b}^{[m-1]}(X_t)
\end{bmatrix}\) are the conditional estimates of the mean vector. Note that the inverse link function \(g^{-1}(\cdot)\) is omitted in cases where \(\hat{\theta} = \eta_{\theta}^{[m-1]}(X_t)\), i.e. the identity link is used. The analytical gradients for this loss function are all available, and defined as follows

\[
\frac{\partial f}{\partial \mu_t} = -\Sigma_t^{-1} (y_t - \hat{\mu}_t),
\]

\[
\frac{\partial f}{\partial L_t} = \left( \Sigma_t^{-1} - \Sigma_t^{-1} (y_t - \hat{\mu}_t)(y_t - \hat{\mu}_t)^T \hat{\Sigma}_t^{-1} \right) L_t,
\]

with \(\frac{\partial f}{\partial \mu_t} = \begin{bmatrix}
\nabla_{\mu_1}^t \\
\nabla_{\mu_2}^t
\end{bmatrix}\) and \(\frac{\partial f}{\partial L_t} = \begin{bmatrix}
\nabla_{\eta_1}^t & 0 \\
\nabla_{\eta_2}^t & \nabla_{\eta_3}^t
\end{bmatrix}\) giving the desired negative gradients.

A. Following the cyclical updating scheme in Mayr et al. (2012), and as described in (Equation (3.14)), cycle through the parameter set and evaluate the negative gradients for all \(t\) given the most recent estimates of the conditional parameters and covariate vector \(X_t\). That is, we evaluate at \(\hat{\eta}_{\theta}^{[m]}(X_t)\) if the parameter has already been updated in the current round \(m\) and at \(\hat{\eta}_{\theta}^{[m-1]}(X_t)\) if the parameter has not yet been updated or if \(\theta\) is the parameter we are in the process of updating. A parameter is not updated if \(m > M_\theta\).

B. Fit each of the components in \(H\) (see Section 4.2.3) to the negative gradient vector to get a vector of \(J\) regression estimators. For each \(h \in H\)

\[
h^*_j := \arg\min_{h_j} \left[ \sum_{t=1}^{T} (\nabla_{\eta_1}^t - h_j(X))^2 \right],
\]

\(H^* = \{h^*_1(X), ..., h^*_J(X)\}\).
C. Choose from the resulting set $H^*$ the best-fitting base learner, denoted as $h^{[m]}$, and update by adding the component to the predictor function scaled by the shrinkage factor (step size) $\nu_\theta$:

$$
\eta_\theta^{[m]} = \eta_\theta^{[m-1]} + \nu_\theta h^{[m]}(X),
$$

$$
\hat{\theta} = g_\theta^{-1}(\eta_\theta^{[m]}(X)).
$$

D. Repeat until all predictors for the parameters in $\Phi$ have been updated.

ii. Repeat the cycle until maximum boosting iterations has been reached.

2. Return the new conditional state-dependent model $k$ to the outer cycle.

The EM algorithm only guarantees convergence to a local minimum, which can be a problem for these types of latent variable models, as their goodness of fit is sensitive to the initialisation. For all models built for later experiments, each model was built 20 times using different sets of initial values. These 20 runs were split into four groups, each with a different number of states $K = 2, 3, 4, 5$, then the best model was chosen, based on the cross-validated BIC.

### 4.2.3 Base Learners

There are several types of base learners that can be used to build the additive prediction functions in the above models; see Section 3.2.2 for an explanation of how the set $H$ of (base learners, covariate) components is specified. In summary, the set $h$ consists of a collection of pre-defined pairs $h_j = (h, x_j)$ where $h$ is the type of base learner, i.e. a decision stump, linear model, or p-spline, etc., and $x_j$ is the specific covariate that $h$ is a function of. In the experiments of this chapter, two classes of base learner are used, namely (i) simple linear models for capturing any linear effects between covariates and response, and (ii) penalised basis splines (p-splines) for capturing non-linear effects, as described in Section 3.1.4. A component is defined on every variable in $X$ with both types of base learner, making the number of candidate components twice the size of the number of exogenous variables.

In the case of p-spline base learners, a smooth effect of an exogenous variable is modelled as a linear combination of b-spline functions on a fixed set of equidistant knots. These base learners include a penalty to control the wigglyness of the function, see Eilers and Marx (1996) for a formal definition and detailed description. The p-spline is a parametric estimator, with parameters that include the number of knots, degrees of freedom (penalty) and order of polynomial basis. Fortunately there are rules of thumb that can be used to reduce the number of these hyperparameters that need to be selected. This include setting the number of knots, with 40 being considered appropriate for most applications, as was suggested in Eilers and Marx.
(1996), who also proposed that the order of polynomial basis should be set to a compensating high value (seven was chosen here), with the degrees of freedom now being the only parameter used to control wiggyness. This leaves only one parameter to estimate, the penalty term, which was chosen by cross-validation.

4.2.4 Model Selection

The model selection process for a gamBoostLSS-built MS-GAMLSS model is needed to determine the number of boosting iterations required to build each of the parameter prediction functions for each of the state-dependent GAMLSS models, as well as choosing the size of penalty for the p-spline base learners. In addition, the number of states of the Markov chain must also be selected. Due to the model estimation having an iterative outer procedure, learning the model requires the sequential construction of multiple boosted GAMLSS models, i.e. $K$ models for each round of the EM algorithm. However, there is a difficulty in that in each round of EM the GAMLSS models are learned with a different weighting over the data set. This is problematic when we need to select the optimal number of boosting iterations as this particular hyperparameter is sensitive to the quantity of data used to estimate the model, and different weightings means different amounts of observations. The model selection procedure is made more inconvenient by the GAMLSS’s architecture, which causes the size of the search grid to increase exponentially with the number of distribution parameters.

It is well known that models built with a small number of data points will tend to overfit more easily than those built with a large number of data points. This can be an issue in a Markov-switching framework, as in each round of EM different amounts of data are assigned to each state and so to each model; during the course of the learning procedure, the number of data points assigned to any particular state-dependent model can decrease or increase quite substantially, causing both underfitting and overfitting to occur during the learning process, and hence affecting the model’s ability to generalise.

In an ideal situation we would like to select the number of boosting rounds for each parameter of each model during every iteration of the EM algorithm. Computationally speaking, however, this is not an attractive proposition for a GAMLSS with even a modest number of distribution parameters, and becomes excessively time consuming if we wish to run EM to convergence (30 to 50+ iterations). To work around the above problem it is proposed to reduce the dimensionality of the search grid by intuitively grouping parameters within each state model, i.e. select a single number of boosting iterations for both of the mean models and another for the diagonal elements of the Cholesky factors, a third search being conducted for the off-diagonal factor.
To reduce the number of times a model selection is carried out, a two-stage selection process is proposed.

The first stage selects the hyperparameters of a single-state model, including the number of boosting rounds; this gives a set of hyperparameters that will not necessarily be optimal for the switching model but are, however, close enough to use as an approximation. The selected number of boosting rounds for this single-state model is then treated as a budget of boosting rounds to be shared amongst the $K$-state models. Therefore during every round of EM the proportion of the dataset assigned to each state is computed and the state models receive their proportion of boosting rounds from the total budget. This method has the result of reducing overfitting during the learning process; however, most importantly, it keeps the building process stable.

The second stage of model selection is used to refine the approximation from the first stage and again consists of $K$ independent model selections, one for each state model. The GAMLSS models are built with an 18-fold cross-validation; the training portion of each fold is weighted by the smoothed posterior probabilities and observations in the validation sets are weighted by the predictive posterior probabilities. Combining the weighted loss from all $K$-state models across all folds and penalising for complexity using the BIC, a comparative measure of fitness is obtained. Repeating the process multiple times with different initialisations of the HMM parameters for each $K = 2, 3, 4, 5$, the model with the highest cross-validated BIC is selected.

**FIGURE 4.1:** Model selection (boosting rounds) search grid for the first (left) and second (right) stages.

Figure 4.1 shows an illustration of the search grid used for selecting the number of boosting iterations in the two stages mentioned above. In the first stage a sparse search grid (left) is used...
for all parameters; for the mean models in the second stage a dense grid (right) is constructed, which allows for more detailed search.

As an example, Figure 4.2 shows model selection results for the first stage of boosting round selection, for the model built for a cryptocurrency study in Appendix A.1 of the appendix. For this model, the cross-validated maximum boosting rounds for each of the parameters’ predictor functions are $M_{\mu_1} = M_{\mu_2} = 656, M_a = M_b = 1194, M_b = 1418$. These are based on pre-specified shrinkage factors (learning rates) of $\nu_{\mu_1} = \nu_{\mu_2} = 0.001, \nu_a = \nu_c = 0.0005, \nu_b = 0.0005$. The shrinkage factors were chosen to be small enough to maintain the stability of the algorithm. The search grid for the degrees of freedom of the p-splines was $\{2.1 : 3.3\}$ with increments of 0.3. Finally, $K = 4$ latent states (regimes) were found to be best suited for this particular dataset.

**FIGURE 4.2:** Model selection results for a model built for a side study of cryptocurrency pairs (described below and in Appendix A.1 of the Appendix A.), using an 18-fold cross-validation.
4.3 DATA

As discussed in the introduction of this chapter, the main experiment considers the time-series of U.S. equity prices. However, before describing this dataset in detail, we note that in addition to the asset allocation experiments, a separate side study was carried out applying the bivariate MS-GAMLSS to jointly model the Bitcoin/USD and Ethereum/USD markets. The study investigated the forecasting model’s ability to capture the time-variation in volatility and dependence between the two cryptocurrencies, both in- and out-of-sample. The content of this side study was not deemed to be within the scope of this thesis, therefore has been relocated to the appendix; information on the data used can be found in Appendix A.1. For both datasets (U.S. equities and cryptocurrencies) high-dimensional sets of features were extracted from OHLC (open, high, low, close) prices and trading volume; Section 4.3.2 describes the feature extraction process used for both the U.S. equity and cryptocurrency data with, where necessary, comments as to how the process differs between the datasets.

4.3.1 U.S. Equities Dataset

A cross-section of 40 U.S. equities was selected at random from all qualifying tradable equities listed on the NYSE, Amex and Nasdaq exchanges; the grounds for qualification for this study were that they must have been publicly traded companies since 2006-12-07. A full list of companies, with sector and industry details can be found in Appendix C.1. These companies were randomly paired together, resulting in 20 two-stock pairs (sub-portfolios). The raw data used in the equities experiment is daily open, high, low, close prices and daily volume of shares traded (OHLCV data). The training set spans the period 2006-12-07 to 2017-03-07, containing 2578 daily observations for each asset. The testing period comprises the following 252 observations from 2017-03-07 to 2018-03-07. The data was directly sourced from the relevant exchanges via the Interactive Brokers historical data service. To assist in accurately accounting for transaction costs, Level I top-of-book data was also obtained, containing the daily time weighted average and end of day bid and ask prices for each asset. The raw OHLCV dataset was cleaned using standard methods and transformed into a usable dataset, as described in Section 4.3.2.

2 https://interactive.brokers.github.io/tws-api/historical_data.html
4.3.2 Feature Extraction

From the cleaned OHLCV data, datasets were constructed with a number of features, as will now be described. Define the set \( \{O_t, H_t, L_t, C_t, V_t\}_{t=1}^{T} \) as the time-series of open, high, low, and closing prices, and the trading volume that occurred between the opening and closing times. Note that cryptocurrency markets do not open or close; the terminology in this case is conventional, i.e. the opening price at time \( t + 1 \) is the closing price at time \( t \).

**Remark 4.** Note that as with other decisions made through this chapter, this dataset was intentionally made to be simple and consist of basic features that can be extracted directly from the raw data. We make no attempt to test the explanatory power of these features; instead, it is left to the learning algorithm to determine the ones to use and the best way to use them.

The basic features computed are:

\[
\begin{align*}
x_{t,1} &= \log \left( \frac{H_t}{L_t} \right), \\
x_{t,2} &= \log \left( \frac{H_t}{O_t} \right), \\
x_{t,3} &= \log \left( \frac{C_t}{L_t} \right), \\
x_{t,4} &= \log \left( \frac{C_t}{O_t} \right), \\
x_{t,5} &= \log \left( \frac{H_t}{L_t} \right) - \log \left( \frac{C_t}{O_t} \right),
\end{align*}
\]

and the one-period rates of change

\[
\begin{align*}
x_{t,6} &= \log \left( \frac{C_t}{C_{t-1}} \right), \\
x_{t,7} &= \log \left( \frac{V_t}{V_{t-1}} \right).
\end{align*}
\]

The response variables are

\[
y_t = \log \left( \frac{C_{t+1}}{C_t} \right).
\]

From the basic feature vector \( \mathbf{x}_t = (x_{t,1}, ..., x_{t,7}) \) two exponential moving averages are computed, one at six periods and the other at 12 periods. These moving averages are added to the feature vectors \( \mathbf{x}_{t,8:14} = (\text{ema}_t(x_{1,6}), ..., \text{ema}_t(x_{7,6})) \) and \( \mathbf{x}_{t,15:21} = (\text{ema}_t(x_{1,12}), ..., \text{ema}_t(x_{7,12})) \). Finally we add the differences between the current features \( x_{t,7} \) and their corresponding averages, i.e. \( x_{t,22:28} = x_{t,1:7} - x_{t,8:14} \) and \( x_{t,29:35} = x_{t,1:7} - x_{t,15:21} \), and the differences between the two periods of averages \( x_{t,36:42} = x_{t,8:14} - x_{t,15:21} \).

In the case of the cryptocurrency datasets, the above features are computed for both BTC/USD and ETH/USD and the two datasets combined into a single dataset indexed by time, where each row contains features generated from the OHLCV time series of both assets. This gives an 84-dimensional covariate set and two-dimensional response. For the U.S. equity pairs data, the same method is used for each equity pair, creating 20 datasets.
4.4 DYNAMIC PORTFOLIO STRATEGIES WITH MS-GAMLSS

To test the efficacy of the bivariate MS-GAMLSS as a forecasting model for dynamic asset allocation strategies, a number of strategies will be backtested using the model to generate the inputs for allocation decisions. Continuing along the lines established in the introduction to this chapter, which are to avoid unnecessary complexity, two of the three asset allocation strategies use portfolio decision rules based on simple mean-variance optimisation methods, as were discussed in Chapter 2, with the third, and even simpler, strategy being based on directional forecasts. The time period over which the strategies are run spans one year, starting at $t_0$ and separated into $T = 252$ trading days. At each point in time $t = \{t_0, t_1, t_2, ..., t_T\}$ the bivariate MS-GAMLSS produces a forecast $\mathbf{y}_{t+1}$ of the joint distribution for a pair of asset price returns over the next trading period; this is in the form of one-step-ahead conditional forecasts of the mean vector $\hat{\mu}_{t+1}$ and covariance matrix $\hat{\Sigma}_{t+1}$.

Managing just one portfolio with a single pair of assets would be ill-advised as investing in only two assets does not offer portfolio optimisation methods a sufficient means to properly manage risk, and it is expected that the performances of individual pairs-trading strategies will exhibit a large amount of variability. It is far more prudent, and more realistic, to manage multiple strategies at once so that idiosyncratic risks of the individual pairs-trading strategies can be more effectively diversified. Therefore, the overarching portfolio strategy is one that manages a master portfolio containing all 40 assets, with the assets grouped into 20 pairs, with each pair modelled by a bivariate MS-GAMLSS and with an independent dynamic asset allocation strategy. The master portfolio assigns equal weight across all pairs at the start; during the test, all pairs strategies are 100% invested, at all times, and are self-funding with no capital entering or exiting during the course of the test period. Thus, the master portfolios that will be backtested can be thought of as equal weighted portfolios of pairs-trading strategies.

The three types of pairs-trading strategies (and therefore master portfolios) that use signals from the bivariate MS-GAMLSS model are:

1. **Long-short (LS)** two-asset mean-variance portfolio (MVP) allocation (allows both long and short positions).
2. **Long-only (LO)** two-asset MVP allocation (allows only long positions).
3. **Directional forecast (DF)** two-asset directional forecast strategy.

In addition, results are presented for a static equally-weighted long-only strategy that assigns an equal initial weight to all 40 assets and does not make any adjustments over the course
of the test. This is the benchmark for comparison and is a proxy for the performance of the market in general.

MEAN-VARIANCE OPTIMISATION WITH MS-GAMLSS

The optimal mean-variance portfolio, also known as the tangency portfolio, is a set of weights over $N$ risky assets for which the expected return is maximised while, at the same time, minimising the risk. This is equivalent to maximising the Sharpe ratio.

**Definition 5.** Sharpe Ratio (ex-ante): Given a rate of return of a riskless asset $R_{rf}$ we define the expected excess return of a portfolio $P$ as $E[R_P - R_{rf}]$ and the expected standard deviation of excess returns as $\sigma_P = \sqrt{\text{var}[R_P - R_{rf}]}$. The ex-ante Sharpe ratio is

$$S = \frac{E[R_P - R_{rf}]}{\sqrt{\text{var}[R_P - R_{rf}]}}$$

The Sharpe ratio is also a main metric used to measure a portfolio’s performance; this is obtained by substituting the above expectations with their realised values. It should be noted that the risk-free rate is set to zero in these experiments.

The optimisation problem to find the mean-variance weights, using a 1:1 margin constraint on short positions, is

$$w := \arg\max_w -\frac{1}{2}w'\Sigma w + w'\mu,$$

s.t. $|w'| I = 1$

where $I$ is a column vector of ones.

A long position can be taken in an asset, as usual, by placing a buy order at 100% of the position’s value and holding the asset; however, regulations require that short positions are backed up by capital corresponding to 150% of the value of the position, that is 100% of proceeds from the sale of borrowed stock and an additional 50% of the position’s value to be held as initial margin. To permit trading on margin in this way will mean that short selling can increase total risk-return exposure, and, therefore, that strategies which sell short will benefit from leverage and hence not be self-funding. In other words, this would permit portfolios to be greater than 100% invested. To ensure the backtested strategies are compared in a fair way, the experiments impose a 200% margin on short positions, meaning that both long and short positions are treated equally and leverage is not allowed. This requirement has an effect equal
4.4 DYNAMIC PORTFOLIO STRATEGIES WITH MS-GAMLSS

100
to imposing a constraint on the portfolio weights that the sum of their absolute values be equal
to one; this also reduces any concerns about maintenance margins.

The second strategy is another mean-variance optimal tangency portfolio; however, for this
strategy as implemented here (long-only MVP), a restriction on short selling is imposed. The
long-only strategy has the portfolio constraints $w' I = 1$.

RECOVERING THE PREDICTIVE DISTRIBUTION

To compute the portfolio weights at time $t$, the inputs for the mean-variance optimisation
problem $(\hat{\mu}_{t+1}, \hat{\Sigma}_{t+1})$ are recovered from the predictor functions of the MS-GAMLSS model
as follows,

$$
\hat{\mu}_{t+1} = \sum_{k=1}^{K} \eta_{\mu, k}(X_t) P(s_{t+1} = k | y_{1:t}), \\
\hat{\Sigma}_{t+1} = \sum_{k=1}^{K} \begin{bmatrix}
\exp (2\eta_{a, k}(X_t)) & \exp (\eta_{a, k}(X_t)) \eta_{b, k}(X_t) \\
\exp (\eta_{a, k}(X_t)) \eta_{b, k}(X_t) & \eta_{b, k}(X_t)^2 + \exp (2\eta_{c, k}(X_t))
\end{bmatrix} P(s_{t+1} = k | y_{1:t}) \\
+ \sum_{k=1}^{K} \begin{bmatrix}
\eta_{\mu_1, k}(X_t)^2 & \eta_{\mu_1, k}(X_t) \eta_{\mu_2, k}(X_t) \\
\eta_{\mu_1, k}(X_t) \eta_{\mu_2, k}(X_t) & \eta_{\mu_2, k}(X_t)^2
\end{bmatrix} P(s_{t+1} = k | y_{1:t}) \\
- \sum_{k=1}^{K} \eta_{\mu_1, k}(X_t) P(s_{t+1} = k | y_{1:t}) \begin{bmatrix}
\sum_{k=1}^{K} \eta_{\mu_1, k}(X_t) P(s_{t+1} = k | y_{1:t}) \\
\sum_{k=1}^{K} \eta_{\mu_2, k}(X_t) P(s_{t+1} = k | y_{1:t})
\end{bmatrix}^T,
$$

where $i = 1 : 2$, $K$ is the number of states, and $P(S_{t+1} | y_{1:t})$ is the predictive posterior prob-
ability distribution encoding the one-step-ahead state uncertainty; details of how to compute
$P(S_{t+1} | y_{1:t})$ using Hamilton’s filter, will be explained in , as it is more directly relevant to the
content of the following chapter.

The formula for $\hat{\Sigma}_{t+1}$ comes from the law of total variance; for example if $\pi_k = P(S_{t+1} = k | y_{1:t})$,
then for the univariate case

$$
\sigma^2_{\text{total}} = \sum_{k=1}^{K} \pi_k \sigma^2_k + \sum_{k=1}^{K} \pi_k \mu^2_k - \left( \sum_{k=1}^{K} \pi_k \mu_k \right)^2.
$$

As well as the dynamic portfolio strategies mentioned above, results are also presented for a
strategy (DF pairs strategy, the third and final strategy in the list given earlier) that chooses to
enter positions based on directional forecasts made by the model. In directional forecasting,
the model predicts the one-step-ahead movement in price of the two assets in the portfolio
and triggers a buy or sell signal for each asset depending on whether the model predicts prices will rise or fall over the next time period. For this strategy, an equal proportion of portfolio funds are allocated to each asset.

The portfolio weights for the DF pairs strategy are computed as follows, for $i = \{1, 2\}$:

$$w_i = \frac{1}{2} \text{sign} \left( \sum_{k=1}^{K} \eta_{i,k}(x_t) P(s_{t+1} = k|y_{1:t}) \right). \quad (4.8)$$

TRANSACTION COSTS

An important feature of the experimental design was to ensure that backtesting, as accurately as possible, emulated conditions of live trading. The most important point to consider in achieving this was an accurate re-creation of the costs incurred when trading financial assets. To this end, transaction costs were derived directly from Level 1 market data; a full derivation of the method used, and a detailed discussion of the reasoning behind the method, can be found in Section 6.3 of Chapter 6, as this chapter will deal extensively with the management of transaction costs.

In the design of all the experiments in this thesis, costs are time-varying and are determined independently for each asset in the portfolio. The costs are derived as a function of the current quoted bid-ask prices at the time of trading $t$. Let $P_{t}^{\text{Bid}}, P_{t}^{\text{Ask}} \in \mathbb{R}^2$ denote the vectors of bid and ask prices for the two assets. Then the vector containing the quoted half-spreads is

$$s_{i,t} = 0.5 \frac{P_{t}^{\text{Ask}} - P_{t}^{\text{Bid}}}{P_{t}^{\text{Bid}} + 0.5 \left( P_{t}^{\text{Ask}} - P_{t}^{\text{Bid}} \right)}.$$

As we only observe quotes as bid and ask prices, the true prices of the assets are unknown; therefore, the denominator above is the midpoint between bid and ask quotes, assuming the true price is the midpoint of the spread.

Then the transaction costs as a function of the change in allocation are

$$\kappa_t(w_{t|t+1}) = \gamma \left\| s_t + I' c \odot (\hat{w}_{t-1|t} - w_{t|t+1}) \right\|,$$  \quad (4.9)

with $s_t \in \mathbb{R}^2$ the vector of half-spreads, $\gamma \in \mathbb{R}$ a scalar tuning parameter, and finally $c \in \mathbb{R}$ representing any additional proportional costs that are not derived from the bid-ask prices; for the experiments below this is set to $c = 0.0005$ (5 basis points). Finally, it must be noted that $\hat{w}_{t-1|t}$ does not necessarily mean the previously assigned optimal weights $w_{t-1|t}$; rather, it reflects the weightings at the new time $t$, bearing in mind that asset prices will have changed.
4.5 RESULTS

Before discussing the results from backtesting the 40-asset master portfolio strategies, it is of interest to first examine the individual performances of the constituent pairs. Table 4.1 shows a breakdown of the results for each of the 20 individual long-short (LS) and long-only (LO) strategies that used the mean-variance-based trading rules to allocate their allotted funds. The end-of-year realised returns are shown with and without the inclusion of transaction costs; the realised Sharpe ratio is also shown. Each pair is compared to the corresponding results for the equally-weighted (EW) benchmark portfolio, representing the alternative strategy without active trading. The first column identifies the two stocks in each pair by their associated ticker symbols. The last column is the average bid-ask spread for the two stocks over the year and is used to give an indication of how expensive that particular pairs-trading strategy was to run.

The results show that, for most cases, the two mean-variance strategies outperformed the benchmark when transaction costs were omitted, with respect to the returns generated by the individual strategies (15/20 and 13/20 in the LS and LO cases, respectively), and with similar outperformance with respect to the realised annualised Sharpe ratios. When transaction costs are taken into account, however, the high costs involved in the LS strategies are evident when comparing the portfolio returns to the benchmark, only eight of the 20 strategies generating higher gains than the benchmark portfolio. The LO strategy also shows a significant performance impairment when costs are deducted, with 10/20 pairs now beating the benchmark returns. In contrast, the directional forecast (DF) strategy is notably less affected by costs, with 13/20 pairs outperforming the benchmark.

The high cost of running the dynamic LS and LO strategies is clearly responsible for significantly impairing their net performance; however, in some cases, the high costs incurred by running these strategies may be justified, as they are the costs of adapting positions often and optimally, gaining the positive benefit of tracking an optimal asset allocation over time. These benefits can be observed by comparing the realised Sharpe ratios of the individual pairs strategies with the corresponding EW strategies; 14 LS pairs and 12 LO pairs beat the EW benchmark based on the Sharpe ratio, partially due to lower portfolio variance, characteristic of optimum tracking strategies.

Unlike the mean-variance strategies, the directional forecasting strategy makes no attempt to minimise risk during the trading period. Results for the individual component pairs-trading DF
TABLE 4.1: Backtesting results for the 20 equity pairs-trading strategies using the bivariate gaussian MS-GAMLSS model. Results include the two mean-variance strategies (long-short portfolio (LS), long-only (LO)), presented alongside the equal weighted (EW) benchmark portfolio. The strategy returns are presented with and without the inclusion of transaction costs. The final column shows the average daily bid-ask spread in terms of percentage of asset prices. The realised Sharpe ratio presented here is annualised.

<table>
<thead>
<tr>
<th>Asset Pair</th>
<th>Returns (%)</th>
<th>Sharpe Ratio</th>
<th>Spread (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LS</td>
<td>LS (inc TC)</td>
<td>LO</td>
</tr>
<tr>
<td>CRI_TA</td>
<td>56.82</td>
<td>19.83</td>
<td>62.24</td>
</tr>
<tr>
<td>WHA_AC</td>
<td>6.78</td>
<td>1.17</td>
<td>24.76</td>
</tr>
<tr>
<td>AMB_BIG</td>
<td>135.77</td>
<td>61.56</td>
<td>161.89</td>
</tr>
<tr>
<td>RTN_PKI</td>
<td>12.00</td>
<td>5.80</td>
<td>35.14</td>
</tr>
<tr>
<td>AEG_CAKE</td>
<td>43.99</td>
<td>29.53</td>
<td>-10.18</td>
</tr>
<tr>
<td>ALB_KFRC</td>
<td>14.64</td>
<td>1.38</td>
<td>7.43</td>
</tr>
<tr>
<td>CTAS_CRVL</td>
<td>93.88</td>
<td>18.53</td>
<td>102.82</td>
</tr>
<tr>
<td>CTB_CR</td>
<td>5.55</td>
<td>-5.62</td>
<td>-6.43</td>
</tr>
<tr>
<td>FRT_FICO</td>
<td>18.69</td>
<td>5.66</td>
<td>15.68</td>
</tr>
<tr>
<td>GBCI_HNI</td>
<td>-6.40</td>
<td>-22.73</td>
<td>-8.35</td>
</tr>
<tr>
<td>HIBB_IEX</td>
<td>55.64</td>
<td>30.27</td>
<td>100.31</td>
</tr>
<tr>
<td>IIVI_JBHT</td>
<td>24.24</td>
<td>10.28</td>
<td>26.05</td>
</tr>
<tr>
<td>KIM_LGND</td>
<td>61.00</td>
<td>43.81</td>
<td>48.47</td>
</tr>
<tr>
<td>MCD_NWBI</td>
<td>6.68</td>
<td>2.52</td>
<td>8.69</td>
</tr>
<tr>
<td>MDLZ_MINI</td>
<td>15.82</td>
<td>4.87</td>
<td>32.74</td>
</tr>
<tr>
<td>MOS_LH</td>
<td>10.43</td>
<td>6.70</td>
<td>33.48</td>
</tr>
<tr>
<td>OGE_PLCE</td>
<td>0.94</td>
<td>-7.63</td>
<td>-22.35</td>
</tr>
<tr>
<td>ROP_SJM</td>
<td>8.89</td>
<td>2.10</td>
<td>3.67</td>
</tr>
<tr>
<td>SKX_TG</td>
<td>154.17</td>
<td>43.59</td>
<td>92.54</td>
</tr>
<tr>
<td>WRE_AAP</td>
<td>6.68</td>
<td>-0.12</td>
<td>-6.73</td>
</tr>
</tbody>
</table>

strategies are shown in Table 4.2. The returns generated by this strategy outperform the EW benchmark on 13/20 occasions and the Sharpe ratio does so on 16/20 occasions. These results make the DF strategy—surprisingly, given that it is a sub-optimal strategy—the best performing with respect to individual pairs, beating both mean-variance methods and the EW benchmark when transaction costs are considered; possible reasons for this are considered later.

Directional forecasting is in essence a binary classification problem. Hence, with financial time-series forecasting, classification evaluation metrics can be used as one of many objective ways to compare the quality of different predictive models over different data sets, with the caveat that the test sample size should be large enough for the metrics to be considered
4.5 RESULTS

TABLE 4.2: Backtesting results for the 20 equity pairs-trading strategies using the bivariate gaussian MS-GAMLSS model for directional forecasting. Results include the 20 individual directional forecast strategies (DF) with S1 and S2 the contributions of each of the two assets in the portfolio. The additional classification results show the bivariate MS-GAMLSS’s performance forecasting up and down price movements.

<table>
<thead>
<tr>
<th>Asset Pair Symbols</th>
<th>Returns (%)</th>
<th>Sharpe</th>
<th>Classification</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total</td>
<td>S1</td>
<td>S2</td>
<td>Accuracy</td>
</tr>
<tr>
<td>CRAI_TXN</td>
<td>35.77</td>
<td>10.19</td>
<td>25.59</td>
<td>3.06</td>
</tr>
<tr>
<td>WHR_ACIW</td>
<td>-2.25</td>
<td>1.14</td>
<td>-3.39</td>
<td>0.11</td>
</tr>
<tr>
<td>AMSC_BIG</td>
<td>21.09</td>
<td>1.44</td>
<td>19.65</td>
<td>1.42</td>
</tr>
<tr>
<td>RTN_PKI</td>
<td>7.12</td>
<td>1.39</td>
<td>5.74</td>
<td>1.03</td>
</tr>
<tr>
<td>AEIS_CAKE</td>
<td>45.23</td>
<td>28.43</td>
<td>25.59</td>
<td>3.06</td>
</tr>
<tr>
<td>ALB_KFRC</td>
<td>16.24</td>
<td>18.73</td>
<td>-2.50</td>
<td>1.06</td>
</tr>
<tr>
<td>CTAS_CRVL</td>
<td>43.26</td>
<td>22.62</td>
<td>20.64</td>
<td>4.53</td>
</tr>
<tr>
<td>CTB_CR</td>
<td>31.55</td>
<td>10.52</td>
<td>21.04</td>
<td>1.96</td>
</tr>
<tr>
<td>FRT_FICO</td>
<td>11.58</td>
<td>3.52</td>
<td>8.06</td>
<td>1.41</td>
</tr>
<tr>
<td>GBCI_HNI</td>
<td>-5.47</td>
<td>-2.13</td>
<td>-3.33</td>
<td>0.62</td>
</tr>
<tr>
<td>HIBB_IEX</td>
<td>60.61</td>
<td>55.21</td>
<td>5.40</td>
<td>2.01</td>
</tr>
<tr>
<td>IIVI_JBHT</td>
<td>35.16</td>
<td>38.23</td>
<td>-3.07</td>
<td>1.70</td>
</tr>
<tr>
<td>KIM_LGND</td>
<td>32.32</td>
<td>12.08</td>
<td>20.24</td>
<td>2.13</td>
</tr>
<tr>
<td>MCD_NWBI</td>
<td>4.89</td>
<td>18.38</td>
<td>-13.49</td>
<td>0.52</td>
</tr>
<tr>
<td>MDLZ_MINI</td>
<td>18.02</td>
<td>15.14</td>
<td>2.88</td>
<td>1.83</td>
</tr>
<tr>
<td>MOS_LH</td>
<td>-0.39</td>
<td>0.82</td>
<td>-1.21</td>
<td>0.17</td>
</tr>
<tr>
<td>OGE_PLCE</td>
<td>1.73</td>
<td>3.34</td>
<td>-1.62</td>
<td>0.62</td>
</tr>
<tr>
<td>ROP_SJM</td>
<td>15.30</td>
<td>21.15</td>
<td>-5.85</td>
<td>1.39</td>
</tr>
<tr>
<td>SKX_TG</td>
<td>48.99</td>
<td>24.25</td>
<td>24.74</td>
<td>3.30</td>
</tr>
<tr>
<td>WRE_AAP</td>
<td>11.32</td>
<td>13.36</td>
<td>-2.04</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Asymptotically consistent. Classification evaluation results are therefore presented in Table 4.2. For this type of problem there are some rules of thumb that help to interpret the quality of the underlying model’s performance, primarily that an accuracy of greater than 53% is considered very good, while 55% or more is considered exceptional. For roughly balanced data, a kappa score twice as much as the amount by which the accuracy score is in excess of 50% (0.5) indicates that the classification is balanced, good at predicting both buys and sells; for example, an accuracy of 53.5% and a kappa of 0.07 would be a very good indicator that the classifier is performing well.

Aggregated classification results for the 20 DF pairs are shown in Table 4.4. The results are obtained from 10,080 predictions, ensuring a consistent result. The model produced an
excellent accuracy score of 54.5%, which was much higher than what was expected from this model; after all, the MS-GAMLSS is a distributional regression model and not trained as a classifier. A kappa score of 0.073 with 54.5% (0.545) accuracy (compared with a kappa value around 0.09) identifies that there is a slight imbalance. This is confirmed by the precision scores, with 55.7% of buy signals being correct compared to 53% of sell signals. The recall results are even more telling, with 67.4% of all upward price movements being correctly identified by the predictive model, whereas only 39.9% of downward price movements were correctly identified.

Moving on to the main systematic portfolio strategies, in which all 40 assets are traded in portfolios of 20 pairs, Table 4.3 shows the results of backtesting, with an initial wealth of $1, with the cumulative portfolio returns series shown in Figure 4.3; all results are shown net of transaction costs. The table demonstrates that all three strategies (LO, LS and DF) outperform the EW benchmark on both returns and Sharpe ratio; this result is in spite of the individual LS and LO pairs proving unable to convincingly beat the benchmark net of costs. When traded as part of a master portfolio, strategies that used forecasts from the bivariate MS-GAMLSS far outperformed the benchmark (EW) which achieved a realised Sharpe ratio of only 0.84, compared to a 1.3 for LO, 2.3 for LS and 2.9 for DF. Likewise end-of-period wealth from investing in the benchmark portfolio generated a return of only 8.9% compared to the 12.6% for LS, 16.7% for LO and 21.6% for DF.

As mentioned in the introduction to this chapter, portfolios containing only two assets do not have the capacity to effectively diversify away idiosyncratic risks, so that alone these pairs strategies are risky investments. The concurrent trading of multiple pairs by the master portfolio has the effect of diversifying the inherent risks of the individual pairs and substantially reducing the risk of the investment. These effects are most evident in the results of the two strategies that allow for short selling (LS and DF). In both of these cases the Sharpe ratios far exceed those of the LO and EW strategies. Moreover, the results show that the realised standard deviations of portfolio returns for LS and DF are less than half those of LO and EW. The most notable realised performance metric for the LS and DF strategies is the maximum drawdown experienced by these strategies. The maximum loss from a peak to a low for the LS strategy was only 1.79%, with DF second with a maximum loss of 3.0%, while in contrast LO had a maximum loss of 8.5%, and the benchmark EW, at one point, fell 9.3%.

The drawdowns series for the entire test period are plotted in Figure 4.4. The LS portfolio suffers only minor losses, with the DF portfolio, though to a lesser extent, also showing a robustness to negative shocks. The LO portfolio, without a mechanism to protect against systemic market downturns, suffers a string of larger losses. However, the benchmark EW portfolio performs even worse, being completely ill-equipped to deal with market downturns, and for the first
### TABLE 4.3: The main strategy results for portfolios containing the 20 equity pairs, each modelled by a bivariate MS-GAMLSS. (LS = long short, LO = long only, DF = directional forecast and EW = equally weighted benchmark portfolio).

<table>
<thead>
<tr>
<th>Strategy type</th>
<th>Returns (%)</th>
<th>Sharpe ratio</th>
<th>sad</th>
<th>max drawdown (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS</td>
<td>12.58</td>
<td>2.278</td>
<td>0.333</td>
<td>-1.7</td>
</tr>
<tr>
<td>LO</td>
<td>16.65</td>
<td>1.342</td>
<td>0.771</td>
<td>-8.4</td>
</tr>
<tr>
<td>DF</td>
<td>21.60</td>
<td>2.89</td>
<td>0.45</td>
<td>-3.0</td>
</tr>
<tr>
<td>EW</td>
<td>8.85</td>
<td>0.796</td>
<td>0.765</td>
<td>-9.3</td>
</tr>
</tbody>
</table>

### TABLE 4.4: Total (all pairs) classification evaluation metrics for the bivariate MS-GAMLSS directional forecasting strategy.

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th>Kappa</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Buy</td>
<td>Sell</td>
<td>Buy</td>
</tr>
<tr>
<td>Total</td>
<td>54.49%</td>
<td>0.0734</td>
<td>55.78%</td>
<td>53.06%</td>
<td>67.39%</td>
</tr>
</tbody>
</table>

### FIGURE 4.3: Out-of-sample cumulative portfolio returns for the four 40-asset portfolio strategies.

Six months of the test period suffers drawdowns that far exceed those seen in all three other strategies. In addition, the series in Figure 4.3 show that February 2018 experienced a systemic market downturn which LS and (as for the whole test period, to a lesser extent) DF weathered.
without significant losses, whereas the EW portfolio, during that period, fell almost 10% in value.

**TABLE 4.5:** Daily average percentage change in portfolio weights (turnover).

<table>
<thead>
<tr>
<th></th>
<th>LS</th>
<th>LO</th>
<th>DF</th>
<th>EW</th>
</tr>
</thead>
<tbody>
<tr>
<td>min</td>
<td>20.8%</td>
<td>12.9%</td>
<td>10.6%</td>
<td>0.39%</td>
</tr>
<tr>
<td>mean</td>
<td>34.5%</td>
<td>25.4%</td>
<td>23.2%</td>
<td>0.39%</td>
</tr>
<tr>
<td>max</td>
<td>52.2%</td>
<td>43.3%</td>
<td>41%</td>
<td>0.39%</td>
</tr>
</tbody>
</table>

These results raise an interesting question, namely as to why the portfolio returns and Sharpe ratio of the suboptimal DF strategy beat those of the optimal mean-variance LS portfolio. The answer can be found by examining the costs of achieving optimality, i.e. the costs incurred by making adjustments to the LS and LO portfolios every day, in order to track the optimal allocation. Figure 4.5 shows the changing weights of four pairs, for the DF and LS strategies, over the test period. By measuring the cumulative effect of weight adjustments over the time period, a comparative measure of the dynamically of a strategy can be calculated by measuring the distance travelled by the weights vector over time; this can be represented as a percentage of wealth and is known as *portfolio turnover*. The four pairs in Figure 4.5 were chosen on
FIGURE 4.5: Portfolio weights for the LS and DF strategies for a selection of equity pairs. Based on average daily adjustments to the LS portfolio, the figures are ranked from least dynamic (top) to most dynamic (bottom). The dashed line and solid line of each colour identify the weights assigned to each asset in the pair.
4.6 CONCLUSIONS

In this chapter, a Markov-switching generalised additive model for location, scale and shape (MS-GAMLSS) was adapted to model bivariate time-varying distributions of the time-series of asset returns. The forecasts of the models provide the inputs to the trading rules for three different types of systematic pairs-trading strategies. The main experiment in this chapter saw the backtesting of a number of simple portfolios containing 20 actively traded equity pairs. All three strategies outperformed the benchmark equally-weighted portfolio despite mixed results for individual pairs strategies.

However, the regular adjustment to portfolio positions caused by time-variation in the forecasts was shown to generate significant amounts of portfolio turnover, and to therefore incur high trading costs. These costs caused significant impairment in performance for the mean-variance-based strategies. Using the Sharpe ratio as a measure of an investor’s utility, on the basis of these results the use of the optimal mean-variance trading rules, when transaction costs are incurred, is not justified, as the sub-optimal, less costly, directional forecasting (DF) strategy was, net of transaction costs, the best performing strategy. This suggests that tracking an optimal portfolio over time is not necessarily the optimal strategy when transaction costs are taken into account, and that if frequent portfolio rebalancing is part of the trading strategy there is a need for some mechanism within the trading rules to prevent certain transactions taking place, thereby lowering costs by reducing portfolio turnover.

The following chapters address both of the issues identified in these results. Chapter 5 extends the framework introduced here to a full multivariate probabilistic forecasting model, thus allowing for more assets to be included in the component strategies, and therefore for better diversification to be achieved by active optimisation. The portfolio strategies in the upcoming
chapters also improve on the mean-variance method of asset allocation, which nowadays is
deemed to be an overly simplified approximation of an investor’s utility, by considering higher-
order moments of the joint asset returns’ time-series. Chapter 6 addresses the problem of
excessive turnover in dynamic allocation strategies. It will be seen in the results of this chapter
that tracking the optimal allocation over time can have a serious impact on the profitability of
strategies in live trading situations where transaction costs are present; to address this problem,
Chapter 6 will introduce a novel cost minimisation adaptation to the optimisation problem that
considers the cost of portfolio adjustments ex-ante of trading and acts to reduce portfolio
turnover if the cost of adjusting positions outways the benefits of tracking an optimal allocation.
Dynamic Asset Allocation with the Markov-switching Generalised Additive Model for Location, Scale, Shape and Dependence

In this chapter, the bivariate modelling framework of Chapter 4 is extended to the full multivariate case, introducing the Markov-switching generalised additive model for location, scale, shape and dependance (MS-GAMLSSD). The novel method is tailored for the probabilistic forecasting of joint distributions of entire portfolios of financial assets, generating time-varying forecasts for use as input signals for dynamic asset allocation strategies. Rigorous backtesting of multiple portfolios of U.S. equities reveal the economic feasibility of the proposed portfolio management strategies.
5.1 INTRODUCTION

By generalising the concept first introduced by the previous chapter, this chapter presents a flexible multivariate time-series probabilistic forecasting model. The model is specifically tailored for forecasting the joint returns distribution of a portfolio of financial assets for the purpose of providing necessary inputs for tracking an optimal asset allocation in systematic trading strategies. Chapter 4’s findings provided important insights into the expected efficacy and feasibility of pursuing the approach of this chapter by extending the univariate GAMLSS modelling framework to a regime-switching model for forecasting of Markov-switching bivariate normal distributions.

The results from backtesting the bivariate model were encouraging and showed that even within a limited portfolio strategy, where the portfolio was composed of multiple asset pairs-trading strategies, a clear performance advantage was observed over the passive benchmarks. However, mixed results were observed in the performances of the individual asset pairs, which presented as an undesirable amount of variation in the performances suggesting there was insufficient management of risk with two-asset portfolio strategies. This finding, as discussed in the introduction to Chapter 4, was to be expected, as the benefits of diversification would barely be evident with such small portfolios. Extending the model by way of the method in this chapter, to the multivariate case ($d > 2$), allows the study of strategies that have greater capacity for portfolio optimisation methods to mitigate risk by more effective active diversification.

In this chapter, the state-dependent conditional distributions that represent each of the regimes of the regime-switching model are extended from a bivariate gaussian (Chapter 4) to a full multivariate distribution composed of skewed Student’s t marginals, for the univariate behaviour of each asset, and a copula function of the dependence structure, to model the relationships between asset price behaviours. Hence by exploiting copula theory, the multivariate distribution is decomposed into univariate marginals and a copula density function. This decomposition is key to overcoming a major problem encountered in the model selection procedure when choosing hyperparameters of conditional multivariate models. The set of marginal distributions which each represent the price behaviour for each individual asset are now four-parameter conditional density functions. This means that, within each regime specific GAMLSS, for every asset’s time series, there are regression models for forecasting the parameters that determine the mean, variance, skewness and kurtosis of each asset’s one step ahead price returns. To provide maximum flexibility for modelling of the dependence structure, specifically, a pair copula
construction (PCC) was chosen. PCCs achieve their flexibility by allowing for the dependence structure to be constructed from a variety of different types of bivariate copula building blocks. As will be discussed in Section 5.2.2, bivariate copula come in many different forms each with their own specific properties that can capture the different forms of asymmetries and tail dependencies that are found within asset price returns. As was shown with the bivariate normal distribution in the previous chapter, all parameters of the PCC copula are made to be time-varying by conditioning on high-dimensional sets of exogenous variables using the gradient boosting form of the GAMLSS learning algorithm.

The model introduced in this chapter brings together, and will extend, in ways discussed later in the chapter, several major contributions to conditional probabilistic modelling introduced over the past few years. The first such contribution was that of Mayr et al. (2012), which showed how we can apply gradient boosting to learn univariate GAMLSS models; this advancement included the introduction of the gamBoostLSS algorithm, described in Section 3.2.3 of Chapter 3, which permitted high-dimensional sets of explanatory variables to be efficiently employed for building the models, with an added bonus of inbuilt variable selection. Next, as also described in Chapter 3, Langrock et al. (2017) and Adam, Mayr and Kneib (2017) introduced Markov-switching GAMLSS models for univariate responses; Adam, Mayr and Kneib (2017), in particular, extended the gamBoostLSS algorithm to introduce MS-gamBoostLSS (see Algorithm 8), a Markov-switching variant of the original gradient boosting algorithm for GAMLSS. An important contribution that, in the same spirit as GAMLSS, used a collection of generalised additive models (GAMs) to model dependence structures for, first, bivariate and, subsequently, multivariate distributions, was that of Vatter (2016) and Vatter and Nagler (2018). In their work, Vatter and Nagler (2018) showed how a complex dependence structure, represented by a conditional pair copula construction, could be estimated by their GAM-PCC sequential estimation method; they were the first to model a multivariate dependence structure using more than one exogenous variable, though did, however, stress in their conclusion that their method was unable to accommodate high-dimensional sets of covariates and suggested that boosting methods may be better suited. Finally, Bernardi and Catania (2018) and Bernardi and Catania (2019) also need to be mentioned. Their model falls within the generalised auto-regressive score (GAS) framework. They introduced a Markov-switching copula model conditioning the dependence structure on both the Markov chain and a small set of exogenous variables and showed that, when modelling time-varying dependence structures of financial asset returns, conditioning on both exogenous variables and a Markov chain gave better results than conditioning on a Markov chain alone.

The dynamic asset allocation strategies backtested in the previous chapter are also here extended by, as has been mentioned, increasing the number of assets in the portfolios, but also
by improving on the method of optimisation from simple mean-variance rules to a expected utility optimisation that considers higher-order moments; this choice permits the optimisation to consider the extra complexities of the forecasted probability distributions in the asset allocation decisions. To test the application of the models, two experiments were carried out, both involving a total of 40 U.S. equities. The first considers a strategy that trades all 40 assets; however, assets are grouped into smaller five-asset portfolios, each of which are modelled by an MS-GAMLSSD, the portfolio strategy being an equally weighted portfolio of these smaller portfolios, where an investor’s wealth is evenly distributed amongst the sub-portfolios. The second experiment repeatedly draws five assets randomly to form a set of 100 portfolios, the strategy then being run on each portfolio, thus providing insights into the expected behaviour of a single five-asset portfolio. Both experiments use a dynamic portfolio optimisation, where probabilistic forecasts are updated on a day-by-day basis and new portfolio weights computed each day as new information is revealed. In both experiments the model is tested against a selection of active and passive strategies and compared to the benchmarks by using a collection of realised performance / risk metrics.

The structure of the chapter is as follows. Section 5.2 provides the necessary background to copula theory and pair-copula constructions (PCCs), which, as mentioned in Chapter 2, are flexible representations of the dependence structure that underlies a multivariate joint distribution. Building upon the MS-GAMLSS framework of Adam, Mayr and Kneib (2017), described in Chapter 3, Section 5.3 introduces the MS-GAMLSSD model tailored to forecast the joint distribution of a portfolio of financial assets’ returns. Section 5.4 details the two types of experiments conducted to test the quality of the forecasting model. Section 5.5 presents the results obtained, and finally Section 5.6 concludes with a discussion of the results.
Further to the technical background already introduced in Chapters 2 and 3, this section introduces the main mechanism that permits the modelling framework of Chapter 4 to be scaled to arbitrarily high dimensions. This presentation begins in Section 5.2.1 by introducing the main theory of copula (Sklar’s Theorem Sklar (1959)) and shows how a multivariate distribution can be decomposed into a set of univariate marginals and a copula function that represents the dependencies between variables. To illustrate the different types of dependencies that can be handled by copula, Section 5.2.2 then presents examples of copula for bivariate analysis. These bivariate copula are important as they form the building blocks for the pair copula constructions to be introduced later in this section. This is followed in Section 5.2.3 by a brief description of what is considered to be the second most important theorem in copula theory (Patton’s Theorem Patton (2003)), which shows that copula densities can be represented as conditional functions and are therefore suited for time series modelling. The background introducing pair-copula constructions (PCCs) follows in Section 5.2.4, which shows how a complex dependence structure can be constructed from a collection of bivariate copula using a hierarchical graphical model known as a vine. Finally in this background section, Section 5.2.5 discusses some known issues relating to the practical use of PCCs, and shows how these can resolved using an appropriate simplifying condition.

### 5.2.1 Overview of Copula Theory

In the search for flexible ways to represent and model multivariate distributions, copula theory and its applications has become an important addition to many fields of research, and, despite being around for over half a century, has been increasing in popularity over the past few years. The backbone to copula theory is based on a fundamental theorem of Sklar (1959). The theorem establishes the link between univariate marginal distributions and their dependencies. Below, Sklar’s fundamental theorem is briefly explained.

Let \( f \) denote the joint density of a \( d \)-dimensional random vector \( X = (X_1, X_2, \ldots, X_d) \). Then \( f \) can be factorised as

\[
\begin{align*}
  f(x_1, x_2, \ldots, x_d) &= f(x_d) \cdot f(x_1, x_2, \ldots, x_{d-1}|x_d) \\
  &\quad \cdot f(x_1, x_2, \ldots, x_{d-2}|x_{d-1}, x_d) \cdots f(x_1|x_2, \ldots, x_d),
\end{align*}
\]
which is a unique decomposition of \( f \) up to a relabelling of the variables. Copulas offer a way to represent the above density in which the marginal behaviours of the individual variables, represented by \( f(x_i), \forall i=1:d \), are decoupled from the description of the dependence structure.

From Sklar’s theorem, let \( F \) be the multivariate distribution function of the random vector \( X = (X_1, X_2, ..., X_d) \), where each \( X_i \) has a marginal distribution \( F_i \). Then for all \( x = (x_1, x_2, ..., x_d) \) there exists a \( C \) for which

\[
F(x) = C(F_1(x_1), F_2(x_2), ..., F_d(x_d)),
\]

and if all \( F_1..F_d \) are continuous functions then \( C \) is unique. Or, from another perspective, if \( C \) is a copula and \( F_1..F_d \) are distribution functions, then the function \( F \) is the joint distribution of \( X \); therefore the copula \( C \) is itself a distribution, but now over uniformly distributed marginals.

It is important to note that a feature of this result is that the marginals \( F_1..F_d \) are not constrained to be from the same family of distributions, and furthermore that the choice of copula is unconstrained by the choice for the marginals; therefore, using copulas to model the joint distribution of a vector of random variables is a much more flexible way in which to construct multivariate models than using generic multivariate distributions such as a multivariate normal or Student’s t. As a consequence, copula methods permit families of marginal distributions for which no multivariate distribution exists. For example, \( X_1 \) may be gamma distributed, \( X_2 \) normally distributed, and so on. This flexibility is possible because a copula is a function of uniformly distributed marginals obtained from the probability integral transforms of each \( X_i \) using their corresponding cumulative distribution functions \( F_i \).

More formally, let \( U_i = F_i(X_i) \) represent the probability integral transform of \( X_i \). It is known that \( U_i \sim \text{Uniform}(0,1) \) for all \( i \) and so one can rewrite \( C \) as

\[
C(u_1, u_2, ..., u_d) = F(F_1^{-1}(u_1), F_2^{-1}(u_2), ..., F_d^{-1}(u_d)),
\]

where \( F^{-1} \) denotes the inverse of \( F \).
To arrive at the joint density of \( X \), we take the cross partial derivatives of Equation (5.2),

\[
f(x) = \frac{\partial}{\partial u_1 \partial u_2 \ldots \partial u_d} F(x)
\]

\[
= f_1(x_1) \times f_2(x_2) \times \cdots \times f_d(x_d) \times \frac{\partial}{\partial u_1 \partial u_2 \ldots \partial u_d} C(F_1(x_1), F_2(x_2), \ldots, F_d(x_d))
\]

\[
= \prod_{i=1}^{d} f_i(x_i) \times c(F_1(x_1), F_2(x_2), \ldots, F_d(x_d)),
\]

where \( f_1, \ldots, f_d \) are the marginal density functions and \( c \) is the copula density. This important result shows that the joint density can be decomposed into the product of its marginal densities and a copula density function \( c(., .) \). An key result from this decomposition comes instantly from examination of the joint log likelihood. Taking the log of Equation (5.3) we get

\[
\log f(x) = \sum_{i=1}^{d} \log f_i(x_i) + \log c(F_1(x_1), F_2(x_2), \ldots, F_d(x_d)).
\]

Hence, if we wish to estimate the parameters of \( f(x) \) by maximising the above joint log likelihood Equation (5.4), each additive term can be estimated separately. More precisely, the marginal densities can first be estimated independently from each other, and then, once the margin models are built, they can be used to obtain the transforms \( u_i = F_i(x_i), \forall i=1:d \) that are required for estimating the parameters of the copula density. This gives rise to a two-step estimation procedure known as the inference function for margins (IFM) method introduced in Joe and Xu (1996) and Joe (2005). The IFM is an important consideration when estimating multivariate distributions, as if one were to attempt to estimate all parameters simultaneously, intractability would quickly become an issue as the dimensionality increased. Likewise, the IFM method also makes hyperparameter selection during the model validation procedure feasible. For these reasons it is adopted in this study.

For an example of the flexibility of the above described copula decomposition and the benefits it can provide over standard multivariate densities, we can look at models of financial markets. For modelling financial price returns, specifically, the Student’s t distribution is a common choice as it can accommodate for the heavy tails prevalent in the distributions of asset price returns. The degree of freedom parameter of the Student’s t controls the ‘heaviness’ of the tails and characterises the shape of the distribution. However, if using a multivariate Student’s t, all the marginal variables are restricted to be specified by the same degrees of freedom parameter, which makes the assumption that all the asset returns being modelled exhibit identical tail behaviour. This assumption is very restrictive and totally contrary to what has been observed in many empirical studies. For a well-known example of one of these studies, Bollerslev (1987) shows that different exchange rates have different shapes to their distributions. Moreover,
Theodossiou (1998) shows similar results for the generalised skewed version of the Student’s t; in this case, evidence was provided for cross-asset heterogeneity regarding the two shape parameters that determine the skewness and kurtosis of the distributions of financial assets’ price returns. Finally, for more arguments supporting the benefits of the flexibility provided by copula representations for financial returns modelling over using standard (out-of-the-box) multivariate distributions, see Patton (2006).

The next subsection introduces the simplest and most flexible class of copula. These are bivariate copula and used, as the name suggests, for representing the dependence between two random variables. Bivariate copula are interesting to study as there are many different types which can account for a wide range of dependence behaviours; some of the most common types of bivariate copula (that are also used in the model introduced in a later section) are presented in the next subsection. Bivariate copula are what form the building blocks of a PCC, also known as a vine copula, and used for modelling of multivariate dependence. The wide range of bivariate copula available is what makes the PCC representation so powerful when modelling the different types of dependencies seen across asset price returns.

5.2.2 Bivariate Copula

For analysing the statistical behaviour of pairs of random variables, there is a wide variety of different types of copula density functions that can be used to capture the range of possible relationships that can occur. For a rigorous and more formal introduction to bivariate copula the reader is referred to Nelsen (2007). Here, for brevity, only a brief introduction is presented. The six bivariate copula presented in this section are the gaussian, the Student’s t, the Clayton, Frank, Gumbel and the Joe copula. The latter four are Archimedean copula and parameterised by a single parameter. The gaussian copula also has just one parameter, but the Student’s t requires two, one for the level of dependence and one for a degrees of freedom parameter defining the strength of dependence in the tails. The gaussian and Student’s t copulas are known as implicit copula as no closed form exists for their distribution functions, whereas the four Archimedean copulas have a distribution function that can be expressed in closed form. The gaussian and Student’s t copulas are part of a class of elliptical copula, meaning they are copulas of elliptically contoured (or elliptical) distributions. Elliptical copula are restricted to have radial symmetry, which for financial applications is too restrictive an assumption. This is because, in many cases, financial returns exhibit a stronger dependence in the negative tail than the positive, see Patton (2004), Patton (2006), Aas et al. (2009).
By far the most popular class of copula for modelling of data that exhibits asymmetric dependence is the family of Archimedean copula, from which the above mentioned Clayton, Gumbel, Frank and Joe are the most commonly used. What makes these copula differ from one other is that they each capture different types of behaviours in the tails. These differences are illustrated in Figure 5.1, which shows that, despite all examples having the same strength...
of dependence, different copula types can capture very different tail behaviours. It can be seen in the figure that the gaussian and Frank copula do not have any dependence in the upper and lower tails, whereas the Student’s t exhibits both upper and lower tail dependence as defined by the degrees of freedom parameter. The Gumbel and Joe copula both show a stronger upper tail dependence but no lower tail dependence. The reverse is true for the Clayton, where, for two random variables, events that occur in the lower tails of the two distributions will coincide more frequently than events in the upper tails. In Figure 5.1 the marginal distributions of both variables are univariate standard normal, and have a Pearson’s correlation coefficient of $\rho = 0.5$.

The Gumbel copula (a.k.a. Gumbel-Hougard copula) is an asymmetric Archimedean copula, exhibiting greater dependence in the positive tail than in the negative tail. As with all asymmetric Archimedean copula, rotations within the functions allow for this asymmetry to be reversed. Figure 5.2 shows the four possible rotations of the Gumbel copula. The top row contains contour plots of the standard Gumbel and its rotation by $180^\circ$; the latter is also known as the survival Gumbel. These two copula are only defined for positive dependence. For the times where negative dependence is exhibited between two random variables, the related copula that mirror the standard and survival copulas are the $90^\circ$ and $270^\circ$ rotations. Therefore for these single parameter Archimedean copula, there are four sub-types available.

The theory and illustrated examples presented so far in this section have been of simple, unconditional copula. For time-series analysis, however, unconditional models of the dependence structures are far inferior to those that are conditioned on a filtration, or on exogenous information, and therefore of limited efficacy in situations where dependencies vary over time. The next subsection presents the contribution of Patton (2004) and shows how Sklar’s theory is readily extendable to the conditional case.

### 5.2.3 Conditional Copula

After Sklar’s major contribution to multivariate analysis, a second major advancement in copula theory was made in Patton (2003) and Patton (2004). The theorem that was central to Patton’s work showed that the existing theory of unconditional copula could be extended to the conditional case. This advancement meant that practitioners could apply copula theory in the analysis of time-varying conditional dependence, an important addition to financial price returns, exchange rates and risk modelling. In these situations it is well known that many external factors influence the first two moments of the predictive distribution. Allowing the dependence to be
conditional on external factors means that all higher co-moments can vary due to exogenous information.

A brief review of conditional copula follows, for more details see Patton (2003). Similarly to the unconditional case, we have two random variables $X_1$ and $X_2$. Extending to the conditional case requires introducing a conditioning vector $W$. Let $F(X_1, X_2|W)$ denote the conditional distribution of $X_1$ and $X_2$ given $W$, and let the conditional marginal distributions of $X_1|W$ and $X_2|W$ be denoted as $F(X_1|W)$ and $F(X_2|W)$, respectively. Assuming that $F(X_1|W)$, $F(X_2|W)$ and $F(X_1, X_2|W)$ are all continuous for simplicity, then, for all realisations $x = (x_1, x_2)$ and $w \in W$, there exists a $C$ for which

$$F_{X_1, X_2|W}(x|w) = C_{X_1, X_2|W}(F_{X_1|W}(x_1|w), F_{X_2|W}(x_2|w)|w)$$

$$= C(u_1, u_2|w),$$

FIGURE 5.2: Rotations of asymmetric Archimedean copula for both positive and negative dependence.
where \( U \in [0, 1]^2 \) and \( U_1 = F_{X_1|W}(X_1|W = w) \), and \( U_2 = F_{X_2|W}(X_2|W = w) \) (where the realisations of \( U \) are \( u_1 = F_{X_1|W}(x_1|w) \), and \( u_2 = F_{X_2|W}(x_2|w) \) are the conditional probability integral transforms of \( X_1 \) and \( X_2 \)). Furthermore, if all \( F_{X_1|W}, F_{X_2|W} \) are continuous functions then \( C \) is unique. It can be shown that a conditional copula has all the properties of an unconditional copula, see Patton (2003).

As in the unconditional case, the joint likelihood can be obtained as follows,

\[
\ell(x_1, x_2|w) = \frac{\partial^2 F(x_1, x_2|w)}{\partial x_1 \partial x_2}
\]

\[
= \frac{\partial F(x_1|w)}{\partial x_1} \cdot \frac{\partial F(x_2|w)}{\partial x_2} \cdot \frac{\partial^2 C(\partial F(x_1|w), \partial F(x_2|w)|w)}{\partial u_1 \partial u_2}
\]

\[
= f_{X_1}(x_1|w) \cdot f_{X_2}(x_2|w) \cdot c_{1,2|w}(F(x_1|w), F(x_2|w)|w)
\]

\( \forall x_1, x_2, w \in \mathbb{R} \times \mathbb{R} \times W, \)

with the log likelihood given by

\[
\mathcal{L}_{X_1, X_2|W} = \log f_{X_1}(x_1|w) + \log f_{X_2}(x_2|w) + \log c_{1,2|w}(F(x_1|w), F(x_2|w)|w)
\]

\[
= \mathcal{L}_{X_1} + \mathcal{L}_{X_2} + \mathcal{L}_C,
\]

which is the same decomposition as seen in the unconditional case. However, in the conditional case above, we can see that the marginal densities and the copula density are all conditioned on the same conditioning variable \( w \). This is a technical requirement for the distribution \( C \) to be a valid multivariate distribution, but, as will be shown in the last part of this section, not always necessary in practical applications.

### 5.2.4 Pair Copula Constructions

As previously mentioned, there are a variety of bivariate copula for a practitioner to draw upon when modelling dependency between two variables; however, when expanding analysis into higher dimensions, the range of available multivariate copula becomes rather limited, and the flexibility afforded to a practitioner is greatly reduced. *Vine copula*, also referred to as *pair copula constructions* (PCC), represent a way of constructing complex multivariate highly dependent models using bivariate copula as building blocks, thus providing bivariate flexibility to modelling multivariate dependence structures. There are a multitude of real data applications of PCCs, especially in the context of finance. A few notable examples include Brechmann and Schepsmeier (2013), Brechmann and Czado (2013), Dissmann et al. (2013), Loaiza Maya, Gomez-Gonzalez and Melo Velandia (2015).
To recap, from Equation (5.3) a bivariate density function can be written as

\[ f(x_1, x_2) = c_{1,2} \left( F_1(x_1), F_2(x_2) \right) \cdot f_1(x_1) \cdot f_2(x_2), \]

with the conditional density

\[ f_{1\mid 2}(x_1 \mid x_2) = c_{1,2} \left( F_1(x_1), F_2(x_2) \right) \cdot f_1(x_1). \]

Extending this to three random variables

\[ f(x_1, x_2, x_3) = c_{1,2,3} \left( F_1(x_1), F_2(x_2), F_3(x_3) \right) \cdot f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3), \]

the conditional density can be written as

\[
f_{1\mid 2,3}(x_1 \mid x_2, x_3) = \begin{align*}
& c_{1,3\mid 2} \left( F_1(x_1 \mid x_2), F_3(x_3 \mid x_2) \mid x_2 \right) \cdot c_{1,2} \left( F_1(x_1), F_2(x_2) \right) \cdot f_1(x_1) \\
= & c_{1,3\mid 2} \left( F_1(x_1 \mid x_2), F_3(x_3 \mid x_2) \mid x_2 \right) \cdot f_{1\mid 2}(x_1 \mid x_2).
\end{align*}
\]

The above construction is iterative and can be extended to arbitrarily high dimension with the general formula

\[
f_{x \mid y}(x \mid y) = c_{x,y_j \mid y_{-j}} \left( F(x \mid y_{-j}), F(y_j \mid y_{-j}) \mid y_{-j} \right) \cdot f_{x \mid y_{-j}}(x \mid y_{-j})
\]

for a vector \( y \) with \( y_j \) the \( j \)th element and \( y_{-j} \) a subset of \( y \) containing all elements except the \( j \)th. Therefore, an entire joint density can be expressed as the product of a hierarchy or cascade of bivariate copula densities and marginal densities. This construction involves conditional distributions of the form \( F(x \mid y_{-j}) \), which, as was seen earlier, are obtained from the derivative of the copula distribution function,

\[
F(x \mid y_{-j}) = \frac{\partial C \left( F(x \mid y_{-j}), F(y_j \mid y_{-j}) \right)}{\partial F(x \mid y_{-j})}.
\]

This way of constructing a multivariate distribution is not unique; relabelling variables leads to many different ways the hierarchy can be organised. For example, with \( D = 5 \) there exist many unique permutations of the variables; this leads to 240 possible ways the related PCC can be constructed. To organise this structure, we use a graphical model known as a vine. The regular vine (R-vine) was introduced in Bedford and Cooke (2001) and Bedford and Cooke (2002). Vines provide a way to schematically represent the organisation of pair copula constructions, and are graphical models in the form of a sequence of nested trees. The trees are organised hierarchically, where the edges of an upper level become nodes for the level below. These
edges signify specific relationships and are represented by bivariate copula densities. Figures Figure 5.3 and Figure 5.4 illustrate two of the main classes of regular vines, known as the canonical (C-vine) and drawable (D-vine). The differences between C- and D-vines are evident by examining the structure of the graphs. For the five-dimensional case, as seen here, both figures show four levels of trees. For the C-vine, each level has one central node which all other nodes at that level are connected. The D-vine is restricted to have a maximum of two connections per node.

Each of the edges of the graphs represent a bivariate copula density, and the edge label represents the subscript of the density, e.g. edge $3, 5|1, 2$ corresponds to the copula density $c_{3,5|1,2}$. For the five-dimensional example, there exist 60 canonical vine decompositions and a further 60 D-vine decompositions, as well as 120 other possible ways to decompose the copula density not classified as C- or D-vines.

A C-vine has the following density function,

$$c(u_1, u_2, \ldots, u_D) = \prod_{i=1}^{D-1} \prod_{j=1}^{D-i} c_{i,i+j|1:(i-1)} \left( F \left( x_i | x_{1:(i-1)} \right), F \left( x_{i+j} | x_{1:(i-1)} \right) | x_{1:(i-1)} \right),$$  \hspace{1cm} (5.6)
5.2 BACKGROUND

**FIGURE 5.4:** An example of a D-vine construction

where the outer product runs over the $D - 1$ trees, while the inner product refers to the $D - i$ pair-copulas in each tree.

A D-vine has the density

\[
c(u_1, u_2, ..., u_D) = \prod_{i=1}^{D-1} \prod_{j=1}^{D-i} c_{j,j+i|(j+1):(j+i-1)} \left( F \left( x_j | x_{(j+1):(j+i-1)} \right), F \left( x_{j+i} | x_{(j+1):(j+i-1)} \right) \right) ,
\]

(5.7)

where again the outer product runs over the $D - 1$ trees, while the pairs in each tree are designated by the inner product.
With all these potential ways of organising the dependence model, and the variety of bivariate
building blocks that can be considered, there are many possible forms the dependence model
can take. Challenges in deciding the best construction include identifying the most appropriate
vine type for the data, i.e. whether a C- or D-vine is best suited, or if a different organisation of
regular vine is optimal. Another important factor to consider is the order of the factorisation; this
dictates which relationships appear at which levels of the vine. Considerations also include the
types of bivariate copula to use for each edge relationship and the specific parameter values for
each of the pair copula density functions. Thankfully there are a number of efficient algorithms
to perform these tasks through appropriate model selection and estimation; these algorithms
arrive at the best possible construction for the data based on standard comparison criteria such
as likelihood, AIC and BIC. For a detailed introduction to algorithms for model selection and
estimation of PCCs see Aas et al. (2009) or Dissmann et al. (2013). For an implementation in R,
see Brechmann and Schepsmeier (2013), which introduces the popular VineCopula package.
Other estimation methods include kernel density based estimation Nagler and Czado (2015),
estimation using splines (Kauermann and Schellhase (2014) and Schellhase and Spanhel (2018)),
the empirical copula Hoebak Haff and Segers (2012) and, most similar to the methods used
in this chapter, Vatter and Nagler (2018), which introduced an estimation procedure for PCCs
using generalised additive models.

5.2.5 The Simplifying Assumption

The complexity of nested PCC models has been known to be problematic when it comes
to estimation and inference. To illustrate the source of the problem, consider three random
variables $X_1, X_2, X_3$, for which one of the many possible factorisations of the joint density is

$$ f_{1,2,3}(x_1, x_2, x_3) = f_3(x_3) f_{2|3}(x_2|x_3) f_{3|1,2}(x_3|x_1, x_2). $$

The equivalent PCC expansion of the joint density, in terms of bivariate copula densities, can
be expressed as

$$ f_{1,2,3}(x_1, x_2, x_3) = f_1(x_1) f_2(x_2) f_3(x_3) 
\cdot c_{1,3}(F_1(x_1), F_3(x_3)) \cdot c_{2,3}(F_2(x_2), F_3(x_3)) 
\cdot c_{1,2|3}(F_{1|3}(x_1|x_3), F_{2|3}(x_2|x_3)|x_3). \tag{5.8} $$

The top line of the r.h.s. contains terms representing univariate marginal densities, the middle
line shows the two (unconditional) bivariate copula densities at the top of the PCC hierarchy,
and the last line contains the conditional copula density at the next level down. As mentioned
in Section 5.2.3, the fundamental theorem of conditional copula states that for $f_{1,2,3}$ to be a valid joint density, the conditional copula takes the form

$$c_{1,2|3} \left( F_{1|3} (x_1 | x_3), F_{2|3} (x_2 | x_3) \right), \quad (5.9)$$

and represents the relationship between $X_1$ and $X_2$ through conditioning on $X_3$. This conditioning occurs in two ways, indirectly through the arguments (conditional distributions), and directly as a conditioning variable. For the trivariate example in Equation (5.8) there is a maximum of one conditioning variable that appears in the decomposition; however, as the dimensionality increases and more levels of the PCC are necessary, the copula functions at the bottom of the hierarchy are conditional on an increasing number of variables. For many studies of PCCs this has led to problems with tractability. The solution, which trades off statistical efficiency for algorithmic efficiency, is to make a simplifying assumption; when estimating or performing inference on a PCC, it is common to make the assumption that the copula densities are independent of the conditioning variables, except through the conditional distributions. Under this assumption, the copula density in Equation (5.9) becomes

$$c_{1,2|3} \left( F_{1|3} (x_1 | x_3), F_{2|3} (x_2 | x_3) \right),$$

and the joint density $f_{1,2,3}$ is now an approximation. Nevertheless, this approximation has been shown to be a good one, and the simplified PCC tends to be used over the non-simplified PCC. For a discussion of the pros and cons of both versions see Haff, Aas and Frigessi (2010).

5.3 MODEL SPECIFICATION

As mentioned in the introduction to this chapter, the model presented in this section is a multidimensional generalisation of the bivariate gaussian model of the previous chapter. The model specification is tailored for forecasting the joint probability distribution of the price returns of a portfolio of assets. To address the challenges discussed in Chapter 1, all features of the forecasted joint density function are rendered time-varying, with the source of variation coming from two distinct sources. The first source can be considered as a 'global' source of time variation, acting on all assets in the portfolio, and presents in the form of a regime-switching mechanism, thus allowing the predicted distribution to switch between, or more precisely, be represented by, a mixture of multiple market regimes. The second source is a 'local' source of
time-variation, where the regime specific distributions are rendered time-varying through the conditioning of all parameters on exogenous information.

The mechanism of regime-switching follows a first-order Markov process and so remains unchanged from that used for the model introduced in the previous chapter. However, on a local level, the extension to a flexible multivariate framework is achieved with the state-dependent models specified in copula form, with skewed Student’s t margins and a PCC model of dependence. Hence, on a local level, means, covariances, coskewness and cokurtosis of the multivariate distribution are all time-varying. This copula representation permits the use of the inference function for margins method discussed in Section 5.2.1. With the IFM the GAMLSS for the multivariate distribution can be broken into parts, allowing the model to be specified as a set of unrelated univariate GAMLSS models and a separate GAMLSS for the PCC dependence model.

To formally introduce these separate components, for a $D$ dimensional response vector $y_t = (u_{t,1}, y_{t,2}, ..., y_{t,D})^T$, each state $k$ of the $K$ state regime-switching model is represented by a multivariate conditional density function that, in terms of a copula decomposition, can be written as

$$
P(y_t | S_{t-1} = k, \Theta_{t-1}) = f_k(y_t | \Theta_{t-1})$$

$$= c_k(F_1(y_{t,1} | \Theta_{t-1}), ..., F_D(y_{t,D} | \Theta_{t-1}) | \Theta_{t-1}) \prod_{d=1}^{D} f_{k,d}(y_{t,d} | \Theta_{t-1}), \quad (5.10)$$

with $\Theta_{t-1}$ containing all information up until $t - 1$ and $S_{t-1}$ being the state of the Markov chain at the current time step.

As can be seen in Equation (5.10) model specification is split into parts. Section 5.3.1 introduces the GAMLSS specification for a conditional skewed Student’s t distribution, representing behaviours of the univariate margins; Section 5.3.2 specifies the pair-copula construction used for the dependence model; Section 5.3.3 recaps the Markov-switching model for the dynamics of switching between states; and Section 5.3.4 brings all these parts together, arriving at a joint-likelihood for model estimation.
5.3 MODEL SPECIFICATION

5.3.1 The Marginal Models

For each state-dependent multivariate distribution, the $D$-dimensional copula decomposition contains $D$ marginal densities. These are independent of each other and therefore treated as a set of univariate GAMLSS models, each modelled using a skewed Student’s t density, which has the following form for its probability density function,

$$f_{sst}(y; \mu, \sigma, \lambda, q) = \frac{\Gamma\left(\frac{1}{2} + q\right)}{\nu \sigma (\pi q)^{\frac{1}{2}} \Gamma(q) (q(\nu \sigma)^{1/2} (\lambda \text{sign}(x-\mu+m)+1)^q + 1)^{\frac{1}{2}+q}},$$

$$m = \frac{2\nu \sigma \lambda q^{\frac{1}{2}} \Gamma\left(q - \frac{1}{2}\right)}{\pi^{\frac{1}{2}} \Gamma\left(\frac{1}{2} + q\right)},$$

$$v = q^{-\frac{1}{2}} \left[ (3\lambda^2 + 1) \left(\frac{1}{2q + 2}\right) - \frac{4\lambda}{\pi} \left(\frac{\Gamma\left(q - \frac{1}{2}\right)}{\Gamma\left(q\right)}\right)^2\right]^{-\frac{1}{2}},$$

parameterised by the set $\Phi = \{\mu, \sigma, \lambda, q\}$.

Using the above formulation means that the first two parameters of $\Phi$, $\mu$ and $\sigma$, are the mean and standard deviation of the distribution. If we were to instead set $m = 0$ then $\mu$ would represent the mode. The $\lambda$ and $q$ are both shape parameters characterising asymmetry (skewness) and heavy tailedness (kurtosis), respectively; only when $\lambda = 0$ and $q \to \infty$ then $Y \sim N(\mu, \sigma)$, else there is deviation from normality. An important point to note with this distribution is that restrictions on the range of the $q$ parameter need to be imposed in order to ensure that the moments of the distribution are well-defined. That is, the mean will be undefined for $q \leq \frac{1}{2}$, variance will be undefined for $q \leq 1$, skewness will be undefined for $q \leq \frac{3}{2}$ and kurtosis will be undefined for $q \leq 2$.

For a fully conditional probabilistic forecasting model of this distribution, all parameters are made to vary over time by conditioning on an information set. This is achieved by using the standard univariate GAMLSS framework for each of the $D$ univariate marginals, which as mentioned in Section 3.1.4.1 involves modelling each parameter with a GAM prediction function.

Let $k = 1: K$ index a specific state-dependent model from the $K$ distinct states of the Markov model, and $d = 1: D$ index a specific marginal variable from the multivariate distribution; then,
we can define the set of marginal density functions \( \{ f_Y(y_{t,d} | \Phi_{t,d,k}) \}_{d=1}^{D} \) to take the form of Equation (5.11). The sets for the purpose of defining the GAMLSS are

\[
\Phi_{t,k} = \left\{ \hat{\theta}_{t,d,k}, \hat{\lambda}_{t,d,k}, \hat{\mu}_{t,d,k}, \hat{\sigma}_{t,d,k}, \hat{\lambda}_{t,d,k} \right\}_{d=1}^{D},
\]

\[
\Xi_k = \left\{ \eta_{\mu_{d,k}}, \eta_{\sigma_{d,k}}, \eta_{\lambda_{d,k}}, \eta_{q_{d,k}} \right\}_{d=1}^{D},
\]

\[
G^{-1} = \left\{ g^{-1}_\mu(\eta_{\mu}), g^{-1}_\sigma(\eta_{\sigma}), g^{-1}_\lambda(\eta_{\lambda}), g^{-1}_q(\eta_{q}) \right\}.
\] (5.12)

In this \( \Xi_{d,k} \) are the sets of additive predictors associated with the parameters of the \( D \) marginal densities from state distribution \( k \); and \( G^{-1} \) is the set of inverse link functions, which is the same for all \( D \) marginals and \( K \) states. Elements of these sets are related by \( \theta_{t,d,k} = g^{-1}_\theta(\eta_{\theta_{t,d,k}}(X_{t-1,\theta,d})) \) for \( \theta \in \Phi_{t,d,k} \), where the indices \( \theta \) and \( d \) on the conditioning variables \( X_{t-1,\theta,d} \) indicate that \( X \) can be different for each marginal response variable and parameter, but remain the same across the different states. To avoid clutter, the indices on \( X \) are dropped; the state model index \( k \) and margin index \( d \) are also dropped more widely for this same reason, and we proceed with the model definition for a single univariate margin.
For each of the univariate margins $d = 1 : D$, distributed by Equation (5.11), the GAMLSS model is defined by the following set of equations,

\[
\begin{align*}
\hat{\mu}_t &= g_{\mu}^{-1}(\eta_{\mu}(X_{t-1})) , \\
\hat{\sigma}_t &= g_{\sigma}^{-1}(\eta_{\sigma}(X_{t-1})) , \\
\hat{\lambda}_t &= g_{\lambda}^{-1}(\eta_{\lambda}(X_{t-1})) , \\
\hat{q}_t &= g_{q}^{-1}(\eta_{q}(X_{t-1})) ,
\end{align*}
\]

with all predictor functions in $\Xi_d$ taking the form

\[
\eta_{\theta}(X_{t-1}) = g_{\theta}\left(\hat{\theta}_t\right) = \eta_{\theta}^{[0]} + \sum_{m=1}^{M_{\theta}} f^{[m]}(X_{t-1}),
\]

\[\forall \theta \in \Phi_t.\]

In this $\eta_{\theta}^{[0]}$ are constants, $M_{\theta}$ is the number of additive models in the ensemble, and the functions $f^{[m]}$ are determined during model estimation.

Finally, to complete the specification of the GAMLSS for the density defined in Equation (5.11), the link functions and their inverses (set $G^{-1}$) that map the output of the predictor functions to the parameter values and back again are defined below. These functions are the same for all $d$ and $k$ and have the forms

\[
\begin{align*}
g_{\mu}^{-1}(\eta_{\mu}) &= \eta_{\mu} , & g(\eta_{\mu}) &= \hat{\mu} \\
g_{\sigma}^{-1}(\eta_{\sigma}) &= \exp(\eta_{\sigma}) , & g(\eta_{\sigma}) &= \log(\hat{\sigma}) \\
g_{\lambda}^{-1}(\eta_{\lambda}) &= \tanh(\eta_{\lambda}) , & g(\eta_{\lambda}) &= \tanh^{-1}(\hat{\lambda}) \\
g_{q}^{-1}(\eta_{q}) &= \exp(\eta_{q}) + 2 , & g(\eta_{q}) &= \log(\hat{q} - 2).
\end{align*}
\]

For our MS-GAMLSSD model, there are a total of $D \times K$ of these univariate GAMLSS models that need to be estimated. The chosen model estimation method uses a modification of the gamBoostLSS algorithm of Mayr et al. (2012) from Section 3.2.3, and will be described further in Section 5.3.5.
5.3.2 The Dependence Model

For the conditional model of the dependence structure, a vine copula approach was taken. The pair copula constructions consist of sets of $D(D - 1)/2$ bivariate copula, each with a nested set of vines determining the organisation of the relationship structure. The state-dependent conditional PCCs are each modelled by a GAMLSS using transformed data obtained by the probability integral transforms of the original response variables.

The state-dependent density function for a PCC organised with a D-vine is

$$c_{PCC}(u_t) = \prod_{i=1}^{D-1} \prod_{j=1}^{D-i} c_{j,j+i|u_{i,j}}(F(y_{t,j}|y_{t,u_{i,j}},\Phi_{t,j}), F(y_{t,j+i}|y_{u_{i,j}},\Phi_{t,j+i}) | z_{t,v_{j,j+i}}), \quad (5.13)$$

where the subset of conditioning variables is indexed by the vector $u_{i,j} = (j + 1) : (j + i - 1)$. The conditional parameter $z_{t,v_{j,j+i}}$ is the single parameter for the bivariate copula representing edge relationship $j + i$ from the $i$th tree of a D-vine.

The available bivariate copula are restricted to the most common single parameter copula. These include the Archimedean Clayton, Gumbel, Frank, Joe, together with their $90^\circ$, $180^\circ$ (survival) and $270^\circ$ rotations, as well as the gaussian copula and the independence copula. Therefore, to represent the edge relationships of the vine, there are a possible 18 different types of bivariate copula density functions to choose from. Additionally, only C- and D- vines are considered; thus for the $D = 5$ case, used in later experiments, there are 120 different ways that the PCC can be organised. The reason for restricting the copula types to those with a single parameter was twofold, primarily to limit the number of parameters requiring estimation, and, secondly, to keep the estimation procedure the same for all parameters.

For the GAMLSS, all parameters of the PCC (one for each pair copula) require suitable link functions that are appropriate for the range of values for which they are defined. However, different copula have parameters which can lie within different ranges of values. A useful property of bivariate copula, however, is that there exists a one-to-one relationship between the primary parameter of the copula and a rank correlation measure known as Kendall’s tau; furthermore, there is a one-to-one mapping between the more common Spearman’s rho and the primary copula parameter. With these useful properties available, Vatter and Nagler (2018) showed how we can then reparameterise the conditional copula as a function of its corresponding Kendall’s tau. This means the measure tau can be estimated in place of the copula parameter. This

\[ v_{a,b} = \left( (D(D - 1)/2 - (D - a + 1) * ((D - a + 1) - 1)/2 + b - a) \right) \]

maps the indices of the two marginal variables to an index $v_{a,b} \in \{1 : D(D - 1)/2\}$ for $a, b \in \{1 : D | a \neq b\}$. 

allows all single-parameter pair copula to be treated the same, with the same link function irrespective of their type. This has an interpretability benefit, as a well-known concordance measure such as tau has a more natural interpretation than a copula parameter and can allow for comparison of the dependence strength across all pairs.

The GAMLSS specification for the dependence model is

\[
\Phi_{t,k}^c = \{\hat{z}_{t,j,k}\}_{j=1}^J, \\
\Xi_k^c = \{\eta_{z,j,k}\}_{j=1}^J, \\
\hat{z}_{t,j,k} = Z_{j,k}(\hat{\tau}_{t,j,k}),
\]  

(5.14)

for \( J = \frac{D(D-1)}{2} \) (the number of bivariate copula, the number of parameters of the PCC density, and, therefore, the number of edge relationships of the vine that we will model). The set \( \Phi_{t,k}^c \) parameterises the whole PCC density. Note that each state \( k \) is modelled with its own PCC, and, therefore, the structures of the vines will differ between state-dependent models; this means that index \( j \) will most likely represent a different relationship between a different pair of marginals, depending on the state \( k \) that is being considered.

The predictor functions \( \eta_{z,j,k} \) of each parameter are in the form of a generalised additive model Equation (3.10). These are required for each state-dependent dependence model \( k = 1 : K \) and all copula parameters \( j = 1 : J \). As was the case with the marginal models, the exogenous conditioning variables \( X_{t-1,i,k} \) are different for each conditional parameter \( z_i \). More information about the specifics of conditioning variables is provided in Section 5.4.3. For now, to avoid clutter in the notation, subscripts are dropped.

The link between Kendall’s tau and the set of conditioning variables is

\[
\hat{\tau}_t = g^{-1}(\eta(X_{t-1})) = g^{-1}\left(\beta^{[0]} + \sum_{m=1}^{M_k} f^{[m]}(X_{t-1})\right),
\]

and therefore is a change to the usual way a GAMLSS model is specified. Usually the LHS of the above would be a parameter of a density function; however, parameters are not being estimated directly in this case. Instead, the \( \tau \)'s are rank correlation coefficients which exist in the interval \( \tau \in [-1, 1] \) and therefore we can use a link function for which \( g : (-1, 1) \rightarrow \mathbb{R} \).
Hence, the inverse hyperbolic tangent, also referred to as the \textit{arctanh} function, is well suited as the link function, and is defined as

\[ \eta(X_{t-1}) = g(\tau) = \frac{1}{2} \log \left( \frac{\hat{\tau} + 1}{\hat{\tau} - 1} \right). \]

(5.15)

For cases where the argument \( \tau \) is a measure of correlation, concordance or agreement between two random variables, as it is here, one refers to Equation (5.15) as the \textit{Fisher z-transform} of \( \tau \). A useful property of the Fisher transform is that \( g(\tau) \) is approximately normally distributed, making it an ideal target for a regression model.

The above transform has the inverse \( g^{-1} : \mathbb{R} \rightarrow (-1, 1) \), defined as

\[ \hat{\tau}_t = g^{-1} (\eta(X_{t-1})) = \frac{\exp (2\eta(X_{t-1})) - 1}{\exp (2\eta(X_{t-1})) + 1}. \]

(5.16)

This function is the inverse Fisher z-transform, which has been demonstrated to be appropriate for this problem, see Li and Duan (1989). It is, however, worth noting here that any strictly increasing, differentiable and invertible \( g^{-1} : \mathbb{R} \rightarrow (-1, 1) \) could be used as an inverse link instead. But, as mentioned in Vatter and Chavez-Demoulin (2015), the Fisher z-transform is the closest there is to a canonical link in this context.

Finally, from each forecasted \( \hat{\tau}_{t,j,k} \) the copula density functions are parameterised by a further transformation \( \hat{z}_{t,j,k} = Z_{j,k}(\hat{\tau}_{t,j,k}) \), where \( Z_{j,k} : [-1, 1] \rightarrow Q_{j,k} \) is an invertible mapping from the domain of the dependence measure to the range of values allowed by the parameter of the particular type of bivariate copula. As mentioned previously, the functions \( \left\{ \{Z_{j,k}\}_{j=1}^{J} \right\}_{k=1}^{K} \) are one-to-one mappings that are specific to each copula type, and therefore depend on the type of copula assigned to the edge relationship indexed by \( j \).

\subsection*{5.3.3 The Dynamics Model}

As mentioned at the beginning of this section, the regime-switching component of the model provides a global source of time-variation by switching between \( K \) models of the conditional multivariate densities, as defined in Section 5.3.1 and Section 5.3.2. The dynamics of switching are governed by a Markov switching process which is defined in detail in Section 3.3.3. Each multivariate conditional density is parameterised at time \( t \) with forecasts made with information
up until time $t - 1$; this information set is identified by $\Theta_{t-1}$. The entire data generating process therefore has the following probabilistic interpretation

$$P(y_{1:T}, S_{1:T}|\Theta_{T-1}) = p(s_1) P(y_1|S_1, \Theta_0) \prod_{t=2}^{T} p(S_t|S_{t-1}) P(y_t|S_t, \Theta_{t-1}),$$

**FIGURE 5.6:** Graphical representation of a Markov switching model with exogenous covariates.

with the state-dependent densities $P(y_t|S_t = k, \Theta_{t-1}) = f_k(y_t|\Theta_{t-1})$. Figure 5.6 shows a graphical representation of a Markov switching model with exogenous covariates $\Theta_{t-1} = X_{t-1}$.

### 5.3.4 The Joint Predictive Distribution

Finally in this section, the model can now be expressed in terms of all its conditional parameters; for every discrete point in time $t = 1 : T$, and for each state $k = 1 : K$,

$$\Phi_{t,d,k}^{m} = \left\{ \hat{\mu}_{t,d,k}, \hat{\sigma}_{t,d,k}, \hat{\lambda}_{t,d,k}, \hat{q}_{t,d,k} \right\}_{d=1}^{D},$$

$$\Phi_{t,k}^{c} = \left\{ \hat{z}_{t,1,k}, \ldots, \hat{z}_{t,J,k} \right\}.$$  

Therefore for each state model $k$ there are $4D$ time-varying parameters that constitute the marginal densities and $J = \frac{D(D-1)}{2}$ conditional parameters for the dependence model.

In addition to the above conditional parameters, we now can formally define the MS-GAMLSSD model in terms of a parameter set

$$\mathcal{M} = \left\{ \{\Phi_{k,d}^{m}\}_{d=1}^{D}, \Phi_{k}^{c}\}_{k=1}^{K}, \mathcal{M}_{MS} \right\},$$

### 5.3 MODEL SPECIFICATION
with $\mathcal{M}_{\text{MS}} = \{A, \pi_0\}$ denoting the parameters for a $K$-state Markov-switching model, including a matrix of transition probabilities $A \in \mathbb{R}^{K \times K}$ and the vector of starting probabilities $\pi_0 \in \mathbb{R}^K$; for a full definition see Section 3.3.2. The parameters $\{\Phi^m_k, \Phi^c_{j,k}\}_{k=1}^K$ are all conditioned on sets of exogenous variables and, as mentioned previously, the conditioning variables in each set are unique for each conditional parameter; hence, to precisely define the information set $\Theta_{t-1}$ containing all information up until time $t - 1$, for all parameters, we have

$$\Theta_{t-1} = \left\{ y_{1:t-1}, \left\{ \left\{ X^m_{1:t-1,\theta} \right\}_{\theta \in \Phi^m_{d,k}} \right\}_{d=1}^D, \left\{ \left\{ X^c_{1:t-1,j,k} \right\}_{j=1}^J \right\}_{k=1}^K \right\}.$$

Breaking this down, we have

- $y_{1:t-1}$ is the observed $D$ dimensional sequence up to the point of prediction.
- $X^m_{1:t-1}$ are sequences of observed covariates used for prediction of the marginal densities. For a given $\theta$ and $d$, the covariates are the same across the $K$ states; however, sets are different for each margin $y_d$ and each parameter of each marginal density $\theta \in \Phi^m_{d,k}$.
- $X^c_{1:t-1}$ are sequences of conditioning variables for the conditional parameters of the PCC density $z_j \in \Phi^c_{j,k}$ in Equation (5.13). The conditioning variables are different for each conditional parameter, as each parameter represents a specific edge relationship of the vine; and as vine structures will be different for different states, so too will be the conditioning variables for each $z_j$ across each state (e.g. Section 5.4.3 describes this for datasets used in the later experiments).

Hereafter in the notation, dropping a subscript indicates the entire set indexed by the subscript, e.g. $X^m_{t,d} = \left\{ X^m_{t,d,\theta} \right\}_{\theta \in \Phi^m_{d}}$ (dropping $\theta$).

We can now fully specify the predictive distribution for a one-step ahead forecast in terms of its conditional density function. This includes the time-varying conditional parameter forecasts $\Phi^m_{t+1}$ and $\Phi^c_{t+1}$; the sets of conditioning variables $X^m_{t+1}$ and $X^c_t$, and the information set $\Theta_t$ more generally; the unconditional model parameters $\mathcal{M}_{\text{MS}}$, for the dynamics of regime-switching; and a vector of conditional switching parameters $\hat{\xi}_{t+1|t}$. The forecasted density for time $t + 1$ given information up until $t$ is

$$f_Y \left( y_{t+1|\Theta_t; \hat{M}} \right) = \sum_{k=1}^K \hat{\xi}_{t+1|t,k} c_{\text{PCC}} \left( u_{t+1|1}, X^c_{t+1,k}; \Phi^c_{t+1,k} \right) \prod_{d=1}^D f_{\text{sst}} \left( y_{t+1,d|X^m_{t,d}; \Phi^m_{t+1,d,k}} \right),$$

(5.19)

with skewed Student’s t marginal densities $f_{\text{sst}}(\cdot)$ and copula PCC densities $c_{\text{PCC}}(\cdot)$ parameterised by the time-varying forecasts $\Phi^m_{t+1}$ and $\Phi^c_{t+1}$, as defined in Equation (5.17). The mixing parameter $\hat{\xi}_{t+1|t,k} = P \left( S_{t+1} = k | \Theta_t; \mathcal{M}_{\text{MS}} \right)$ is the predictive posterior probability of being in each state at time $t + 1$ given all information up until time $t$. This can be obtained by using
the forward algorithm in Definition 1; we can run the forward pass to \( t \) and obtain the current filtered posterior probability \( \tilde{\alpha} (t) = P (S_t | \Theta_t) \), then propagate this into the future using transition probabilities, to obtain a forecast of the probability of being in each state at the next point in time \( t + 1 \), hence

\[
\hat{\xi}_{t+1|t} = \tilde{\alpha} (t) A.
\]  

(5.20)

As time elapses, and we arrive at time \( t + 1 \), new information is revealed and the information set is now \( \Theta_{t+1} \). We now want to forecast \( t + 2 \), therefore need to infer the parameter \( \hat{\xi}_{t+2|t+1} \). To do so first requires correcting the forecast of \( \hat{\xi}_{t+1|t} \) with newly revealed information to get \( \xi_{t+1|t+1} = \tilde{\alpha} (t + 1) \). Therefore, with \( y_{t+1} \) now known, we construct a \( K \times K \) diagonal matrix

\[
P_{t+1} = \text{diag} \left( f \left( y_{t+1} | X^m_l, X^c_l; \Phi^m_{l+1,1}, \Phi^c_{l+1,1} \right), \ldots, f \left( y_{t+1} | X^m_l, X^c_l; \Phi^m_{l+1,K}, \Phi^c_{l+1,K} \right) \right),
\]

(5.21)

with the off-diagonals equal to zero and diagonal entries containing the likelihood of \( y_{t+1} \), for each of the \( K \) state-dependent models, also referred to as the soft evidence. Then the correction update to \( \xi_{t+1|t+1} \) is obtained with

\[
\xi_{t+1|t+1} = \frac{\hat{\xi}_{t+1|t} P_{t+1}}{\hat{\xi}_{t+1|t} P_{t+1} I},
\]

with \( I \) a length \( K \) column vector of ones. Then the new forecast of \( \hat{\xi}_{t+2|t+1} \) follows from repeating Equation (5.20). The recursion repeats every time step as new information becomes available; this recursive updating scheme is the well-known Hamilton filter, introduced in Hamilton (1989). The filter allows state uncertainty to be inferred in an online manner. As a new observation in the time-series is revealed, the filter first corrects its forecast of the state uncertainty at that time step, then generates a new forecast for the next datapoint, yet to be revealed, in the series.

The model specification presented in this section contains a vast amount of structural flexibility; this is in part due to the state-dependent densities being represented as a copula decomposition, which decouples the joint predictive density into two types of component model, defined in Section 5.3.1 and Section 5.3.2. Additionally, global time-variation through the dynamic nature of regime-switching, defined in Section 5.3.3, means each state-dependent model is able to specialise in distinct patterns that change over time. In the following section, the methodology of the procedure used to learn the MS-GAMLSSD is presented. The methodology provides an even greater addition to the flexibility of the model by using a component-wise gradient boosting learning algorithm which permits high dimensional conditioning variables to be considered.
5.3.5 Model Estimation

The procedure for learning the model in this case is similar to the procedure used in Chapter 4. It is an iterative process consisting of an inner and outer cycle which repeats until convergence. The outer cycle handles necessary inference to establish the model’s distinct regimes, and the inner cycle estimates the models for the state-dependent conditional density functions. The efficient inference algorithms available within the architecture of hidden Markov models (HMMs) (Section 3.3.2) are used to handle the outer cycle of the algorithm, making this similar to the E-step of the Baum-Welch algorithm for HMMs, which was described in Algorithm 7. Numerical optimisation is also available for this task, such as used in Langrock et al. (2017); however, an EM-based method has the benefit of smoothed inference, requiring fewer iterations, shortening computation time, and establishing better separation of the data into states. This makes the outer cycle of the MS-GAMLSSD estimation procedure identical to that of the bivariate model in the previous chapter; the details of this are available in Section 4.2.2. This section will therefore concentrate on the details of the inner cycle while referring to past sections in case where the material has already been covered.

To recap the basics of the outer cycle, the expectation step of the Baum-Welch algorithm uses the forward-backward algorithm, in Definition 3, to estimate the quantities \( \{ \phi_{t,i,j} \}^T_{t=2} \) and \( \{ \gamma_{t,i} \}^T_{t=1} \) for \( i, j = 1 : K \) and for each time step of the time series. From these quantities, we have a form for the complete data log-likelihood

\[
L \left( Y, \Theta_T; \mathcal{M} \right) = \sum_{i=1}^{K} \underbrace{\gamma_{t,i} \log \left( \pi_i \right)}_{\text{dependent on } \pi} + \sum_{i=1}^{K} \sum_{j=1}^{K} \sum_{t=1}^{T-1} \underbrace{\phi_{t+1,i,j} \log \left( A_{i,j} \right)}_{\text{dependent on } A} + \sum_{i=1}^{K} \sum_{t=1}^{T-1} \gamma_{t+1,i} \log f_Y \left( y_{t+1} | \Theta_{t,i}; \Phi_{t+1,i,j}, \Phi_{t+1,i}^c \right). \tag{5.22}
\]

The inner cycle is analogous to the M-step of the Baum-Welch algorithm; however, in this case, we want to estimate conditional densities. Note that when using the EM approach the mixing parameters \( \gamma \) are very different from the mixing parameters of the predictive density \( \xi_{t+1|t+1} \) as seen in Equation (5.19); although they both serve the same purpose of encoding state uncertainty, the \( \gamma \)s are estimated using information from both the past and future, whereas
5.3 MODEL SPECIFICATION

$\xi_{t+1|t+1}$ is estimated with information from the previous time step only. The top and middle row of Equation (5.22) are dependent on the two parameters of the HMM, which are estimated in the usual way, see Algorithm 7.

The component of the inner cycle that remains to be discussed is the process for estimating the many functions used for prediction of the parameters of the state-dependent densities (the bottom row of Equation (5.22)); these are the sets $\Xi^m$ and $\Xi^c$. To estimate the predictor function for the state-dependent models, a modification to the gamBoostLSS algorithm was required. If we were to use the gamBoostLSS algorithm in the same manner as in Chapter 3, i.e. by specifying a single GAMLSS for a state density, it would cause multiple problems for this multivariate extension; these include problems of stability of the boosting algorithm and intractability of hyperparameter search. However, by using a copula decomposition Equation (5.10) of the joint likelihood, the estimation procedure will involve $D + 1$ separate GAMLSS models for each state-dependent density, one for each univariate marginal density and one for the PCC. As discussed in Section 5.2.1, using the IFM approach for estimating copula, the inner cycle becomes a two-step procedure. In step 1, models for the $D$ univariate marginal densities are estimated in parallel using $D$ (standard) univariate GAMLSS models estimated with a weighted form of the gamBoostLSS algorithm. In the step 2, the data is transformed into uniformly distributed marginals by their probability integral transforms. Using the newly transformed response variables, the GAMLSS model of the PCC density is then estimated using weighted component-wise boosting, as provided by the gamBoostLSS algorithm. The inner workings of this algorithm will not be revisited here; for the standard (unweighted) form of gamBoostLSS, see a detailed description in Section 3.2.3, and for the methodology for estimating the weighted form required for Markov-switching models, see the inner cycle of the algorithm presented in Algorithm 8; additionally, the procedure used here follows the general method of that shown in Section 4.2.2.

The contribution to the log-likelihood for a single datapoint is

$$
\sum_{k=1}^{K} \gamma_{t+1,k} \log f_Y (y_{t+1} | \Theta_{t,k}; \Phi_{t+1,k}^m, \Phi_{t+1,k}^c) = \sum_{k=1}^{K} \sum_{d=1}^{D} \gamma_{t+1,k} \log f_{sst} (y_{t+1,d} | X_{t,d}^m; \Phi_{t+1,d,k}^m) \\
+ \sum_{k=1}^{K} \gamma_{t+1,k} \log c_{PCC} (u_{t+1} | X_{t,k}^c; \Phi_{t+1,k}^c).
$$
Therefore for step 1, the loss functions for the GAMLSSs are derived from the negative log of the density $f_{sst}$ with the form in Equation (5.11) with each observation weighted by its corresponding $\gamma_{t+1,k}$.

Once the models of the conditional densities in step 1 have been estimated, then for all $y_{t+1,d}$ we obtain $u_{t+1,d}$ from

$$u_{t+1,d,k} = F_{sst}(y_{t+1,d} | X_{t,d}^m; \Phi_{t+1,d,k}^m),$$

and can then estimate the PCC model in step 2. The loss function for this model is the negative log of the density $c_{PCC}$ from Equation (5.13), again weighted appropriately by $\gamma_{t+1,k}$.

At the start of every inner cycle, the gamBoostLSS algorithms for the $K \times (D + 1)$ GAMLSS models need appropriate initialisation. This is necessary due to the state uncertainty having changed from the previous round, giving us a new set of observation weights for each state, $W_{1:T} = \gamma_{1:T}$. For the models in step 1, initialisation only requires finding starting (offset) values for the predictor functions. For the skewed Student’s t, this is easily done by using weighted maximum likelihood estimation (WMLE) to get unconditional estimates of the parameters. Then for each state $k = 1 : K$ and corresponding weighting $W_k$ we get $\eta_{\theta,k} = g_\theta(\hat{\theta}_{\text{WMLE},k})$ for all GAM predictor functions in $\Xi_k$.

Initialisation of the model for the PCC is somewhat more involved than for the marginal models. Offsets for the prediction functions are required; however, a change in weights will mean the entire structure of the PCC will, most likely, also need to change, to reflect the revised state uncertainty; consequently, the PCC density function, and therefore the GAMLSS specifications, can change for each round of EM, i.e. each time the inner cycle is called. This is an inbuilt flexibility that allows diversity across regimes to be discovered while the learning procedure progresses and is handled automatically every time the state-dependent PCC models are initialised. As was discussed in Section 5.2.4, the multiple factors to consider when establishing a PCC include: the type of vine (C- or D-); the structure of the vine, i.e. what relationships appear in each level of the hierarchy; the type of bivariate copula to model each relationship (edges of the vine); and, finally, the (unconditional) parameters for the bivariate density functions. The problem of PCC initialisation was overcome by using functionality from the R package VineCopula (Schepsmeier et al. (2015)). The package contains all necessary model selection and estimation functionality to fit an unconditional PCC to weighted data; in addition, likelihood and gradient evaluation is also achieved using functionality from this package. Initialisation of the PCC automatically creates a new specification for the GAMLSS model in terms of the sets in Equation (5.14). This specification includes offset values for the predictor functions $\Xi_k$, i.e. $\eta_{z_j,k} = g(Z_{j,k} (\hat{z}_{j\text{WMLE},k}))$, where $Z_{j,k}$ is the function to map the unconditional parameter
estimate \( \hat{z}_{j,WMLE,k} \) to Kendall’s tau, and \( g \) maps \( \hat{\tau} \) to the real line, as described in detail in Section 5.3.2. Initialisation also includes a bivariate copula type being assigned to each pair, which determines the form of the functions \( Z_{j,k} \).

Finally, before presenting the step-by-step description of the learning procedure for a MS-GAMLSSD model, a quick note on model selection. In Chapter 3, arguments were made that by fixing hyperparameters associated with the base learners (p-splines) to values considered to provide a reasonably high amount of regularisation, and further setting the step size for the boosting algorithm to be low, then only the number of boosting rounds for each predictor function needs to be chosen during the model selection process. Preliminary experiments found that better generalisation was achieved when the individual state-dependent models were able to generalise well individually, as independent predictive models. It is not immediately obvious why this would be the case, but it makes sense that a regime identified in training should also be expected to occur in the future. Hence, to maintain good generalisation of the state-dependent models during the learning process, where the inner cycle is called multiple times, it is necessary to reassess the hyperparameter choice on a regular basis; this is primarily to prevent the model discovering regimes which focus on outlier regions or extreme events which would cause it to derogate from good generalisation.

The steps for estimating the MS-GAMLSSD are as follows:

1. **Initialisation**
   a) Generate a set of random weights \( W = \{ (w_{t,1}, w_{t,2}, \ldots, w_{t,K}) \}_{t=1}^T \) over the \( K \) states, where for each time-indexed observation in the training set, \( w_t \sim \text{multinomial}(K) \).
   b) Initialise the HMM parameters \( \pi \) and \( A \) of the Markov-switching model Section 5.3.3.
   c) Build initial state-dependent models for each \( k = 1 : K \):
      i. For each of the \( D \) GAMLSS models of the marginal densities Section 5.3.1, initialise predictor functions \( \Xi^{[0]} = \{ \eta_{\mu}^{[0]}, \eta_{\sigma}^{[0]}, \eta_{\lambda}^{[0]}, \eta_{p}^{[0]} \} \), for each parameter, with the weighted maximum likelihood estimates, \( \eta_{\theta,d}^{[0]} = g_{\theta} (\hat{\theta}_{WMLE,d}) \).
      ii. Perform model selection to discover the number of boosting rounds for each parameter of each model, see Section 5.3.6.
      iii. Build the \( D \) models for the marginal densities with the weighted gamboostLSS algorithm, using weights vector \( W_k \).
      iv. With the estimated models, obtain the conditional densities

\[
    f_{y,t+1,d|x_{t+1,d},k, \hat{\mu}_{t+1,d,k}, \hat{\sigma}_{t+1,d,k}, \hat{\lambda}_{t+1,d,k}, \hat{q}_{t+1,d,k}}
\]
parameterised by conditional forecasts. From this obtain a probability integral transform for each observation

$$u_{t+1,d,k} = F_{sst}^{-1}\left(y_{t+1,d} | X_{t,d}^m, \hat{\mu}_{t+1,d,k}, \hat{\sigma}_{t+1,d,k}, \hat{\lambda}_{t+1,d,k}, \hat{\gamma}_{t+1,d,k}\right),$$

for \(d = 1 : D\) and \(t = 1 : T\). Then \(u_t = (u_{t,1}, u_{t,2}, ..., u_{t,D})\) become the response variables for training the PCC dependence model.

v. Using the same weights \(W_k\) and the new response variables \(U = (u_1, u_2, ..., u_T)\), select the optimal vine structure and set of pair copulas for each bivariate relationship. This is performed using an MLE approach from the R Package ‘VineCopula’ (Schepsmeier et al. (2015)), using the model selection algorithm found in Brechmann and Schepsmeier (2013). The model selected from this procedure becomes the copula density that is to be modelled with a GAMLSS.

vi. Perform model selection to discover the best number of boosting rounds for each parameter of each model, see Section 5.3.6.

vii. Build the GAMLSS model for the copula density.

2. Start outer cycle (EM algorithm)

a) Infer \(\left\{\{\phi_{t,i,j}\}_{i=1}^K\right\}_{t=1}^T, \left\{\gamma_{t,i}\right\}_{i=1}^T\) from forward-backward algorithm in Definition 3 with soft evidence, Equation (5.21), evaluated with conditional forecasts. Then for each state \(k = 1 : K\):

i. Build the GAMLSS models of the \(D\) marginal densities, using the new weights \(W_k = \gamma_k\):
   A. Compute offset parameter values for univariate margin models with weighted MLE.
   B. Perform model selection for new number of boosting rounds, Section 5.3.6.
   C. Rebuild models for the univariate margins with chosen hyper-parameters.

ii. Transform \(u_{t+1,d,k} = F_{sst}^{-1}\left(y_{t+1,d} | X_{t,d}^m, \hat{\mu}_{t+1,d,k}, \hat{\sigma}_{t+1,d,k}, \hat{\lambda}_{t+1,d,k}, \hat{\gamma}_{t+1,d,k}\right),\) for all \(t = 1 : T\) and build a GAMLSS for the PCC density:
   A. Select new PCC dependence structure and initialise predictor functions with the new weights \(W_k = \gamma_k\).
   B. Perform model selection for new number of boosting rounds, Section 5.3.6.
   C. Rebuild GAMLSS for the PCC dependence model with the chosen hyper-parameters.

iii. Update HMM parameters \(\{\pi, A\}\) using Algorithm 7.

b) Evaluate model and decide if convergence has been reached. If not, repeat (a).

---

**PRACTICAL NOTE**

Using an EM approach allows us to identify, relatively early on in the building process, whether or not the process will eventually result in a model that reflects a good local minimum. This should be evident from about five or so expectation maximisation rounds; for the reasoning behind why we can make such an inference, see Mizrach and Watkins (1999). This early termination allows builds which are not proceeding well to be cancelled early, without the need to expend
computation time. As is always the case with the EM approach, multiple builds are required of each model, with the model with the best cross-validated performance being selected for forecasting over the test data. For the process detailed above, the unorthodox interim cross-validation of boosting rounds explained above means that we may not observe a smooth path to convergence such as, for example, in the case for the Baum-Welsh HMM procedure, and the EM round where convergence criteria are met may not be the best observed during the building process. If this is the case, then the model is rebuilt, using the same initialisation, up to the round that provided the best validation results. For the models used in the experiments of this chapter we observed that 25-35 rounds of EM tended to be required to obtain good models, needing approximately 24-36 hours of computation per model build. For the experiments in this chapter, this could only be achieved by using the HPC cluster provided by UCL Research Computing Services.

### 5.3.6 Model Selection

As explained in the model selection section in the previous chapter Section 4.2.4, due to changes in the weightings of the training data that occur after each round of EM, the number of boosting rounds previously selected may not remain optimal from one round of the outer cycle to the next. Furthermore, as mentioned in the previous section, it was found to be beneficial to maintain good generalisability of the individual state-dependent models in order to avoid fitting to outlier regions of the training set and thereby rendering certain state models ineffective for forecasting. Therefore, regular model selection revisions were needed throughout the building procedure.

For the bivariate gaussian model in the previous chapter, a heuristic approach was presented to limit the computations required to reselect these hyperparameters. The predictor functions were separated into groups based on the type of parameter that they model, then a single value for the number of boosting rounds was selected for each group. Recall, for example, in the bivariate case that the two mean parameters were grouped together, and the optimal number of boosting rounds was selected as one value for both parameters. This had the result of reducing the dimensionality of the search grid and therefore computation time. Model selection was then performed once at the beginning of the building process, and as the weighting over the dataset changed during each EM round, the number of boosting rounds were dynamically assigned to each parameter group of each state based on a function of total weight being assigned to that state. Although by no means optimal, this had the effect of discovering good models throughout the build; once complete, a final model selection was conducted using a dense search grid to further improve generalisability of the state-dependent models.
For a higher dimensional multivariate model this approach would be inadvisable. The one-size-fits-all nature of the approach becomes more and more ineffectively simplistic as the dimensionality increases; therefore, a different method was needed. Thankfully the copula decomposition, IFM estimation method, and vine structure of the PCC all offer ways to overcome the issue of unfeasible search grids that would otherwise be a barrier to performing model selection of this type; for example, a $D$ dimensional multivariate gaussian has a grid complexity of $2D^2(D-1)/2$, meaning, in just two dimensions, a search over 10 values would require 10 million models to be evaluated, with this number exploding exponentially as $D$ increases. From the decomposition of the log-likelihood, model selection is separated into two parts. The first part selects the number of boosting rounds for the univariate margin models and consists of $D$ standard uncomplicated grid searches, and is described in the first of the subsections below. The second part chooses the number of boosting rounds for the dependence model which, with a grid complexity of $D(D-1)/2$, means that a standard grid could not be used; this hyperparameter selection therefore required a different approach, a novel method that substantially reduces the complexity of the search and computation time, which will be presented in the second of the subsections below.

**MODEL SELECTION FOR UNIVARIATE MARGIN MODELS**

Model selection of hyperparameters for the univariate margin models is very straightforward. Each univariate marginal density is modelled by a standard univariate GAMLSS model and therefore all parameters are independent of parameters from all the other marginal models. Therefore standard procedures for selecting the number of boosting rounds can be used. The R package ‘gamBoostLSS’ contains functionality for this purpose.

There were two different types of grid searches used. During the building procedure, interim searches were made over sparse grids in order to reduce computation time. After a satisfactory convergence point had been reached, a dense grid was then used, centred in the region of the previously discovered hyperparameter values.

**MODEL SELECTION FOR DEPENDENCE MODEL**

Restricting the PCC to consider only bivariate copula with one parameter meant that a set of $D(D-1)/2$ hyperparameters required selection. For even a small $D$ this will result in an intractable search grid if a standard search is attempted. Therefore, to overcome this problem, a new heuristic search method was proposed and implemented, which works on the principle of early stopping. Early or optimal stopping is a method familiar in machine learning in which validation loss is monitored during the learning process; when the loss over a validation set
starts to increase, the process is terminated, in order to avoid the model becoming overly specialised to the training set. This is a very simple method when the loss function consists of a single estimated quantity. In the case of a PCC the loss function contains predictions from multiple additive models where estimates are dependent on each other in a complex way. This has the problematic effect that the number of boosting rounds selected for one predictor function can affect the number needed for another to be optimal which, under best practices, would usually require a exhaustive grid search to determine a hyperparameter set that is globally optimal.

The method suggested here selects a set containing the number of boosting rounds for the predictor functions of each bivariate copula parameter without the need to conduct an exhaustive grid search. This alternative scheme substantially reduces computational demand, though at the cost of no guarantee of the global optimum. Nevertheless, because of the design of the scheme, it is assumed that the local solution comes close. The scheme works by exploiting both the independences within specific groups of hyperparameters and the incremental nature of building models with gradient boosting. This allows a series of sequential line searches based on early stopping principles to be performed. Groups of independent hyperparameters are updated one at a time, cycling through groups until a stable solution is found.

The method involves first identifying the groups of parameters of the PCC density, where every parameter of a group is independent of all other parameters in the same group; that is, groups of parameters for which a change in one parameter’s value does not cause a change in the gradient evaluation with respect to any other member in the parameter group. Specifically, this means that the number of boosting rounds selected for the predictor function of one group member will not affect the number chosen for any of the others. A way to group the parameters of the PCC density in this way can be found by examining the hierarchical structure of the vine graphical model. Referring back to the graphical representation of a D-vine in Figure 5.3 we can see that the trees are organised hierarchically, where the edges of an upper level become the nodes of the level below. Hence, the edges on the top level, represented by a set of bivariate copula, are not dependent on relationships on any other level, nor are they dependent on each other, and therefore the parameters associated to these bivariate copula are assigned to the first group. For the level below, it can be seen that although these relationships are dependent on those of the level above, importantly, there is again independence between all relationships within this level. Therefore, parameters associated to edge relationships on the second level make up the second group, and so on, descending the vine, grouping parameters based on the level in which the edge relationship they are associated with appears in the vine. This means the grouping and ordering of groups is both intuitive and follows naturally from the structure of the vine.
5.3 MODEL SPECIFICATION

For the $D = 5$ example of a D-vine graphical model in Figure 5.3, the grouping and order of the groups in terms of pair copula densities, reading left to right with the first group representing the top level, and the last representing the bottom level of the vine, are

$$\left\{c_{1,2}, c_{2,3}, c_{3,4}, c_{4,5}\right\}, \left\{c_{1,3|2}, c_{2,4|3}, c_{3,5|4}\right\}, \left\{c_{1,4|2,3}, c_{2,5|3,4}\right\}, \left\{c_{1,5|2,3,4}\right\}. \quad (5.24)$$

The groups in terms of parameters of the PCC are

$$\left\{z_1, z_2, z_3, z_4\right\}, \left\{z_5, z_6, z_7\right\}, \left\{z_8, z_9\right\}, \left\{z_{10}\right\}. \quad (5.25)$$

As a side note, the above two ways to represent the groupings illustrate the relationship between the indexing of model parameters and the corresponding bivariate copula densities in Equation (5.13).

The groups in Equation (5.24) show that the parameters in a particular group are dependent on parameters in all groups that come before it; therefore, a cyclical scheme is adopted where the model selection procedure cycles through the groups, only performing a search on the hyperparameters in one group while the other groups have hyperparameters set at their most recently selected values.

The selection scheme starts by setting all hyperparameters to zero except for those associated to the GAMs of parameters contained within the first group in Equation (5.25); for these, the number of boosting rounds is set to an arbitrary high number, chosen to be higher than required. Here 1500 rounds were selected; from preliminary test builds this allowed for a step length to be chosen for each parameter group such that we would expect to see a selectable minimum in the validation loss before the 1500 round maximum was reached. A GAMLSS model is then estimated; once this is completed, each of the parameters has its predictor function evaluated against a validation set. This is performed one at a time by means of a simple line search over the 1500 boosting rounds.

The boosting round that results in the minimum validation loss is then selected for each member of the group. The process is then repeated for the second group; however, this time, rather than setting all hyperparameters to zero, the number of boosting rounds for parameters in the previous group is set to the recently discovered values. The scheme cycles through all groups multiple times, updating a group at a time while holding the other groups’ hyperparameter values at the most recent choice; this is repeated until the hyperparameter values stop changing.
Let the set \( \{ M_j \}_{j=1}^{10} \) denote the number of boosting rounds used to build the predictor functions of the PCC GAMLSS model; then the model selection process described above can be illustrated schematically,

\[
\begin{align*}
\{ \{ M^{(1)}_{z_1}, M^{(1)}_{z_2}, M^{(1)}_{z_3}, M^{(1)}_{z_4} \} \}, & \quad \{ \{ M^{(0)}_{z_5}, M^{(0)}_{z_6}, M^{(0)}_{z_7} \} \}, \quad \{ M^{(0)}_{z_8}, M^{(0)}_{z_9} \}, \quad \{ M^{(0)}_{z_{10}} \} \} \text{ round 1} \\
\{ \{ M^{(1)}_{z_1}, M^{(1)}_{z_2}, M^{(1)}_{z_3}, M^{(1)}_{z_4} \} \}, & \quad \{ \{ M^{(1)}_{z_5}, M^{(1)}_{z_6}, M^{(1)}_{z_7} \} \}, \quad \{ M^{(1)}_{z_8}, M^{(1)}_{z_9} \}, \quad \{ M^{(1)}_{z_{10}} \} \} \\
\{ \{ M^{(1)}_{z_1}, M^{(1)}_{z_2}, M^{(1)}_{z_3}, M^{(1)}_{z_4} \} \}, & \quad \{ \{ M^{(1)}_{z_5}, M^{(1)}_{z_6}, M^{(1)}_{z_7} \} \}, \quad \{ M^{(1)}_{z_8}, M^{(1)}_{z_9} \}, \quad \{ M^{(1)}_{z_{10}} \} \} \\
\{ \{ M^{(1)}_{z_1}, M^{(1)}_{z_2}, M^{(1)}_{z_3}, M^{(1)}_{z_4} \} \}, & \quad \{ \{ M^{(1)}_{z_5}, M^{(1)}_{z_6}, M^{(1)}_{z_7} \} \}, \quad \{ M^{(1)}_{z_8}, M^{(1)}_{z_9} \}, \quad \{ M^{(1)}_{z_{10}} \} \} \\
\{ \{ M^{(1)}_{z_1}, M^{(1)}_{z_2}, M^{(1)}_{z_3}, M^{(1)}_{z_4} \} \}, & \quad \{ \{ M^{(1)}_{z_5}, M^{(1)}_{z_6}, M^{(1)}_{z_7} \} \}, \quad \{ M^{(1)}_{z_8}, M^{(1)}_{z_9} \}, \quad \{ M^{(1)}_{z_{10}} \} \} \\
\{ \{ M^{(2)}_{z_1}, M^{(2)}_{z_2}, M^{(2)}_{z_3}, M^{(2)}_{z_4} \} \}, & \quad \{ \{ M^{(1)}_{z_5}, M^{(1)}_{z_6}, M^{(1)}_{z_7} \} \}, \quad \{ M^{(1)}_{z_8}, M^{(1)}_{z_9} \}, \quad \{ M^{(1)}_{z_{10}} \} \} \text{ round 2} \\
\vdots & \quad \vdots \quad \vdots \quad \vdots \\
\end{align*}
\]

where \( M^{(n)}_{z_j} \) denotes the selected number of boosting rounds for the model of the \( j \)th PCC parameter after the \( n \)th cycle of the procedure. Convergence can be determined by examining the size of the change in \( M^{(n)} \) from the previous round and weighing up the trade-off between the computational cost of an additional round and the expected improvements to be gained with the expected changes in \( M^{(n+1)} \). This can equally be determined by examining, from one cycle to the next, the change in loss over all validation folds. It was found here that improvements became marginal after the third cycle.

5.4 DYNAMIC PORTFOLIO STRATEGIES

In the two experiments in this chapter, the forecasting performance of the MS-GAMLSSD is evaluated with regards to its effectiveness as a signal generating component of a systematic trading strategy. In both experiments the MS-GAMLSSD is used to forecast the predictive distribution of the joint price returns of a portfolio of U.S. equities over time, with the rolling forecasts providing the inputs into dynamic asset allocation strategies.

To be able to reap some benefits of diversification, while limiting the computation time used for model building, the framework of the MS-GAMLSSD was limited to five dimensions; therefore, each portfolio contained five risky assets, taken from a pool of 40 randomly chosen U.S. equities. The dynamic portfolio strategies manage the five-asset portfolios over one year of test data,
by making asset allocation rebalancing decisions on a daily basis based on rolling forecast updates from the MS-GAMLSSD.

The rebalancing decisions result from the solution to a higher-moment expected utility maximisation problem, based on an investor that displays constant relative risk aversion (CRRA). This allocation strategy was first proposed in Jondeau and Rockinger (2003a) to test the effectiveness of considering probabilistic forecasts in asset allocation problems that consider all moments and co-moments of a joint distribution up to the fourth order.

The strategy tested in the first experiment (Experiment 1) was designed to resemble a portfolio strategy more likely to be seen in real world portfolio management and, therefore, was proposed to test the suitability of the method for use in live trading situations. The proposed strategy actively manages a portfolio of all the 40 U.S. equities, which is achieved by subsetting them into eight smaller portfolios, and allocating the initial wealth equally across all eight. Each sub-portfolio is modelled using the MS-GAMLSSD and independent dynamic asset allocation strategies, as described in the next section, are backtested. The investor does not reallocate wealth from one sub-portfolio to any other during the investment test period; this has the effect of achieving additional benefits of diversification, albeit in a naive way, and is a more realistic approach to portfolio management as the idiosyncratic risk of individual portfolios are removed.

The second experiment (Experiment 2) was proposed to rigorously evaluate the performance of individual five-asset portfolio strategies. To do this, the experiment runs 100 strategies each with five randomly selected assets, and assesses the average behaviour of the portfolios in terms of the range of possible wealth trajectories, along with expected portfolio performance metrics. The aim of this experiment is to expose the entire range of possible outcomes that could be expected when investing in a single five-asset portfolio.

Before introducing the specifics of these experiments, the dynamic asset allocation strategy is presented below; this is in two parts, first the method for obtaining the probabilistic forecasts from the MS-GAMLSSD model is presented, followed by a brief overview of the higher moment optimisation problem whose solution is used to obtain portfolio weights.
5.4.1 Probabilistic Forecasts

The dynamic portfolio strategy requires an objective function to be optimised for the portfolio to be rebalanced; this occurs every day during the investment period, and every optimisation is independent of all others, and hence the dynamic strategy is formed from a series of static optimisation problems. The input to the optimisation objective is a one-step-ahead probabilistic forecast of a portfolio’s asset price returns. This can be represented in the form of the first four central moments of the assets’ conditional joint returns distribution. This subsection shows how these forecasted moments are obtained from the conditional parameter forecasts generated by the MS-GAMLSSD.

The \( D \)-dimensional response vector,

\[
y_{t+1} = (y_{t+1,1}, y_{t+1,2}, \ldots, y_{t+1,D})^T,
\]

represents the one-step-ahead price returns for the assets in the portfolio. The random vector \( y_{t+1} \in Y_{t+1} \) has a joint distribution which can be defined in terms of its first four central moments,

\[
Y_{t+1} \sim Y_{r|t+1} \left( \hat{M}_{t+1}, \hat{\Sigma}_{t+1}, \hat{S}_{t+1}, \hat{K}_{t+1} \right), \tag{5.26}
\]

where for a \( D \) dimensional distribution \( M_{t+1} \in \mathbb{R}^D \), \( \Sigma_{t+1} \in \mathbb{R}^{D \times D} \), \( S_{t+1} \in \mathbb{R}^{D \times D \times D} \) and \( K_{t+1} \in \mathbb{R}^{D \times D \times D \times D} \) are the forecasted mean vector, covariance matrix, and co-skewness and co-kurtosis arrays for the price returns of the assets in the portfolio over the time period \([t, t+1]\). To avoid any confusion over the naming of terms of the aforementioned quantities, it should be noted that the names 'skewness' and 'kurtosis', when used correctly, refer to the third and fourth standardised moments. However, here the quantities \( \hat{S}_{t+1} \) and \( \hat{K}_{t+1} \) actually represent the third and fourth non-standardised central co-moments; the two definitions are closely related but not the same. This is a distinction frequently neglected to be made in the literature but nevertheless a very important one. The elements of the arrays of the second to fourth moments are computed by the following expectations,

\[
\hat{S}_{t+1,i,j} = \mathbb{E} \left[ (y_{t+1,i} - \mu_{t,i}) (y_{t+1,j} - \mu_{t,j}) \right],
\]

\[
\hat{S}_{t+1,i,j,l} = \mathbb{E} \left[ (y_{t+1,i} - \mu_{t,i}) (y_{t+1,j} - \mu_{t,j}) (y_{t+1,l} - \mu_{t,l}) \right],
\]

\[
\hat{K}_{t+1,i,j,l,m} = \mathbb{E} \left[ (y_{t+1,i} - \mu_{t,i}) (y_{t+1,j} - \mu_{t,j}) (y_{t+1,l} - \mu_{t,l}) (y_{t+1,m} - \mu_{t,m}) \right].
\]
At each time point $t$, forecast the set of conditional parameters $\Phi_{t+k}^c$, given current observation $v_t$, and generate $N$ samples from the copula density; this is performed using the R package ‘VineCopula’ (Schepsmeier et al. (2015)).

\[
\tilde{u}_{t+1,k} \leftarrow c \left( \cdot | X_{t}^c; \Phi_{t+1,k}^c \right) \text{ for } k = 1 : K
\]

Likewise, forecast the conditional parameters for the marginal distributions $\Phi_{t+1,d,k}^m$ for each $d$ and $k$. Then, given the sampled realisations $\tilde{u}_{t+1,k} \in [0, 1]^D$, quantile function $F_{Y_{t+1}}^{-1}$ and predictive probabilities $\pi_{t+1}$,

\[
\tilde{y}_{t+1,d} = \frac{K}{\sum_{k=1}^{K} \pi_{t+1,k} F_{Y_{t+1}}^{-1} \left( \tilde{u}_{t+1,d,k} | X_{t,d}; \Phi_{t+1,d,k}^m \right)}.
\]

Compute the sample central moments and co-moments in Equation (5.26) from the set of $N$ simulated observations $\{\tilde{y}_{t+1,d}\}_{i=1}^{N}$. Note the mean vector is given explicitly from the weighted sum of the conditional means of each state distribution $M_{t+1} = \sum_{k=1}^{K} \pi_{t+1,k} \hat{\mu}_{t+1,k}$, and does not need to be approximated by the sample mean.

**Algorithm 9:** A scheme for generating approximations of the co-moments of the price returns distribution using sample estimates.

These expectations, with respect to the joint distribution in Equation (5.19), are able to be obtained using numerical integration methods; however, Bernardi and Catania (2018) found that this approach can be imprecise and slow, and a better method is available through simple Monte Carlo simulation. Therefore, we obtain approximations of the central moments of Equation (5.26) by sampling from the joint predictive distribution of $Y_{t+1}$, that has been parametrised by the forecast generated by the MS-GAMLSSD distributional regression model Equation (5.26). The sampling scheme is presented in Algorithm 9. This scheme includes sampling from a PCC, which is a reasonably involved process, but achievable using available software, and so the details are omitted here. For a full description of how to sample from a vine copula, see Aas et al. (2009).

It is more convenient for the upcoming description of methods to express the moments $S_{t+1} \in \mathbb{R}^{D \times D \times D}$ and $K_{t+1} \in \mathbb{R}^{D \times D \times D \times D}$ as 2D matrices, as has been shown in De Athayde and Flôres Jr (2004) and Bernardi and Catania (2018); this is done by slicing through the layers and pasting each layer onto the previous layer sideways, thus transforming $\mathbb{R}^{D \times D \times D} \rightarrow \mathbb{R}^{D \times D^2}$ for the array $S_{t+1}$ and $\mathbb{R}^{D \times D \times D \times D} \rightarrow \mathbb{R}^{D \times D^3}$ for the array $K_{t+1}$. As will be seen next, this way of representing high-dimensional arrays is very convenient to show the calculation of the portfolio moments needed for the optimisation objective detailed in the next subsection.
5.4.2 Higher Moment Dynamic Portfolio Strategy

The asset allocation strategy actively manages a portfolio of five risky assets and does not include any risk-free assets. The portfolios are rebalanced on a daily basis based on signals from the predictive model. The backtest period is one year, partitioned into \( T = 252 \) days \( t = \{t_0, t_1, t_2, \ldots, t_T\} \), where, at each time \( t \), an investor’s wealth \( W_t \) is allocated across the five assets. This proportioning of wealth, decided at time \( t \), and lasting until time \( t+1 \), is represented by the vector of portfolio weights

\[
\mathbf{w}_{t:t+1} = (w_{1,t:t+1}, w_{2,t:t+1}, \ldots, w_{D,t:t+1})^T,
\]

where each element denotes the proportion of total wealth that is assigned to each asset for the time period \([t, t+1)\).

From the conditional parameter forecasts from Equation (5.26) for the horizon \([t, t+1)\), the previous section showed how to obtain the estimated moments \( \hat{\mu}_{t|t+1}, \hat{\Sigma}_{t|t+1}, \hat{S}_{t|t+1}, \hat{K}_{t|t+1} \) of the joint distribution. Given this information about the expected behaviour of the assets in the portfolio, the problem is to find an optimal allocation of wealth \( W_t \) that satisfies the risk return preferences of the investor. For these experiments, these preferences are represented by a constant relative risk aversion (CRRA) utility function,

\[
U(W_{t+1}; \lambda) = \begin{cases} 
W_{t+1}^{1-\lambda} & \text{for } \lambda > 1 \\
\frac{1}{1-\lambda} \log (W_{t+1}) & \text{for } \lambda = 1
\end{cases},
\]

where \( \lambda \) is the parameter of risk aversion. The end-of-period wealth \( W_{t+1} \) is the initial wealth \( W_t \) compounded by the sum of asset returns weighted by the allocation, i.e. \( W_{t+1} = W_t (1 + \mathbf{w}^T_{t|t+1} y_{t+1}) \).

Because of the properties of a CRRA utility function, we can set initial wealth \( W_t = 1 \) for every \( t \); hence, the dynamic strategy is broken down into a series of static one-period portfolio rebalancing decisions, all with the same initial wealth, irrespective of any previous gains or losses. There is no loss of generality by setting \( W_t = 1 \) for all \( t \), as with CRRA utility, risk aversion does not change with a change in wealth; this choice, importantly, helps by simplifying the objective function, as we can always express the terminal wealth as \( W_{t=1} = 1 + \mathbf{w}^T_{t|t+1} y_{t+1} \).

The objective of a static portfolio optimisation problem is to maximise the expected utility subject to portfolio constraints, that is, to find an allocation \( \mathbf{w}_{t|t+1} \) that satisfies

\[
\mathbf{w}_{t|t+1} := \arg\max_w \mathbb{E} [U(W_{t+1})].
\]
The following chapter examines this optimisation problem in more detail and provides a full derivation of the following approximate solution to Equation (5.28). In brief, the issue is that the expectation of future wealth with respect to the joint distribution Equation (5.19), has no closed form solution; however, Jondeau and Rockinger (2006a), Martellini and Ziemann (2010) and Jondeau and Rockinger (2012) have suggested that a fourth order Taylor series expansion can be used as a good approximation. The Taylor series of Equation (5.28), taken around the point of initial wealth, is

$$
\mathbb{E} [U(W_{t+1})] \approx \sum_{k=0}^{\infty} \frac{U^{(k)}(W_t)}{k!} \mathbb{E} \left[ (W_{t+1} - W_t)^k \right].
$$

(5.29)

Substituting the CRRA utility function Equation (5.27) into Equation (5.29) and taking expectations with respect to the distribution Equation (5.26), the expectation for the $k$th order term becomes

$$
\mathbb{E} \left[ (W_{t+1} - W_t)^k \right] = \mathbb{E} \left[ (w'_{t|t+1} y_{t+1})^k \right] = \mathbb{E} \left[ R_{t+1}^P \right] = m_{k,t+1},
$$

where $R_{t+1}^P$ is the end-of-period portfolio return and $m_{k,t+1}^P$ the $kth$ order non-central moment of the portfolio return distribution. Therefore, the above equality shows the approximation can be expressed as a function of the first four non-central moments of the portfolio returns distribution, which, from the probabilistic forecast in Equation (5.26), are implicitly available for any given $w_{t|t+1}$. That is, given a portfolio allocation $w_{t|t+1}$ and the forecasts from the assets’ price returns distribution, the first four central moments for the portfolio returns distribution are

$$
\hat{\mu}_{P,t+1} = w'_{t|t+1} \hat{\mu}_{t+1},
$$

$$
\hat{\sigma}_{P,t+1}^2 = w'_{t|t+1} \hat{\Sigma}_{t+1} w_{t|t+1},
$$

$$
\hat{s}_{P,t+1} = w'_{t|t+1} \hat{S}_{t+1} \left( w_{t|t+1} \otimes w_{t|t+1} \right),
$$

$$
\hat{k}_{P,t+1} = w'_{t|t+1} \hat{K}_{t+1} \left( w_{t|t+1} \otimes w_{t|t+1} \otimes w_{t|t+1} \right),
$$

(5.30)

where $\otimes$ denotes the Kronecker product and $S_{t+1} \in \mathbb{R}^{D \times D^2}$, $K_{t+1} \in \mathbb{R}^{D \times D^3}$ have been transformed into their equivalent $2D$ array representations.
From Equation (5.30), we can now obtain the non-centred moments using the following transformations,

\[\begin{align*}
    m_{1,t+1}^P &= \hat{\mu}_{P,t+1}, \\
    m_{2,t+1}^P &= \sigma_{P,t+1}^2 + \hat{\mu}_{P,t+1}^2, \\
    m_{3,t+1}^P &= s_{3,P,t+1} + 3\sigma_{P,t+1}^2 \hat{\mu}_{P,t+1} + \hat{\mu}_{P,t+1}^3, \\
    m_{4,t+1}^P &= k_{4,P,t+1} + 4s_{3,P,t+1}^2 \hat{\mu}_{P,t+1} + 6\sigma_{P,t+1}^2 \hat{\mu}_{P,t+1}^2 + \hat{\mu}_{P,t+1}^4.
\end{align*}\] (5.31)

Therefore, the approximate solution to Equation (5.28) and the objective function to be optimised is

\[\begin{align*}
    \mathbb{E}[U(W_{t+1})] &\approx \frac{1}{1 - \lambda} + m_{1,t+1}^P - \frac{1}{2} \lambda m_{2,t+1}^P + \frac{1}{6} \lambda (\lambda + 1) \lambda m_{3,t+1}^P - \frac{1}{24} (\lambda + 2) (\lambda + 1) \lambda m_{4,t+1}^P.
\end{align*}\] (5.32)

Finally, to complete the optimisation problem, a leverage constraint \(\sum_i |w_{i,t+1}| = 1\) is applied to the weight vector. This constraint allows the short selling of assets; however, it also reflects two important restrictions regarding short positions; first, that the proceeds from a short sale are not available for investment, and, second, that additional funds equating to 100% of the value of the short position be held as initial margin\(^2\). This is equivalent to a 200% initial margin on short positions, in excess of regulatory requirements of 150%. This constraint means that long and short positions are treated equally, and leads to more conservative, but fair, portfolio performances by not allowing long-short strategies to benefit from leverage obtained through short selling. As an example, the standard textbook long-short constraint \(\sum_i w_{i,t+1} = 1\) permits proceeds from short sales to be reinvested into the portfolio’s long positions, thus meaning for a dynamic strategy, there will be times when total exposure to risk assets will be greater than 100% of wealth, which will introduce a bias into the results favouring strategies that short the most. Setting the leverage constraint to 1 prohibits leverage of any kind, ensures that portfolio strategies are self-funding and always 100% invested.

\(\text{\footnotesize\(^2\) For an example of U.S. equity margin requirements for short selling, see Interactive Brokers (IBKR) guide at Interactive Brokers Margin Overview}\)
5.4.3 Data

A set of forty U.S. equities were randomly selected for these experiments (using the same criteria as in Chapter 4); the time-series spans from 2007/12/07 to 2018/03/07, the last 252 days (one year of trading days) of which were set aside for testing. The raw data used in this experiment is daily open, high, low, close (OHLC) prices and daily volume of shares traded. All datasets used to train the models contain a high-dimensional set of covariates that were created by extracting features from the OHLC market data. The list of features that make up the conditioning variables is too long to include in this chapter; however, interested readers can find a description of the features, alongside the operations required to create them, in Appendix B. To summarise, the conditioning variables extracted from the market price data fall into a number of sub-types, namely

- Moving average crossovers
- Bollinger band crossovers
- Candlestick functions
- Volatility, correlation and relative betas
- Technical indicators

The data used to train the models in both experiments consist of multiple datasets, categorised into two main groups. The first group contains the asset-specific datasets used for estimation of the marginal models, and the second group contains the datasets used for dependence model estimation. With the five-dimensional MS-GAMLSSD used here, a maximum of 30 individual datasets are needed; to reduce this number, we created one (282 feature) asset-specific dataset for each of the five univariate GAMLSS models, assigning the same conditioning variables to all parameters in $\Phi^{\eta_d}$; therefore $X_{\theta_i,d,k} = X_{\theta_j,d,k}, \forall i,j$.

The ten datasets $\{X_{z_1}, X_{z_2}, ..., X_{z_{10}}\}$ used for training the PCC dependence model are each composed of a union of the two marginal model datasets for which parameter $z_j$ determines the strength of dependence; hence the input space of the dependence model contains a total of 5640 features (2820 are unique).

In addition three types of base learner are specified. For the marginal models a decision stump and p-spline as functionals of each feature are specified, and for the dependence model decision stumps and p-splines are also specified for each feature; however, additionally a bivariate p-spine is also specified where the two features included in each base model are the corresponding features for each of the two assets whose relationship is being modelled, e.g. if $z_1$ is the parameter that determines the strength of relation between asset 3 and 5 then...
for all features in $X_3$ and $X_5$ a spatial effect base learner $\text{bispacial}(X_{3,i}, X_{5,i})$ is specified for each feature $i$; for more information on model pre-specification for model-based boosting see Section 3.2.2, and for a deeper discussion see Hofner, Mayr and Schmid (2014) and Hofner et al. (2014).

### 5.4.4 Backtesting Methodology

As mentioned at the beginning of this section, the two experiments are based on backtesting of the dynamic allocation strategy. As explained previously, the first is a test that involves trading of forty assets within one strategy; the second examines the average performance of a single randomly drawn five-stock portfolio. In both cases MS-GAMLSSD models are built using the training sets and tested, without re-estimation, over the one-year test period. Of the original pool of forty stocks there were found to be three that consistently caused the estimation procedure to fail and their inclusion in a portfolio meant that a model could not be built; therefore, these stocks had to be removed from the asset pool. This meant that the forty asset portfolio intended for Experiment 1 became 35 assets (seven five-asset portfolios) which were drawn randomly from the remaining asset pool of 37.

#### EXPERIMENT 1

In the first experiment 35 US equities are randomly partitioned into seven disjoint portfolios. The investor’s wealth is evenly distributed over these seven portfolios to create an equally weighted portfolio of sub-portfolios. Each sub-portfolio is modelled by a MS-GAMLSSD and independently performs its own dynamic allocation strategy, re-assigning its allocated wealth over its five assets on a daily basis. No funds enter or leave the individual sub-portfolio’s accounts during the backtesting period of 252 days.

#### EXPERIMENT 2

For the second experiment the average behaviour of a single randomly drawn portfolio is investigated. Here 100 portfolios, each containing five assets, are drawn randomly with replacement from the pool of 37 U.S. equities. Models are built for each portfolio and the trading strategy described in Section 5.4.2 is backtested over the test period.

The results of these experiments are compared to those of alternative strategies that use more traditional time-series forecasting methods; these include:

1. Full multivariate GAS model (MGAS MVSK) with a conditional Student’s t distribution, for which forecasts of the mean, variance, skewness and kurtosis (MVSK) of the returns
distribution can vary over time. The model is re-estimated at the beginning of each month using a 44 day moving window.

2. Partial multivariate GAS model (MGAS VSK), which has an unconditional mean, computed by sample estimates; the variance and higher moments (VSK) are conditional on past observations and can vary over time. The model is re-estimated at the beginning of each month using a 44 day moving window.

3. Traditional dynamic Markowitz optimal mean variance long-short portfolio strategy (Sample MV) using sample estimates of mean and variance generated by past observations using a weighted average estimator. The model is re-estimated daily using an effective 22 day window.

4. Naive equal weighted portfolio (EW), a passive strategy where wealth is assigned equally to all assets in a portfolio for the duration of the strategy.

The necessity of a regime-switching configuration for the model is also validated by comparing the MS-GAMLSSD with its single-state version, with results for both the switching (MS) and single state (SS) models being presented. Finally, results are presented alongside the Standard and Poor's 500 Index (S&P500), representing the performance of the market as a whole during the test period. To evaluate the performance of the test strategies, a selection of well known return / risk metrics for gauging portfolio performance are presented. These include the standard realised versions of Sharpe ratio, Sortino ratio, Calmar ratio and maximum drawdown; average daily portfolio turnover and transaction costs are also documented, in terms of percentage of allocation (max = 200%) and wealth, respectively.

TRANSACTION COSTS

As mentioned in the previous chapter, an important feature of the experiments was to ensure that backtesting as accurately as possible mimicked the conditions of a live trading environment. The most important point to consider in achieving this was an accurate re-creation of the costs incurred when trading financial assets. To this end, transaction costs were derived directly from Level 1 market data; a full derivation of the method used, and a detailed discussion of the reasoning behind the method, can be found in Section 6.3 of the next chapter.

The test results of both experiments are presented for all strategies with and without the inclusion of transaction costs; as well as the true (net) test results, which incurred costs every time positions are changed, zero cost results (gross) are also reported, which assume a theoretical 'frictionless' market where assets are bought and sold at the midpoint price. Both types of results are recorded in order to evaluate an important factor that determines the profitability of a strategy; this factor is portfolio turnover, which is a consequence of dynamic
strategies that is not controlled for by the model making the forecast, or by the optimisation that determines the portfolio weights. Presenting both gross and net results provides better insight into how the strategies are generating returns, and how expensive, or even how feasible, they are to operate.

5.5 RESULTS

EXPERIMENT 1: PORTFOLIO OF PORTFOLIOS

![Figure 5.7: Experiment 1: (35-asset portfolio). Shows the gross wealth curves of the strategies when transaction costs are not considered.](image)

Figure 5.7 shows the wealth curves of the strategies over the test period without adjusting for costs (gross). This figure shows that when a frictionless market is assumed, the two strategies using the GAMLSSD forecasts significantly outperform all other strategies, with the switching MS-GAMLSSD outperforming its single state version. However, Figure 5.8 shows that, when transaction costs are considered, and the wealth curves adjusted to reflect the costs of rebalancing the portfolios, the high turnover rates of the MS-GAMLSSD and SS-GAMLSSD, as well as the MGAS (MVSK) strategies, become apparent in the results, and performance for
TABLE 5.1: Experiment 1 (thirty five asset portfolios): Strategy performance metrics. Panel A reports the results for the strategies without transaction costs; Panel B the results when transaction costs have been deducted; and Panel C the results for the two benchmark portfolios. The letters MVSK stand for mean, variance, skewness and kurtosis, and denote which of these quantities are forecasted by the particular model.

EXPERIMENT 1

Panel A: (Gross)
No Transaction Costs

<table>
<thead>
<tr>
<th>Portfolio</th>
<th>Return (%)</th>
<th>Sharpe ratio</th>
<th>Sortino ratio</th>
<th>Calmar ratio</th>
<th>Max drawdown (%)</th>
<th>Portfolio turnover (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-GAMLSSD</td>
<td>53.36</td>
<td>6.2079</td>
<td>11.326</td>
<td>17.159</td>
<td>-2.23</td>
<td>71.47</td>
</tr>
<tr>
<td>SS-GAMLSSD</td>
<td>40.64</td>
<td>4.918</td>
<td>8.123</td>
<td>9.755</td>
<td>-3.15</td>
<td>73.16</td>
</tr>
<tr>
<td>MGAS MVSK</td>
<td>5.62</td>
<td>1.3201</td>
<td>1.9029</td>
<td>1.196</td>
<td>-4.55</td>
<td>120.21</td>
</tr>
<tr>
<td>MGAS VSK</td>
<td>17.71</td>
<td>1.8829</td>
<td>2.6108</td>
<td>2.507</td>
<td>-6.0</td>
<td>5.64</td>
</tr>
<tr>
<td>Sample MV</td>
<td>13.21</td>
<td>1.7882</td>
<td>1.9029</td>
<td>2.877</td>
<td>-3.85</td>
<td>13.32</td>
</tr>
</tbody>
</table>

Panel B: (Net)
Including Transaction Costs

<table>
<thead>
<tr>
<th>Portfolio</th>
<th>Return (%)</th>
<th>Sharpe ratio</th>
<th>Sortino ratio</th>
<th>Calmar ratio</th>
<th>Max drawdown (%)</th>
<th>Annual costs (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-GAMLSSD</td>
<td>21.19</td>
<td>2.7763</td>
<td>4.2593</td>
<td>5.1683</td>
<td>-3.42</td>
<td>22.64</td>
</tr>
<tr>
<td>SS-GAMLSSD</td>
<td>10.59</td>
<td>1.4529</td>
<td>2.0629</td>
<td>1.9753</td>
<td>-4.77</td>
<td>22.99</td>
</tr>
<tr>
<td>MGAS MVSK</td>
<td>-22.42</td>
<td>-6.0423</td>
<td>-6.3363</td>
<td>-0.9918</td>
<td>-25.56</td>
<td>30.95</td>
</tr>
<tr>
<td>MGAS VSK</td>
<td>16.05</td>
<td>1.7283</td>
<td>2.3829</td>
<td>2.2631</td>
<td>-6.1</td>
<td>1.42</td>
</tr>
<tr>
<td>Sample MV</td>
<td>9.21</td>
<td>1.2694</td>
<td>1.8173</td>
<td>1.8359</td>
<td>-4.29</td>
<td>3.66</td>
</tr>
</tbody>
</table>

Panel C: (Benchmarks)
Market Portfolio Proxies

<table>
<thead>
<tr>
<th>Portfolio</th>
<th>Return (%)</th>
<th>Sharpe ratio</th>
<th>Sortino ratio</th>
<th>Calmar ratio</th>
<th>Max drawdown (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EW</td>
<td>17.91</td>
<td>1.4221</td>
<td>1.9818</td>
<td>1.9891</td>
<td>-7.77</td>
</tr>
</tbody>
</table>

the MS-GAMLSSD and SS-GAMLSSD, in terms of portfolio return, becomes more comparable to the other strategies, and closer to the two benchmark portfolios, for which the effects of transaction costs are negligible.

3 All strategies involving CRRA utility optimisation used risk aversion of $\lambda = 5$. 
FIGURE 5.8: Experiment 1: (35-asset portfolio). Shows the net wealth curves of the strategies when transaction costs are considered.

However, despite the high portfolio turnover and large cost incurred from running the dynamic strategy using the MS-GAMLSSD model, this method still obtained the highest return over the test period, 20.19%, beating the equal weighted portfolio’s 16.03% and the S&P 500’s 14.88%. Moreover, from Table 5.1, when comparing the risk / return performance metrics, the MS-GAMLSSD’s performance is superior to all the other methods across all metrics. The MS-GAMLSSD strategy obtains a realised (arithmetic) Sharpe ratio of 2.78, beating the next-best score from MGAS (VSK) of 1.73 and a notably high Sharpe ratio from the naive equally weighted portfolio of 1.42. It further showed notable performance advantage in relation to the Calmar ratio and maximum drawdown results. These two, related, metrics indicate how well a strategy is able to prevent large losses. The MS-GAMLSSD model’s relative robustness to large market downturns is evident from an examination of the wealth curve of the S&P 500. There is a notable market downturn during February 2018 that impacts all portfolio strategies; however, the MS-GAMLSSD only loses 3.45% from its high during this period, compared to the 9.17% fall in the S&P 500, and an 8.18% loss in the equally weighted portfolio.³

Results from the single-state SS-GAMLSSD shows a weaker relative performance compared to its Markov switching counterpart. The comparison of the gross and net results from the strategy using forecasts from the single-state model, i.e. with and without transaction costs, is an example which illustrates the magnitude of the reduction in performance that dynamic strategies face in real world situations where costs are a critical factor. The high portfolio
5.5 RESULTS

In Experiment 1, an asset allocation strategy was proposed that split an investor’s wealth over 35 assets by equally weighting a portfolio of seven small portfolios each containing five assets. This allocation was proposed in order to more accurately resemble one that would be considered in the real world, where portfolio strategies would consider large numbers of assets in order to effectively reap the risk-mitigating benefits of diversification. The purpose of Experiment 2 was to examine the average performance of the smaller portfolios (where strategies allocate wealth over only five assets) individually; as would be expected, the results reflect the lower level of risk diversification than was attainable by the larger, more diversified turnover necessary for the strategy to achieve good gross performance effectively eroded away much of the profits, which resulted in a net annual return of 10.59%. Nevertheless, despite underperforming both the market and the equally weighted portfolio on the basis of annual net return, the SS-GAMLSSD outperforms the S&P500 on all risk / return metrics, and outperforms the equally weighted strategy on all metrics except the realised Calmar ratio.

EXPERIMENT 2: SINGLE PORTFOLIO STRATEGY

TABLE 5.2: Average strategy performance metrics from 100 randomly selected five-stock portfolios. Panel A reports the results for the strategies without transaction costs, Panel B shows the results when transaction costs have been deducted, and Panel C shows results for the two benchmark portfolios. The letters M, V, S, and K, where used, stand for mean, variance, skewness, and kurtosis, and denote which moments are considered in the portfolio optimisation by a particular strategy.

<table>
<thead>
<tr>
<th>Panel 2</th>
<th>No Transaction Costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Return (%)</td>
<td>Sharpe ratio</td>
</tr>
<tr>
<td>MS-GAMLSSD</td>
<td>50.8</td>
</tr>
<tr>
<td>( +/-4.13)</td>
<td>( +/-0.201)</td>
</tr>
<tr>
<td>SS-GAMLSSD</td>
<td>38.03</td>
</tr>
<tr>
<td>( +/-3.51)</td>
<td>( +/-0.189)</td>
</tr>
<tr>
<td>MGAS (MVSK)</td>
<td>2.63</td>
</tr>
<tr>
<td>( +/-1.78)</td>
<td>( +/-0.165)</td>
</tr>
<tr>
<td>MGAS (VSK)</td>
<td>19.84</td>
</tr>
<tr>
<td>( +/-2.23)</td>
<td>( +/-0.161)</td>
</tr>
<tr>
<td>Sample MV</td>
<td>15.57</td>
</tr>
<tr>
<td>( +/-2.2)</td>
<td>( +/-0.148)</td>
</tr>
</tbody>
</table>
### Table 5.3: Experiment 2: Average strategy performance metrics from 100 randomly selected five-stock portfolios. Panel A reports the results for the strategies without transaction costs, Panel B shows the results when transaction costs have been deducted, and Panel C shows results for the two benchmark portfolios. The letters M, V, S, and K, where used, stand for mean, variance, skewness, and kurtosis, and denote which moments are considered in the portfolio optimisation by a particular strategy.

#### EXPERIMENT 2

**Panel B: (Net)**

<table>
<thead>
<tr>
<th></th>
<th>Return (%)</th>
<th>Sharpe ratio</th>
<th>Sortino ratio</th>
<th>Calmar ratio</th>
<th>Max draw-down (%)</th>
<th>Annual costs (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-GAMLSSD</td>
<td>22.92</td>
<td>1.6974</td>
<td>2.7171</td>
<td>3.8041</td>
<td>-7.3107</td>
<td>19.7</td>
</tr>
<tr>
<td></td>
<td>(+/-2.81)</td>
<td>(+/-0.185)</td>
<td>(+/-0.321)</td>
<td>(+/-0.606)</td>
<td>(+/-0.727)</td>
<td>(+/-1.82)</td>
</tr>
<tr>
<td>SS-GAMLSSD</td>
<td>14.34</td>
<td>1.0996</td>
<td>1.6713</td>
<td>2.1655</td>
<td>-8.841</td>
<td>18.72</td>
</tr>
<tr>
<td></td>
<td>(+/-2.63)</td>
<td>(+/-0.176)</td>
<td>(+/-0.273)</td>
<td>(+/-0.433)</td>
<td>(+/-0.898)</td>
<td>(+/-1.78)</td>
</tr>
<tr>
<td>MGAS (MVK)</td>
<td>-25.28</td>
<td>-2.8368</td>
<td>-3.4536</td>
<td>-0.8614</td>
<td>-32.634</td>
<td>32.07</td>
</tr>
<tr>
<td></td>
<td>(+/-1.86)</td>
<td>(+/-0.243)</td>
<td>(+/-0.264)</td>
<td>(+/-0.042)</td>
<td>(+/-2.175)</td>
<td>(+/-1.5)</td>
</tr>
<tr>
<td>MGAS (VSK)</td>
<td>17.69</td>
<td>1.316</td>
<td>1.9774</td>
<td>2.1436</td>
<td>-9.7409</td>
<td>1.89</td>
</tr>
<tr>
<td></td>
<td>(+/-2.24)</td>
<td>(+/-0.164)</td>
<td>(+/-0.256)</td>
<td>(+/-0.343)</td>
<td>(+/-0.958)</td>
<td>(+/-0.44)</td>
</tr>
<tr>
<td>Sample MV</td>
<td>11.54</td>
<td>0.8763</td>
<td>1.3731</td>
<td>1.4919</td>
<td>-9.748</td>
<td>3.59</td>
</tr>
<tr>
<td></td>
<td>(+/-2.26)</td>
<td>(+/-0.157)</td>
<td>(+/-0.25)</td>
<td>(+/-0.329)</td>
<td>(+/-0.787)</td>
<td>(+/-0.21)</td>
</tr>
</tbody>
</table>

#### Panel C: (Benchmarks)

<table>
<thead>
<tr>
<th></th>
<th>Return (%)</th>
<th>Sharpe ratio</th>
<th>Sortino ratio</th>
<th>Calmar ratio</th>
<th>Max draw-down (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EW</td>
<td>15.91</td>
<td>1.0252</td>
<td>1.4768</td>
<td>1.5634</td>
<td>-11.3296</td>
</tr>
<tr>
<td></td>
<td>(+/-2.27)</td>
<td>(+/-0.135)</td>
<td>(+/-0.199)</td>
<td>(+/-0.260)</td>
<td>(+/-0.810)</td>
</tr>
</tbody>
</table>

Portfolios of Experiment 1, with lower values for the risk / return statistics being observed in Experiment 2 across all of the strategies tested.

Notwithstanding the lower capacity for diversification, the results of Experiment 2 mirror the relative ordering across the tested strategies displayed in the 35-asset previous experiment.
5.5 RESULTS

FIGURE 5.9: Fan chart of the wealth curves for 100 randomly selected five-asset portfolios without transaction costs.

Again, the strategy using probabilistic forecasts from a MS-GAMLSSD model outperforms all other methods, as well as the market (S&P 500 index), on all metrics. The annual net return of 22.95% (20.14 - 25.73, 95% confidence interval) obtained with the MS-GAMLSSD exceeds that of the second-best performing method MGAS (VSK), which achieved 17.69% (15.45 - 19.93, 95% c.i.). Without adjusting for transaction costs, MS-GAMLSSD strategies obtained an average return of 50.8% (46.62 - 54.98, 95% c.i.) and a Sharpe ratio of 3.294 (3.093 - 3.495, 95% c.i.) which after adjusting for transaction costs is reduced to 1.697 (1.512 - 1.882, 95% c.i.). By contrast, the MGAS (VSK) generates only a 19.84% (17.61 - 22.07, 95% c.i.) gross annual return, and a Sharpe ratio of 1.456 (1.295 - 1.617, 95% c.i.), reducing to 1.316 (1.152 - 1.48, 95% c.i.) when costs are considered.
With a more detailed look into the results of the two best performing strategies (MS-GAMLSSD and MGAS (VSK)), with particular attention paid to the differences between gross and net performance, we can see that there is a much larger difference between the gross and net terminal wealth of the MS-GAMLSSD strategies than there is for the MGAS (VSK); a much greater proportion of the wealth generated by the MS-GAMLSSD strategy is needed to transact rebalancing of the portfolio than we see with the MGAS (VSK). From this we can infer, on the positive side, that forecasts from the MS-GAMLSSD are affecting a much greater change in allocation of assets over the test period and that, therefore, the method is much more agile at responding to changes in information. This agility is measured in these experiments by average turnover, which Table 5.2 shows was only 7.48% for the MGAS (VSK) but a far higher 68.18% for the MS-GAMLSSD strategy. This translates to a cost of 19.7% of the portfolio value needed
FIGURE 5.11: Experiment 2: Average wealth curves for gross (top) and net (bottom).
to operate the higher turnover MS-GAMLSSD strategy, compared to only a 1.87% cost for the MGAS (VSK). The other MGAS strategy tested (MVS) provides an example where high turnover can make a strategy completely unfeasible; in this case, the strategy forecasts are of low quality paired with frequent portfolio rebalancing, resulting in large losses that can be attributed to transaction costs.

Although, when transaction costs were not taken into account, the single state SS-GAMLSSD far out-performed all market and model benchmarks, when transaction costs were accounted for, the single state SS-GAMLSSD, as in Experiment 1, did not perform as well as the MGAS (VSK) and the equal weighted portfolio. With similar turnover and annual costs as its Markov-switching counterpart the SS-GAMLSSD was unable to generate large enough returns to effectively offset the costs and beat the two, better performing, low-turnover portfolio strategies. Using a one-sided two-sample Welsh test, testing against the null hypothesis that the performance metric statistics of the MS-GAMLSSD and SS-GAMLSSD models are not greater than those of the naive equally weighted benchmark, for the net SS-GAMLSSD results, the null could not be rejected with 95% confidence for the Sharpe and Sortino ratios; however, the null was rejected for the Calmar ratio and the maximum drawdown. For the MS-GAMLSSD model net results the null was rejected at 95% confidence for all performance metrics.

Returning to the main MS-GAMLSSD model, Figure 5.9 shows the distribution over time of the realised wealth trajectories of all 100 portfolio strategies. (In both this and Figure 5.10, the average of the 100 equally weighted portfolios is used as the proxy for the market portfolio, indicated by the red line.) Out of the 100 MS-GAMLSSD five-asset portfolios, only five generated a gross annual return of less than 23.14%. All 100 finished above the benchmark market portfolio, while in comparison, for the 100 individual equally weighted portfolios, there were only 46 portfolios that had an annual return which finished above the market.

In the fan chart of Figure 5.10, where now costs have been deducted, 68% of the portfolios following the MS-GAMLSSD strategy outperformed the market portfolio, though it is also the case that all but the worst-performing 5% of these portfolios ended up with positive returns for the year, in contrast with the naive, equally-weighted, strategy in which 10% of portfolios finished with negative returns. Looking at these results from a more positive angle, the MS-GAMLSSD strategy had 26 portfolios finish with an annual return above 30% and 14 with an annual return in excess of 40%, in contrast to the naive strategy which had 11 finish above

---

4 Here we do not report the results for the directional forecast (DF) strategy, despite its good results in Chapter 4. Using the MS-GAMLSSD forecasting model, DF results came in as expected, being outperformed by the primary strategies and unable to repeat the performance seen in the previous chapter. Therefore, these results were not deemed to add value and so not included here.
30% and only one of the 100 naive equally weighted portfolios generating an annual return in excess of 40\%.

5.6 CONCLUSION

In this chapter, a Markov-switching generalised additive model for location, scale, shape and dependence was introduced for the purpose of probabilistic forecasting of multivariate financial time series. The MS-GAMLSSD, estimated via a gradient boosting method, produces time-varying conditional forecasts of all features of a probability distribution; in marked contrast to the vast majority of time-series forecasting models, forecasts are conditioned on high-dimensional sets of exogenous variables. The use of a copula decomposition of the multivariate density allowed for flexible modelling of the dependent structure using pair-copula constructions and provided a solution to overcome issues involved in multivariate distributional regression model selection and estimation. The specification of the model was tailored to handle the challenges of capturing the many known statistical regularities in multidimensional financial price returns data including, but not limited to, time-variation in the price returns’ distribution, the non-linearities and asymmetries in shape and dependencies, as well as the intertemporal changes in market regimes. The domain for the model’s application was also specifically considered in its design, which made it suited to act as a signal generating component of a wider systematic trading strategy for dynamic asset allocation.

Two experiments were undertaken to test the MS-GAMLSSD’s performance in this role; in both cases, it was found the strategy using forecasts generated by the MS-GAMLSSD model outperformed all other strategies and market benchmarks on annual return and on the entire range of portfolio risk performance metrics. Comparing the Markov-switching MS-GAMLSSD with its single state version, it was shown that there are clear benefits in performance from adopting the regime switching specification over using the single state model. The strategy of trading a portfolio of multiple sub-portfolios from Experiment 1 showed that significant additional risk mitigation could be achieved by the naive diversification of multiple portfolios, each modelled by the MS-GAMLSSD. Furthermore, it was shown that the portfolio return from this strategy came close to the expected return discovered in Experiment 2, which examined the average behaviour of individual portfolios, but with significantly less risk to an investor. It can be concluded that a strategy of the type considered in Experiment 1 would be viable for application in a live trading situation with exposure to market transaction costs.
In both experiments the inclusion of transaction costs was a key factor in order to infer the generalisability of the backtests and therefore the viability of strategies within a live trading environment. For this reason results were reported both with and without considering the cost of rebalancing, with costs derived directly from real-world market prices. This provided interesting insight into how the various strategies, MS- and SS-GAMLSSD and the considered alternatives, generated wealth over the test period. It was found that portfolio turnover varied substantially across the different strategies tested, with the MS- and SS-GAMLSSD strategies requiring frequent rebalancing to track the optimal portfolio over time, and therefore generating high costs that substantially reduced growth in their profit trajectories, terminal wealths, and, hence, all realised performance metrics being also subject to reductions due to high strategy operating costs.

Examining strategy performances with transaction costs omitted allowed for the direct assessment and comparison of the models’ predictive performances, whereas the net results, adjusted for costs, were not by themselves sufficient to draw conclusions on forecast quality. This is due to the dynamic allocation strategy not considering the impact of costs when making rebalancing decisions; strategies are therefore not optimal when transaction costs are included, and hence do not sufficiently reflect the quality of forecasts generated by the models. This means that the predictive performances of the models that the MS-GAMLSSD and SS-GAMLSSD were tested against were most likely of poorer relative quality than the net results suggest, while the opposite is true for the quality of forecasts from the MS-GAMLSSD and SS-GAMLSSD models.

A strategy that generated low portfolio turnover is indicative of the predictive model doing very little work revising forecasts over time, whereas high turnover indicates the opposite. High turnover paired with high gross performance is suggestive of a good quality predictive model, whereas a low turnover and high net performance suggests that the strategy may have benefited from the underlying market conditions and not necessarily from the quality of forecasts; the S&P500 during the test period clearly shows a very strong bullish trend which would have the effect of assisting portfolios that predominately took and maintained long positions, ensuing a market condition bias in their net results. This was evident for the Sample (MV) and MGAS (VSK) strategies, and, of course, reflected in the very good results for the naive equally weighted strategy.

For an ideal dynamic portfolio strategy, there is clearly a trade-off that could be made between the positive effects of using forecasts from a good predictive model displaying high responsiveness to new information, such as was seen with the MS-GAMLSSD, and the negative effect of the costliness of the high portfolio turnover that is generated by the frequency and magnitude of forecast revisions. In other words, it is expensive to have a good quality predictive model in
a systematic strategy of the type used here, but there may be some manner in which strategy profitability can be improved. This could be achieved with a portfolio strategy that directly optimises net performance rather than simply assuming that good net performance will naturally follow from gross performance optimisation. Such a strategy is the emphasis of the following chapter.
Part III

OPTIMAL EXECUTION
Transaction Cost Minimisation for Dynamic Asset Allocation Strategies

Despite overall good performance compared to other models and benchmarks, the MS-GAMLSSD model and associated portfolio strategies were shown to incur significant cost due to the high amounts of portfolio turnover they generated in tracking the optimal allocation. This drawback is not one uniquely associated with the models tested in the previous two chapters; it is common to any models whose forecast sensitivity results in high portfolio turnover. Forecasting models that are sensitive to changes in information will inevitably force portfolios to rebalance frequently, and traditional portfolio optimisation methods are unaware of the costs of their proposed rebalancing. To this end, in this chapter a new portfolio optimisation method for dynamic asset allocation is introduced. In this new method the proposed transaction costs for a rebalancing are incorporated into the utility function through a discounting of the investor’s wealth. The new cost minimisation method is tested against that of Chapter 5; the results show the new method significantly outperforms the original on a broad range of portfolio metrics.
6.1 INTRODUCTION

Most well-known theories which remain at the centre of the fields of financial economics and mathematical finance come attached with important underlying assumptions, one of these assumptions that we have seen time and time again being that the financial markets are 'frictionless'. The theoretical concept of frictionless markets implies that investors operate in an environment in which there are no restrictions to trading financial securities, and that this, most unrealistically, includes no costs. Assuming that trading is free to investors has provided some elegant mathematical solutions to classical problems, with option pricing and hedging with replicating portfolios as well-known examples. However, when the assumption is relaxed, the inclusion of frictions substantially changes the nature of these problems.

Over the years, researchers have revisited some of the classical models and theories in the financial literature while reconsidering their value in the presence of transaction costs; these include the Cox-Ross-Rubinstein binomial option pricing model (Cox, Ross, and Rubinstein (1979)), for which Boyle and Vorst (1992) showed how to rebalance a replicating portfolio in discrete time for the case where transaction costs are proportional. Merton, whose original work was notoriously reliant on frictionless market assumptions, provided in a later contribution a method for a replicating portfolio in discrete time with an approach that incorporated an allowance for transaction costs incurred by portfolio rebalancing (Merton and Samuelson (1992)), H. E. Leland (1985) investigated within a continuous-time framework and derived a Black-Scholes type approximation for the option price in the presence of proportional transaction costs. Notable early work on asset allocation problems with transaction costs includes that of Davis and Norman (1990); their contribution provides a solution for Merton’s portfolio problem with proportional transaction costs. Merton’s problem (Merton (1975)) is a typical example of a continuous time financial model whose function is reliant on the assumption of frictionless markets. Davis and Norman (1990) comment that “Any attempt to apply Merton’s strategy in the face of transaction costs would result in immediate penury, since incessant trading is necessary to hold the portfolio on the ‘optimal’ Merton line.” More recently, revisiting the problem of portfolio selection, Gârleanu and Pedersen (2013) make the important point that globally optimal solutions to problems that neglect to include market frictions, such as the Markowitz tangency portfolio (Markowitz, 1952), are no longer optimal once transaction costs are involved. With respect to dynamic asset allocation in particular, they note that “The Markowitz portfolio is a moving target, since the return-predicting factors change over time. Due to transaction costs, it is obviously not optimal to trade all the way to the target all the time.”
The above statements highlight the core issue associated with optimal portfolio selection under transaction costs: in a real-world situation it is usually not viable to track an optimal (target) strategy too closely over time because the costs of doing so can easily erode any profits. Instead, a rational investor must weigh up if the benefits of the proposed rebalancing are worth the cost. This creates an additional sub-problem for portfolio managers; as stated in H. Leland (2000), “frequent rebalancing to keep assets close to their target proportions will incur high trading costs.” But infrequent revision will create a 'tracking error' relative to the ideal portfolio strategy. Early work that considered this issue, albeit with respect to a single risky asset and a riskless bank account, include Davis and Norman (1990), who followed the work of Constantinides (1979), to show there exists a region within the space of portfolio weights, between the current allocation and optimal choice, for which it is not worthwhile to adjust position in the allocation. Therefore, there are times when holding the current suboptimal allocation is actually preferred to making the costly adjustment to the theoretical optimum. Under these circumstances, the policy of asset allocation can be determined by a region for which trading is not recommended, which became known as a no-trade region. The concept of no-trade regions was further examined in the case of a single risky asset and money market account in Dumas and Luciano (1991), Shreve and Soner (1994) and Liu and Loewenstein (2002). For multiple risky assets, the concept of trading policies based on a no-trade region was extended by Akian, Menaldi, and Sulem (1996), Atkinson, Pliska, and Wilmott (1997) H. Leland (2000), and Lynch Lynch and Tan (2010). However, as commented by DeMiguel, Martin-Utrera, and Nogales (2015), much of the early work on transaction costs mentioned above is from the point of view of optimal investment and consumption problems in continuous time, or optimisation of a multi-period expected utility. There were few experimental studies that specifically tackled the issue of practical portfolio re-balancing as part of a greater dynamic portfolio allocation strategy.

Gârleanu and Pedersen (2013) suggested a method to find an 'optimal' portfolio allocation that lies somewhere between the current allocation and a globally optimal target portfolio; they show that by assuming transaction costs are quadratic in the weight adjustment vector, and proportional to the covariance matrix, the new optimal portfolio can be expressed as a linear combination of the current and globally optimal allocations. They conclude that, rather than fully rebalancing the portfolio to a globally optimal choice, when transaction costs are included it is more profitable to also minimise turnover, and thus take a smoother and more conservative adjustment in the direction of the globally optimal portfolio, but not all the way to it.

Portfolio turnover, defined as the proportion of the portfolio bought or sold as part of a re-balancing, is the single most contributory factor that determines the cost of the re-balancing when transaction costs are proportional. With proportional costs that are known, ex-ante of trading, turnover can be reduced as part of an optimisation objective by including a penalty
term to the objective function that penalises the rebalancing. By investigating this idea, a very interesting result has been discovered in Olivares-Nadal and DeMiguel (2018) and Hautsch and Voigt (2019). Both studies show that including the minimisation term in the objective function quite remarkably can improve a portfolio performance considerably, with two reasons given for this. The first, as expected, is the reduction in turnover, and therefore cost. The second, less obvious, is the additional term acting as a regularisation term in the optimisation function, so addressing the problem of estimation error in forecasts of mean and covariance. This technique is therefore able to assist in the provision of better quality inputs to the optimisation function.

In their study, Olivares-Nadal and DeMiguel (2018) show there is an equivalence between three different robust portfolio optimisation methods and methods that consider transaction costs. They provide results that show that including transaction costs in the optimisation problem has an effect equivalent to shrinkage from a LASSO regularisation, when proportional transaction costs are specified, and an L2 (ridge) regularisation for quadratic costs. They further show that as estimation error is addressed as a consequence of reducing transaction costs, scaling the transaction cost components by a regularisation hyperparameter, calibrated by cross-validation, can further improve performance.

In a similar study, Hautsch and Voigt (2019) showed that managing transaction costs ex ante, that is, known at the time of a rebalancing decision, as part of the asset allocation problem, adjusts the portfolio weight vector to become a regularised version of the efficient allocation. They illustrate the effectiveness of penalising rebalancing of portfolios when this rebalancing is costly and show this method can have the effect of substantially increasing portfolio performance over the case where transaction costs are ignored. They also provide a theoretic equivalency between the inclusion of proportional transaction cost minimisation and shrinkage methods of the covariance matrix, thereby showing that accounting for costs can coincidentally improve performance by mitigating the impact of parameter uncertainty. They conducted experiments on large portfolios, using a selection of different mean variance estimation methods. They found that strategies that did not account for transaction costs lost significant amounts of money. Explicitly adjusting for transaction costs, however, strongly changed the outcomes: the adjusted strategies produced significantly positive average portfolio returns and reasonable Sharpe ratios. The portfolio turnover was also seen to be strongly attenuated and was lowered to less than 1% of the original portfolio turnover incurred by the strategies which made no attempt to reduce transaction costs.

The study carried out in this chapter differs from both those of Olivares-Nadal and DeMiguel (2018) and Hautsch and Voigt (2019) by introducing cost minimisation directly into the portfolio choice problem through a redefinition of an investor’s initial wealth while also considering higher
moments of the portfolio returns distribution; more precisely, the new method is an extension of
the optimisation problem used by Jondeau and Rockinger (2006a) and Jondeau and Rockinger
(2012), which is based on a fourth order Taylor expansion of the expected utility function. For
an investor that demonstrates constant relative risk aversion (CRRA), the new method redefines
the investor’s initial wealth as a function of the known costs of a proposed rebalancing. With
this adaptation to the CRRA utility function, we derive a new objective function from expected
utility theory that gives an optimal portfolio solution when facing variable costs for rebalancing
a portfolio. Unlike the two aforementioned studies, experiments in this chapter use transaction
costs that are a function of real world bid and ask prices and provide proportional transaction
costs that are based on the bid / ask spread. Therefore, unlike previous studies that have
treated costs as a flat, fixed proportion of the size of a re-balancing, the dynamic portfolio
strategies in this chapter accommodate costs that vary across different assets and time. If
costs are assumed the same for all securities, and do not vary over time, then the problem
is simply that of turnover reduction, with similarities to LASSO regularisation with the costs
acting as a scaling hyperparameter. However, incorporating variable costs into the optimisation
means that optimisation of the new objective function is, in addition to being an expected
utility maximisation, also a de facto cost minimisation problem, where solutions will tend to
discriminate against more costly transactions in favour of ones that are less expensive to
implement. The new optimisation method is tested against the original method of Jondeau
and Rockinger (2012), as was used in the dynamic portfolio strategies from Chapter 5. In the
results of both the previous research chapters, it was found that high-turnover strategies saw a
significant suppression in performance due to the frequent unfettered reallocation of wealth,
which consequently generated high costs and clearly reduced profitability. The experiments
in this chapter duplicate those of the previous chapter with the only difference being the new
objective function for the asset allocation problem. The results of Chapters 5 and 6 (this chapter)
are then compared and contrasted. The remainder of the chapter is arranged as follows: In
Section 6.2, background material for expected utility theory is overviewed in brief, along with the
higher-moments portfolio optimisation method of Jondeau and Rockinger (2006a) and Jondeau
and Rockinger (2012). Section 6.3 describes the issues connected to transaction costs. In
Section 6.4, the new utility function and extension of the Taylor series expansion method is
introduced, which includes proportional transaction costs. In Section 6.5, the experiments
involving dynamic portfolio strategies, performed using forecasts generated by the models
introduced in Chapter 5, are described, followed by the results of the experiments in Section 6.6.
The chapter concludes with a discussion in Section 6.7.
6.2 BACKGROUND

6.2.1 Higher Moment Portfolio Optimisation

This subsection revisits the optimisation method of Chapter 5, in order that this method can now be extended. For the optimal allocation strategies studied in this chapter, we consider a rational investor having full information about the distribution of their future wealth. Using this information, which is periodically updated, the investor engages in a dynamic asset allocation strategy by maximising their expected utility at points over the investment period. That is, the dynamic strategy is made up of a series of static portfolio choice problems at discrete points over the investment period $T$, that is $\{t_0, t_1, t_2, ..., t_T\}$. At every point $t$, new information is revealed regarding the expected joint behaviour of the assets over the next period, and a decision is made to rebalance the portfolio to a new allocation that maximises an investor’s expected end-of-period utility.

Expected utility maximisation is a commonly used framework in portfolio selection problems. The framework is based around a utility function chosen to reflect the preferences and risk tolerance of an investor. The utility function $U$ is a mapping of the terminal wealth $W_{t+1}$, at some incremented time point in the future, to a real valued measure of the investor’s utility, given their preferences. The wealth at time $t + 1$ consists of two components, the current wealth $W_t$ and the one-period return of the portfolio $R_{t+1}$. With discrete compounding this is simply defined as $W_{t+1} = W_t (1 + R_{t+1})$, where the return at time $t + 1$ is $R_{t+1} = \sum_i w_{i,t}r_{i,t+1}$, with $w_{i,t+1}$ representing an element of the weight vector $w_{i,t+1}$. For a portfolio of $N$ risky assets, the allocation of wealth distribution across the $N$ risky assets over the time period $t+1$ is represented by the portfolio weight vector $w_{t+1} = (w_{1,t+1}, w_{2,t+1}, ..., w_{N,t+1})$.

When applying expected utility theory to portfolio choice problems, the goal is to find a $\hat{w}_{t+1}$ that maximises future expected utility; the objective function of the optimisation problem can then be written as

$$\hat{w}_{t+1} := \arg\max_w \mathbb{E}[U(W_{t+1})].$$
The above expectation cannot be represented in closed form; however, for an infinitely differentiable utility function $U$, the function $U : W \rightarrow \mathbb{R}$ can be approximated by the following Taylor series expansion, that is taken around the point of initial wealth $W_t$,

$$U(W_{t+1}) \approx \sum_{k=0}^{\infty} \frac{U^{(k)}(W_t)}{k!} (W_{t+1} - W_t)^k, \quad (6.1)$$

where the superscript $(k)$ refers to the $k$th order derivative.

Taking the expectation of both sides above, we get

$$E[U(W_{t+1})] \approx \sum_{k=0}^{\infty} \frac{U^{(k)}(W_t)}{k!} E[(W_{t+1} - W_t)^k]. \quad (6.2)$$

Therefore, the terms on the r.h.s. of Equation (6.2) are expressed in terms of the central moments of the distribution of future wealth $m_k = E[(W_{t+1} - W_t)^k]$ over the period $[t, t+1]$.

Truncating Equation (6.1) to the fourth order is assumed to still provide an accurate approximation of $U$; however, there is some debate as to how good an approximation truncation provides. Arguments to justify the fourth order expansion in the context of expected utility can be found in Dittmar (2002) and Kimball (1993), and were also discussed in Section 2.3 of Chapter 2.

As the problem is with regards to a future value $W_{t+1}$, the expectation of both sides of the above becomes

$$E[U(W_{t+1})] \approx U^{(0)}(W_t) + U^{(1)}(W_t) E[(W_{t+1} - W_t)] + \frac{U^{(2)}(W_t)}{2} E[(W_{t+1} - W_t)^2] + \frac{U^{(3)}(W_t)}{6} E[(W_{t+1} - W_t)^3] + \frac{U^{(4)}(W_t)}{24} E[(W_{t+1} - W_t)^4]. \quad (6.3)$$

As discussed in Chapter 5, defining $U$ as a CRRA utility function, we can set $W_t = 1$ which means $E[(W_{t+1} - W_t)^k] = E[R_{t+1}^k]$, where $R_{t+1}$ is the portfolio return, and so the above can be expressed in terms of the first four non-central moments of the portfolio returns distribution

$$E[U(W_{t+1})] \approx U^{(0)}(W_t) + U^{(1)}(W_t) m_1 + \frac{U^{(2)}(W_t)}{2} m_2 + \frac{U^{(3)}(W_t)}{6} m_3 + \frac{U^{(4)}(W_t)}{24} m_4. \quad (6.3)$$

Therefore, when applying the expected utility framework to an asset allocation problem with higher moments, the objective function for the optimisation problem simply consists of two
6.2 BACKGROUND

With the above formulation for the expected utility, as seen in Equation (6.3), the problem extends the theory of mean variance optimisation into four dimensions. For example, Markowitz’s tangency portfolio is an optimal portfolio that considers the trade-off between return and risk, as measured by the expected mean and variance of a portfolio. The optimal solution to the above objective function, likewise, can also be considered as a tangency point, but in four-dimensional space, as discussed in Martellini and Ziemann (2010). In the standard mean-variance case, the investor’s preferences, as expressed by the choice of utility function $U$, are to favour a higher mean (return) and a lower variance (risk). When considering utility with the higher moments taken into account, a rational risk-averse investor’s preference will be to favour odd moments and disfavour the even moments of the portfolio returns distribution.

6.2.2 Portfolio Moments

To solve the optimisation problem introduced above, the inputs required by the objective function are the forecasts of the one-step-ahead portfolio non-central moments. To compute these values, the forecasts of the first four components of the joint distribution of asset returns are required. The response vector $\hat{y}_{t+1} = (\hat{y}_{1,t+1}, \hat{y}_{2,t+1}, \ldots, \hat{y}_{d,t+1})$ is the vector of forecasted asset returns. The random vector $y_{t+1} \in Y_{t+1}$ has the joint distribution

$$Y_{t+1} \sim D \left( \hat{M}_{t+1}, \hat{\Sigma}_{t+1}, \hat{S}_{t+1}, \hat{K}_{t+1} \right)$$

where, for a $d-$dimensional distribution, $\hat{M}_{t+1} \in \mathbb{R}^d$, $\hat{\Sigma}_{t+1} \in \mathbb{R}^{d \times d}$, $\hat{S}_{t+1} \in \mathbb{R}^{d \times d}$ and $\hat{K}_{t+1} \in \mathbb{R}^{d \times d \times d}$ are the forecasted mean, covariance, coskewness and cokurtosis of the price returns of the assets in the portfolio over the time period $[t, t + 1]$.

The coskewness and cokurtosis tensors can be expressed as two-dimensional standard arrays, as suggested by De Athayde and Flôres Jr (2004) and adopted in Martellini and Ziemann (2010), Boudt, Lu, and Peeters (2015), Jondeau and Rockinger (2012) and Bernardi and Catania (2019), amongst others. The tensors of coskewness and cokurtosis, of dimensions $(N \times N \times N)$ and $(N \times N \times N \times N)$ respectively, are transformed into the $(N \times N^2)$ and $(N \times N^3)$ matrices by slicing each $(N \times N)$ layer and pasting them, in the same order, sideways.

For a given weight vector $w_{t+1} = (w_{1,t+1}, w_{2,t+1}, \ldots, w_{D,t+1})$, the central moments of the portfolio returns distribution can be written as a function of the forecasted comoments of the joint
distribution of asset returns over the period \([t, t+1]\) and the assignment \(w_{t+1}\) of wealth into each of the assets. These central moments can be computed with the following method,

\[
\begin{align*}
\hat{\mu}_{P,t+1} &= w_t'^t \hat{\mu}_{t+1} \\
\hat{\sigma}^2_{P,t+1} &= w_t'^t \hat{\sigma}_{t+1} w_{t+1} \\
\hat{s}_{P,t+1} &= w_t'^t \hat{s}_{t+1} (w_{t+1} \otimes w_{t+1}) \\
\hat{k}_{P,t+1} &= w_t'^t \hat{k}_{t+1} (w_{t+1} \otimes w_{t+1} \otimes w_{t+1}),
\end{align*}
\]

\hspace{1cm} (6.4)

where \(\otimes\) is the Kronecker product. The final step in obtaining the non-central moments of the portfolio distribution from their corresponding central moments computed in Equation (6.4) requires the following computations,

\[
\begin{align*}
m_{1,t+1} &= \hat{\mu}_{P,t+1} \\
m_{2,t+1} &= \hat{\sigma}^2_{P,t+1} + \hat{\mu}^2_{P,t+1} \\
m_{3,t+1} &= \hat{s}^3_{P,t+1} + 3 \hat{\sigma}_{P,t+1} \hat{\mu}_{P,t+1} + \hat{\mu}^3_{P,t+1} \\
m_{4,t+1} &= \hat{k}^4_{P,t+1} + 4 \hat{s}^3_{P,t+1} \hat{\mu}_{P,t+1} + 6 \hat{\sigma}^2_{P,t+1} \hat{\mu}^2_{P,t+1} + \hat{\mu}^4_{P,t+1}.
\end{align*}
\]

\hspace{1cm} (6.5)

### 6.2.3 The Utility Function (CRRA)

The basic asset allocation strategy in this chapter, which has previously been used in Martellini and Ziemann (2010), Jondeau and Rockinger (2012) and Bernardi and Catania (2019), involves a rebalancing of the portfolio weights at points indexed by \(t\). In order to allow the use of the assumption made in the previous subsection, that at every \(t\) the initial wealth \(W_t = 1\) will not influence the solution, the function \(U(\cdot)\) should display a constant relative risk aversion (CRRA) property. Simply put, an investor’s risk aversion under this condition is not affected by their level of current wealth. Letting \(\lambda\) be a scalar constant that represents the level of an investor’s risk aversion, the following function,

\[
U(W_{t+1}; \lambda) = \begin{cases} 
\frac{W_{t+1}^{1-\lambda} - 1}{1-\lambda} & \text{for } \lambda > 1 \\
\log (W_{t+1}) & \text{for } \lambda = 1
\end{cases}
\]

is a utility function that displays constant relative risk aversion.
Approximating the utility function by a fourth order Taylor expansion and taking expectations of both sides yields

\[
\mathbb{E}[U(W_{t+1})] \approx U^{(0)}(W_t) + U^{(1)}(W_t)m_{1,t+1} + \frac{U^{(2)}(W_t)}{2}m_{2,t+1}^2 + \frac{U^{(3)}(W_t)}{6}m_{3,t+1}^6 + \frac{U^{(4)}(W_t)}{24}m_{4,t+1}^4
\]

\[= \frac{1}{1-\lambda} + m_{1,t+1} - \frac{\lambda}{2}m_{2,t+1}^2 + \frac{\lambda(\lambda + 1)}{6}m_{3,t+1} - \frac{\lambda(\lambda + 1)(\lambda + 2)}{24}m_{4,t+1}^4 \tag{6.6}\]

as the objective function of the portfolio optimisation problem.

6.3 TRANSACTION COSTS: THE BID-ASK SPREAD

When considering costs of trading securities that happen to be known up front and, additionally, happen to be proportional to the cash value of a position, the costs incurred through the bid-ask spread are by far the most detrimental to the profitability of a strategy. This is especially the case for strategies that require frequent trading and therefore have a high rebalancing turnover. Bid-ask spreads are provided at the time of trading and their effect can be represented as a proportion of market price, in basis points (bps). The spread across different securities can vary widely over time; for example, Chan, Chung, and Johnson (1995) show that spreads for NYSE securities tend to be highest at the beginning and end of the day, whereas Chan, Christie, and Schultz (1995) show that, by contrast, spreads for NASDAQ traded securities are relatively stable throughout the day but narrow significantly near the close.

Bid-ask spreads are derived from the difference in the prices that a seller is willing to sell at, and a buyer is willing to buy at. These two prices are quoted by market-makers, to whom devolves the responsibility of the efficient functioning of a financial market. The highest price at which a market-maker will buy the stock is called the bid price, whereas the lowest price among those willing to sell is known as the ask price. The true price of a financial security is not known and lies somewhere in between the quoted bid and ask prices. This means that when we trade we are always paying slightly more than the true price when buying and receiving slightly less when selling. These differences are a type of transaction cost and can be extremely large when accumulated over the course of a trading strategy, especially one that involves the frequent buying and selling of assets.
In the last few years, online trading platforms for retail investors have emerged, which operate on a commission-free basis. This has led to traditional brokerages, in order to compete, being forced to dramatically reduce commissions for placing trades. For online retail investment brokerage eToro, revenues are generated by providing their own proprietary bid-ask prices just outside the spread offered by the market. Under this cost structure, which is also the model for many other platforms, the bid-ask spread is the sole cost for an investor to trade. This revolution in retail investing has also led to many platforms offering fractional share dealing. This system allows an investor to place any denomination of cash into any security, whereas before, if one were to want to invest in a company such as Amazon (AMZN), the minimum amount would be one full share, which at the time of writing would mean an investment of $3590. Today, due to these advances in retail investment, it is possible for a portfolio allocation to be implemented exactly according to a optimisation solution, for any level of wealth. Therefore methods in this chapter are directly applicable to real world practice.

The asset-specific factors generally understood to influence the size of the bid-ask spread are volatility, trading volume and market value. In Atkins and Dyl (1997) the authors show that higher volatility is related to wider bid-ask spreads; Menyah and Paudyal (2000) show that spreads vary with the liquidity of a stock and, in their analysis of futures markets, Wang and Yau (2000) show a positive relationship between volatility and the bid-ask spread and, after controlling for other factors, an inverse relationship between the bid-ask spread and trading volume.

Because of the way in which stock markets efficiently operate, when an investor enters or exits a position they will incur an immediate loss proportional to the size of the position. This amount is proportional to the difference between the bid price and the true market price, in the case of a buy, or the closing of a short position; or oppositely, the difference between the bid price and the true price when entering into a short position or closing a long position. The assumption that the true market price lies exactly halfway between the bid and ask prices is a reasonable one, and also very convenient, as the costs incurred through the spread can then be separated into opening and closing liabilities. That is, when opening a long position in an asset, an investor will pay the ask price, an amount above the market price equal to half of the difference between the bid and ask prices, as quoted at the time the trade is made, and when closing the position an investor will sell for less than the market price, computed in the same way. Note this price difference is unlikely to be the same as when opening the position, as the bid-ask spreads are stochastic quantities that vary over time. Nevertheless the contribution the bid-ask spread has over the lifetime cost of buying, holding and selling an asset can be separated into two parts. These are at the points where buying and selling takes place, and the costs are furthermore

---

3 This was written prior to AMZN's 20/1 stock split (current price $98.84 (23/03/23) )
known at the time these actions happen. This means that these costs can be considered in the decision-making process required for portfolio rebalancing.

**Transaction Costs and Portfolio Turnover**

The costs derived from the bid-ask spread are proportional costs, and increase with the amount of trading activity, so for the computation of costs for a proposed portfolio rebalancing, a major factor influencing the expense is the portfolio turnover. Plainly speaking, the more trading activity that occurs during a portfolio rebalancing, the higher the turnover, and the higher the cost. At each time \( t \) during the dynamic strategy, there is a current allocation \( w_{t-1|t} \in \mathbb{R}^N \), the subscript \( t-1|t \) indicating that this portfolio weight vector was assigned one period in the past. Now, at time \( t \), new forecasts are produced for the next period, and the allocation requires a rebalancing decision in response to the new information provided by the forecasting model.

To correctly compute the expected turnover one must know the current allocation, which would be expected to be different from the allocation last selected. During the previous holding period \( t-1|t \), asset prices would have changed, and the current allocation \( w_{t-1|t} \) will no longer reflect the true allocation of wealth; therefore, \( w_{t-1|t} \) requires re-normalising. For a portfolio that permits short selling and has a leverage constraint equal to 1, that is \( \sum_i |w_{i,t-1|t}| = 1 \), the normalised weights are

\[
\hat{w}_{i,t-1|t} = \frac{w_{i,t-1|t} (1 + y_{i,t})}{\sum_i |w_{i,t-1|t} (1 + y_{i,t})|},
\]

where \( y_{i,t} \) is the previous period’s price return for asset \( i \) and \( |.| \) denotes absolute value.

The portfolio turnover for the period \( t|t+1 \) is simply the absolute value of the change in weights between the current re-normalised allocation \( \hat{w}_{t-1|t} \) and the proposed new allocation \( w_{t|t+1} \): \n
\[
\Delta_t = |\hat{w}_{t-1|t} - w_{t|t+1}|. 
\]

Therefore, \( \Delta_t \in \mathbb{R}^N \) is a vector containing the one-period absolute change in weights for each asset in a portfolio, and its \( L1 \) norm \( \|\Delta_t\| \) is the turnover rate.

In this analysis, costs are time-varying and are determined independently for each asset in the portfolio. The current costs are obtained from the quoted bid-ask prices. Let \( P_{t}^{\text{Bid}}, P_{t}^{\text{Ask}} \in \mathbb{R}^N \).
6.4 THE NEW COST MINIMISATION OBJECTIVE FUNCTION

denote the vectors of bid and ask prices for the assets in the portfolio. Then the vector containing
the quoted half spreads in basis points is

\[ s_{i,t} = 0.5 \left( \frac{p^{Ask}_{i,t} - p^{Bid}_{i,t}}{p^{Bid}_{i,t}} + 0.5 \left( \frac{p^{Ask}_{i,t} - p^{Bid}_{i,t}}{p^{Bid}_{i,t}} \right) \right). \]

As we only observe quotes as bid and ask prices, the true prices of the assets are unknown,
and therefore the denominator used above is the midpoint between bid and ask quotes.

An expression for the transaction costs as a function of portfolio weights is then

\[ \kappa_t(\mathbf{w}_{t|t+1}) = \gamma \left\| \left( \mathbf{s}_t + t' \mathbf{c} \right) \odot \left( \mathbf{\hat{w}}_{t-1|t} - \mathbf{w}_{t|t+1} \right) \right\|, \tag{6.7} \]

with \( s_t \in \mathbb{R}^N \) the vector of half spreads in basis points, \( \gamma \in \mathbb{R} \) a scalar tuning parameter, and
finally \( c \in \mathbb{R} \) representing any additional proportional costs that are not derived from the bid-ask
prices. In the experiments in this chapter \( \gamma \) is set to 1; however, as mentioned in Olivares-Nadal
and DeMiguel (2018), this parameter can easily be tuned with cross-validation to improve
performance. For this it would be intuitive to use a portfolio performance metric such as the
negative Sharpe ratio as the loss function. Furthermore, as a catch-all for any additional costs,
experiments in this chapter use \( c = 0.001 \) (10 basis points being a value sufficient to account
for all trading volumes).

6.4 THE NEW COST MINIMISATION OBJECTIVE FUNCTION

As previously mentioned, at every time step of the portfolio strategy, when a new portfolio
choice problem arises, the initial wealth can be set to \( W_t = 1 \) with no effect on the solution to
the optimisation. Unfortunately, when ex-ante transaction costs are considered, these are costs
that must be settled at the time the trade is made and therefore will discount the investor’s
available initial wealth. As a result setting \( W_t = 1 \) in the same way as before does not account
for these expenditures and must be replaced by an initial wealth that is adjusted for costs.
To achieve this we redefine the initial wealth adjusted for costs as \( \bar{W}_t = W_t (1 - \kappa_t) \), where
\( \kappa_t = \kappa_t(\mathbf{w}_{t|t+1}) \) is a function of weights \( \mathbf{w}_{t|t+1} \) as defined in Equation (6.7).
As well as determining the portfolio moments, this formulation means the weight vector now also determines the size of the discount factor for initial wealth. Any returns over the next period will be compounded from a lower adjusted initial wealth and therefore, for the same return, this will also reduce the terminal wealth for that period. With this in mind, the next period wealth can now be expressed as a function of $w_{t|t+1}$:

$$W_{t+1} = \bar{W}_t (1 + R_{t+1}^p) = W_t \left(1 - \kappa \left(w_{t|t+1}\right)\right) \left(1 + w_{t|t+1}'y_{t+1}\right). \quad (6.8)$$

Due to properties of the CRRA utility function, $W_t$ is set to 1; therefore, the new expression for the initial wealth is simply

$$\bar{W}_t = (1 - \kappa \left(w_{t|t+1}\right)). \quad (6.9)$$

The fourth order truncated Taylor expansion of the utility function centered around the adjusted initial wealth is

$$U(W_{t+1}) \approx U^{(0)}(\bar{W}_t) + U^{(1)}(\bar{W}_t)(W_{t+1} - \bar{W}_t) + \frac{U^{(2)}(\bar{W}_t)}{2}(W_{t+1} - \bar{W}_t)^2 + \frac{U^{(3)}(\bar{W}_t)}{6}(W_{t+1} - \bar{W}_t)^3 + \frac{U^{(4)}(\bar{W}_t)}{24}(W_{t+1} - \bar{W}_t)^4. \quad (6.8)$$

Substituting derivatives of the CRRA utility function and taking expectations:

$$\mathbb{E}[U(W_{t+1})] \approx \frac{\bar{W}_t^{1-\lambda}}{1-\lambda} + \bar{W}_t^{-\lambda} \mathbb{E}[(W_{t+1} - \bar{W}_t)]$$

$$- \frac{1}{2} \left(\lambda \bar{W}_t^{-\lambda+1}\right) \mathbb{E}[(W_{t+1} - \bar{W}_t)^2]$$

$$+ \frac{1}{6} \left(\lambda + 1\right) \lambda \bar{W}_t^{-(\lambda+2)} \mathbb{E}[(W_{t+1} - \bar{W}_t)^3]$$

$$- \frac{1}{24} \left(\lambda + 2\right) (\lambda + 1) \lambda \bar{W}_t^{-(\lambda+3)} \mathbb{E}[(W_{t+1} - \bar{W}_t)^4].$$

The next step is to substitute the expressions for $W_{t+1}$ and $\bar{W}_t$ from Equation (6.8) and Equation (6.9) into the four expressions for the central moments of $W_{t+1}$. For $k = 1:4$

$$\mathbb{E}\left[(W_{t+1} - \bar{W}_t)^k\right] = \mathbb{E}\left[(W_t ((1 + R_{t+1}^p) - 1))^k\right]$$

$$= \mathbb{E}\left[((1 - \kappa_t) ((1 + R_{t+1}^p) - 1))^k\right]$$

$$= \mathbb{E}\left[((1 - \kappa_t) R_{t+1}^p)^k\right]$$

$$= (1 - \kappa_t)^k \mathbb{E}\left[R_{t+1}^{pk}\right].$$
Therefore, the central moments of the wealth distribution can be expressed as the non-central moments of the portfolio returns distribution, each multiplied by a scaling factor. The scaling factors are functions of the adjusted initial wealth raised to the power of the order of the moment. This rule follows from well-known statistical properties for expectations of random variables. The new objective function can then be written as

\[
E[\mathcal{U}_{CM}(W_{t+1})] \approx (1 - \kappa_t)^{1-\lambda} \left( \frac{1}{1-\lambda} + (1 - \kappa_t)^{-\lambda} (1 - \kappa_t) m^p_1 \right)
- \frac{1}{2} \left( \lambda (1 - \kappa_t)^{-\lambda+1} \right) (1 - \kappa_t)^2 m^p_2
+ \frac{1}{6} \left( (\lambda + 1) \lambda (1 - \kappa_t)^{-\lambda+2} \right) (1 - \kappa_t)^3 m^p_3
- \frac{1}{24} \left( (\lambda + 2) (\lambda + 1) \lambda (1 - \kappa_t)^{-\lambda+3} \right) (1 - \kappa_t)^4 m^p_4,
\]

where the \(m^p_k\) are the non-central moments of the portfolio returns distribution, as defined in Equation (6.5). This expression simplifies to

\[
E[U_{C6}(W_{t+1})] \approx (1 - \kappa_t)^{1-\lambda} \left( \frac{1}{1-\lambda} + m^p_1 - \frac{1}{2} \lambda m^p_2 + \frac{1}{6} (\lambda + 1) \lambda m^p_3 - \frac{1}{24} (\lambda + 2) (\lambda + 1) \lambda m^p_4 \right)
= A(W_t)U_{C5}(W_{t+1}),
\]

where \(U_{C5}(.)\) denotes the original objective function from Equation (6.6) as used in the experiments of Chapter 5. Therefore the new cost minimising objective is a factorisation of a function of terminal wealth (original objective Equation (6.6)), and a function of adjusted initial wealth \(A(W_t) = \bar{W}_t^{1-\lambda}\), where \(\bar{W}_t\) is defined in Equation (6.9).

### 6.4.1 The Expected Utility Surface

To illustrate how the inclusion of these costs affects the solution to the optimisation problem this subsection presents a simple example of a two-stock portfolio and a static (one period) portfolio rebalancing problem. From an initial starting allocation, an investor must decide how to enter into a new allocation given full probabilistic forecasts of the joint asset returns distributions and redistribute their wealth according to their expected utility.

The optimal allocation that omits transaction costs would clearly be the preferred choice when trading is free, and a rational investor would strive to achieve that allocation; for this reason it is referred to as the target portfolio. The allocation from which the investor is rebalancing
is referred to as the initial portfolio, and the solution to the portfolio choice problem under transaction costs is identified as the optimal portfolio.

For the purpose of this illustration, the quality of forecasts for the inputs of the optimisation problem are not important; therefore, simple sample estimates of asset moments are used. These are generated from 308 days of daily asset returns. The two assets chosen are the common stock for Tesla Inc. (TSLA) and that of Delta Airlines (DAL) over the time period 2019-12-31 to 2021-03-23. During this time these assets performed very differently, with the TSLA price increasing 8-fold and DAL, initially, losing 40% of its valuation only to gain most of it back.

The daily observations were used to generate forecasts based on simple sample estimates of the mean, covariance matrix and coskewness and cookurtosis tensors \( \hat{M}_{t+1}, \hat{\Sigma}_{t+1}, \hat{S}_{t+1}, \hat{K}_{t+1}, \) as described in Section 6.2.2. These forecasts were then used as inputs for the optimisation problem with and without transaction costs, which resulted in the two allocations 'Optimal' and 'Target' respectively.

Figure 6.1 shows the differences in the allocations across a number of different values for the transaction cost of each asset. The left column shows the positions of the initial, optimal and target portfolios plotted over the expected utility surface of the optimal strategy; the right column, for comparison, shows the same portfolios but over the surface of the optimisation problem without transaction costs. All optimisation problems in this section are unconstrained.

### TABLE 6.1: Descriptive statistics of the two assets.

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>sd</th>
<th>skew</th>
<th>kurtosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSLA</td>
<td>0.00824</td>
<td>0.0549</td>
<td>-0.00249</td>
<td>2.3009</td>
</tr>
<tr>
<td>DAL</td>
<td>0.000338</td>
<td>0.0471</td>
<td>-0.13903</td>
<td>6.4077</td>
</tr>
</tbody>
</table>

Each row of Figure 6.1 has the same initial allocation; however, different values of the bid-ask spreads are considered for each asset, shown in Table 6.2, which also shows the turnover and total cost of the rebalancing for all three panels A1, A2 and A3. The figures illustrate how including the costs as part of the process of selection shifts the maximum to a new location in the space of weights.

In panel A1, the cost of trading TSLA (horizontal axis) is extremely high and the cost of DAL (vertical axis) is much lower. Nevertheless trading in both assets is severely restricted. The distance from initial allocation to the target requires a turnover of 170%, which for the costs is deemed too high, and the target is far from the revised optimum. If the cost of trading was
FIGURE 6.1: The initial, optimal and target portfolios for varying costs. Each of the rows identified by A1, A2 and A3 has a different cost to trade the two assets. Top row A1 shows the cost of TSLA = 125bps and DAL = 10bps; in row A2 TSLA = 20bps and DAL = 30bps; and in row A3 TSLA = 10bps and DAL = 50bps. The left column displays the surface of the expected utility function which includes transaction costs and can be compared to the right column which shows the same portfolios on the surface of the utility function without transaction costs.

ignored by the optimisation problem, then the investor would have had to pay 1.4% of their wealth to transact the new positions. This fee is reduced to 0.26% by choosing the optimal portfolio. In panel A2 the spread for DAL is high and for TSLA is low relative to its attractiveness.
6.4 THE NEW COST MINIMISATION OBJECTIVE FUNCTION

TABLE 6.2: The turnover and costs relating to the portfolio rebalancing problem in Figure 6.1.

<table>
<thead>
<tr>
<th>Spread (bps)</th>
<th>Turnover (%)</th>
<th>Cost (bps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSLA</td>
<td>DAL</td>
<td></td>
</tr>
<tr>
<td>A1</td>
<td>250</td>
<td>20</td>
</tr>
<tr>
<td>A2</td>
<td>80</td>
<td>60</td>
</tr>
<tr>
<td>A3</td>
<td>20</td>
<td>10</td>
</tr>
</tbody>
</table>

TABLE 6.3: The turnover and costs relating to the portfolio rebalancing problem in Figure 6.2.

<table>
<thead>
<tr>
<th>∆t</th>
<th>Turnover (%)</th>
<th>Cost (bps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSLA</td>
<td>DAL</td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td>0.0</td>
<td>0.404</td>
</tr>
<tr>
<td>B2</td>
<td>0.73</td>
<td>0.0</td>
</tr>
<tr>
<td>B3</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

to an investor. The solution shows that the optimal position in DAL is to remain unchanged, and approximately half the target change in position for TSLA is permitted. In panel A3 more reasonable costs are considered, and the optimal allocation is much closer to the target and only a small saving is made; therefore, despite the low cost in this example, it is not optimal to trade all the way to the target portfolio.

In Figure 6.2 panels B1, B2 and B3 show portfolio solutions for problems that all have identical costs; in these examples, however, the other factor influencing the solution is varied, namely the initial weight vector. The portfolio turnover is directly computed using initial weights and therefore its value relative to the target is a major influence on turnover and cost.

All panels in Figure 6.2 show problems that contain transaction costs of 25 basis points for both assets. The panels B1 and B2 show the importance of initial values, as these can dramatically restrict the trading of certain assets. For the same transaction costs the initial positions in B1, \( \hat{w}_{t-1|t} = (0.5, 0.5) \), completely restrict the trading of TSLA, whereas in B2, where \( \hat{w}_{t-1|t} = (-0.5, -0.5) \), only TSLA is traded. Panel B3, where \( \hat{w}_{t-1|t} = (0.5, -0.5) \), is also an interesting case where the optimal solution is not to trade in either asset. Results are presented in Table 6.3, where the first section shows the change in weight of each asset’s position, \( \Delta_t = |\hat{w}_{t-1|t} - w_{t|t+1}| \), and the second section shows the portfolio turnover relative to initial weights for both the optimal and target portfolios. As seen in the previous examples, portfolio turnover is reduced in all cases. This is further seen in the final section which shows that total cost is also reduced in all examples.
6.4 THE NEW COST MINIMISATION OBJECTIVE FUNCTION

FIGURE 6.2: The unconstrained initial, optimal and target portfolios for varying initial allocations. The left column shows the surface with the inclusion of transaction costs and the right column is without costs. The cost for each asset for all examples is 25bps.

In Figure 6.3 the initial portfolio was given as $\hat{w}_{t-1|t} = (-0.5, -0.5)$, and a different subset of the original data was chosen to generate new sample estimates, so that a new optimisation problem is obtained in which the difference between initial and target portfolios is large. Figure 6.3 shows a selection of portfolios that were obtained by varying the scaling parameter $\gamma$ in Equation (6.7). This parameter has been referred to by Olivares-Nadal and DeMiguel (2018) as equivalent to
6.4 THE NEW COST MINIMISATION OBJECTIVE FUNCTION

FIGURE 6.3: The unconstrained optimal solutions for varying scaling parameter $\lambda$; the red dot indicates $\lambda=1$. Costs in basis points are TSLA = 80bps and DAL = 25bps, Expected returns for this problem were TSLA = 143.2bps and DAL = 29.4bps.

the L1 (LASSO) regularisation parameter. The figure shows that for values of $\gamma > 2.25$ there is no change in allocation for DAL, the change in position being only an increase in TSLA stock. Below this value, as $\lambda$ is reduced to zero, the portfolio weights move towards the target portfolio in a more direct way, by increasing positions in both DAL and TSLA.
6.5 EXPERIMENTS

To test the effectiveness of the new optimisation method with transaction cost minimisation, the experiments in this chapter compare and contrast the new method with the original method, as used in the previous chapter. To this end the back-tests of the two experiments in Chapter 5 are re-run, substituting the new objective function into the daily utility maximisation problem assigning weights to the assets in the portfolios. Strategies that use the new cost minimisation objective, abbreviated to C6, are then directly compared with the results from the previous chapter (C5). The only factor that varies between the C5 and C6 strategies is the objective function; all else is equal. That is, the same assignment of stocks to portfolios are used, and therefore the same models and forecasts are also used. In summary, the experiments carried out are

1. In the first experiment, the 35 U.S. equities are partitioned into the same seven sub-portfolios as used in Experiment 1 (Chapter 5). These seven portfolios are then combined at the beginning of the strategy into an equally weighted portfolio of five stock sub-portfolios. Hence the total initial wealth is shared equally at the beginning of the strategy and no funds are transferred from one sub-portfolio to any other during the test period. The models previously built in Chapter 5, for these sub-portfolios, and their forecasts, are then used as inputs for two dynamic asset allocation strategies, one using the original C5 allocation objective, and the other using the C6 objective with turnover reduction (cost minimisation). As with the experiments of Chapter 5, the strategies are tested over one year, and annualised risk / return metrics are used to gauge and compare performances.

2. For the second experiment, the average performance for a single five stock portfolio is investigated. The 100 portfolios used here were randomly drawn for Experiment 2 of Chapter 5 and the models previously built for these portfolios are backtested over a year with both the C5 and C6 objective functions. Annualised risk / return metrics are used to gauge and compare the average performances of the two strategies.
6.6 RESULTS

6.6.1 Experiment 1

The results from this experiment can be found in Table 6.4. Using the same arguments as in Chapter 5, both gross and net performance are presented for both the C5 and C6 methods. Table 6.4 contains three panels. Panel A shows results for the Markov-switching (MS-GAMLSSD) model and Panel B for the single-state (SS-GAMLSSD) model; finally, Panel C contains results for the naive equal weighted portfolio (EW) alongside the S&P 500 index, to compare performances with the market. We will start with the results for the gross performance of the two methods, where no transaction costs are deducted. It should be expected that, if trading is free, in the
TABLE 6.4: Comparative results for the cost minimisation strategy with the original results from Chapter 5 Experiment 1.

**EXPERIMENT 1**

<table>
<thead>
<tr>
<th>Panel A</th>
<th>Previous Chapter (C5)</th>
<th>Cost Minimisation (C6)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GROSS</td>
<td>NET</td>
</tr>
<tr>
<td>Switching MS-GAMLSSD</td>
<td>Portfolio Return (%)</td>
<td>53.36</td>
</tr>
<tr>
<td></td>
<td>Sharpe ratio</td>
<td>6.2079</td>
</tr>
<tr>
<td></td>
<td>Sortino ratio</td>
<td>11.3259</td>
</tr>
<tr>
<td></td>
<td>Calmar ratio</td>
<td>17.1594</td>
</tr>
<tr>
<td></td>
<td>Max drawdown (%)</td>
<td>-2.23</td>
</tr>
<tr>
<td></td>
<td>Avg turnover (%)</td>
<td>71.47</td>
</tr>
<tr>
<td></td>
<td>Annual cost (%)</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Panel B</th>
<th>Previous Chapter (C5)</th>
<th>Cost Minimisation (C6)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GROSS</td>
<td>NET</td>
</tr>
<tr>
<td>Single state SS-GAMLSSD</td>
<td>Portfolio return (%)</td>
<td>40.64</td>
</tr>
<tr>
<td></td>
<td>Sharpe ratio</td>
<td>4.918</td>
</tr>
<tr>
<td></td>
<td>Sortino ratio</td>
<td>8.123</td>
</tr>
<tr>
<td></td>
<td>Calmar ratio</td>
<td>9.7551</td>
</tr>
<tr>
<td></td>
<td>Max drawdown (%)</td>
<td>-3.15</td>
</tr>
<tr>
<td></td>
<td>Avg turnover (%)</td>
<td>73.16</td>
</tr>
<tr>
<td></td>
<td>Annual cost (%)</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Panel C</th>
<th>Return (%)</th>
<th>Sharpe</th>
<th>Sortino</th>
<th>Calmar</th>
<th>Max drawdown (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EW</td>
<td>17.91</td>
<td>1.4221</td>
<td>1.9818</td>
<td>1.9891</td>
<td>-7.77</td>
</tr>
</tbody>
</table>

The absence of any costs for rebalancing a portfolio the C5 method would discover a globally optimal portfolio, and allocations using the C6 method will always be suboptimal to that of the original method, by design. And indeed the results confirm this, showing a clear reduction in performance for the C6 method relative to the original C5 method. This expected reduction in performance is observed for both panels A and B. However, these gross results do not
6.6 RESULTS

FIGURE 6.5: Comparative wealth curves for the cost minimisation strategy (blue) against the original strategy from Chapter 5 (black) using the non-switching single state SS-GAMLSSD. Solid lines represent the wealth curves that have had transaction costs deducted, while dashed lines represent the strategies without transaction costs considered. The red line indicates the benchmark portfolio.

reflect the real world; for this we require the true (net) results, which have had transaction costs deducted. From these net results we can see a clear increase in performance when the C6 method is compared to the original C5 method.

For annual returns produced using the MS-GAMLSSD model, gross results between the C5 and C6 strategies decline from 53.36 % to 38.20%, yet net results increase from 21.19 % to 30.94%. A similar pattern is observed in the realised Sharpe ratio. From C5 to C6 gross results show the Sharpe ratio decline, whereas when transaction costs are deducted, the Sharpe ratio increases from 2.78 with the original C5 strategy to 3.41 when using the cost minimising C6 method. This pattern is consistent across the results in Panel A and Panel B of Table 6.4 for all performance metrics. With all else equal, the across-the-board increases in net performance can be attributed to the use of the C6 method.

The reasons behind this increase in performance are evident from examining the reduction in turnover achieved by using the C6 strategy. Attenuation of portfolio turnover inevitably ends in
a reduction in the costs of operating dynamic asset allocation strategies that would otherwise produce excessive amounts of portfolio turnover. From panel A, results for the MS-GAMLSSD model show that the annual cost to operate the strategy is reduced from 22.64% of portfolio value to just 5.3%, a 4.25-fold reduction. Average portfolio turnover is reduced from 71.5% to 22.3%, a 3.2-fold reduction. The discrepancy between the magnitude of the reduction between turnover and cost is evidence that the C6 strategy is effectively discriminating against the more expensive trades and favouring finding optimal solutions by trading the less expensive stocks at the time.

In Chapter 5, the results for the single-state SS-GAMLSSD model were inferior to those of its Markov-switching counterpart. Despite good gross performance, due to the high turnover and cost of operating the strategy with this model, it failed to convincingly outperform all other test models, which had significantly less turnover and were much cheaper to operate. Implementing the C5 method with the SS-GAMLSSD model, however, substantially increases performance. Annual return increases from 10.59% to 19.31% and Sharpe ratio rises from 1.45 to 2.411, with similar increases in performance as measured by Sortino ratio, Calmar ratio and maximum drawdown.

The wealth curves in Figure 6.4 and Figure 6.5 illustrate the difference in trajectories between the C5 and C6 strategies. In these figures, the dashed lines illustrate the gross wealth curves and solid lines the net. The red line is the benchmark naive equal weighted portfolio, black lines represent the use of a C5 method, and blue lines are used for C6. For both the strategies using GAMLSSD models the same pattern is observed: reduction in gross results, yet an improvement in the net performance. As with the results of the previous chapter the results here for both the C5 and C6 methods show a marked increase in performance for the Markov-switching MS-GAMLSSD over its single state version.

**TURNOVER ATTENUATION**

To better understand why we observe the improvements in performance presented above, it is useful to illustrate what is happening on an asset-by-asset level and compare the differences between the dynamics of the asset allocation for the strategy when cost minimisation is considered in the optimisation (C6), to the original strategy (C5) presented in Chapter 5, when the optimisation problem has no information about the cost of rebalancing the portfolio.

Figure 6.6 is an illustrative example taken from the results from portfolio #5 of Experiment 1. This figure contains four subplots, each of which has five elements representing each of the

---

4 Interquartile Range \( (IQR) \): \( IQR = Q_3 - Q_1 \).
FIGURE 6.6: The top panel shows box plots for the bid-ask spreads (in basis points) for the five stocks in portfolio #5 of Experiment 1. The second panel shows the corresponding box plots of the contribution to daily turnover by each stock when transaction costs are not part of the optimisation problem. The third panel shows the turnover when transaction costs are considered in the optimisation. The bottom panel shows the median reduction in daily turnover resulting from the cost minimisation method alongside the total reduction in turnover for the duration of the strategy.
TABLE 6.5: Comparison of strategy test performance for portfolio #5 using the original Chapter 5 (C5) and cost-minimising (C6) strategies. The top panel shows results without transaction costs deducted (gross). The lower panel shows the true results when transaction costs are included (net).

<table>
<thead>
<tr>
<th>PORTFOLIO #5</th>
<th>Return (%)</th>
<th>Sharpe</th>
<th>Sortino</th>
<th>Calmar</th>
<th>max-drawdown (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gross Performance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Previous Chapter (C5)</td>
<td>89.48</td>
<td>4.7055</td>
<td>8.0315</td>
<td>20.8378</td>
<td>-3.89</td>
</tr>
<tr>
<td>Cost Minimisation (C6)</td>
<td>42.17</td>
<td>2.6118</td>
<td>4.4112</td>
<td>6.8531</td>
<td>-6.03</td>
</tr>
<tr>
<td>Net Performance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Previous Chapter (C5)</td>
<td>20.96</td>
<td>1.4482</td>
<td>2.101</td>
<td>3.3351</td>
<td>-6.27</td>
</tr>
<tr>
<td>Cost Minimisation (C6)</td>
<td>30.77</td>
<td>2.0099</td>
<td>3.2812</td>
<td>4.8631</td>
<td>-6.27</td>
</tr>
</tbody>
</table>

five stocks in the portfolio. The first three of the plots in the figure are box plots, where the top and bottom edges of the box represent the first and third quantiles of a distribution; the line in the middle of the box is the median; the whiskers extending from the box represent $1.5 \times IQR^4$ above and below the 1st and 3rd quantiles, respectively; and the dots indicate any outliers outside of this range. The first panel represents the distributions of daily bid-ask spreads in basis points for each stock. The second and third panels represent the distribution of daily turnover for each of the stocks in portfolio #5. There are 252 observations in the test set.

From the top panel, it can be seen that the bid-ask spreads, and therefore costs, vary greatly between the stocks, with TG being the most expensive stock in the portfolio, by far (median = 72bps); it also has the highest amount of variability of all the assets. Additionally, CACI (median = 30bps), with a median less than half that of TG, is still much more expensive than the remaining three stocks. EBAY to the contrary is not only cheap to trade (median = 12bps) but has low variability in the bid-ask spread. Ideally for dynamic allocation strategies with high turnovers, one would not consider including stocks such as TG or CACI in the portfolio because trading them frequently would most likely contribute negatively to portfolio gains, whereas JBHT, KBH and EBAY all have narrow spreads with low variability and so are more attractive to include in dynamic portfolio strategies.

The second panel in Figure 6.6 shows the box plots for each of the stocks’ individual daily turnovers when using the original portfolio optimisation objective function that does not consider transaction costs when choosing to reallocate funds. In this case, the strategy has most of its portfolio turnover produced by changing allocation in TG, followed by CACI, the two most expensive stocks to trade. The size of the box and extended upper whiskers also indicates very high variability for TG, which includes some days with extremely large changes in position. Therefore, the data shows the original strategy frequently chooses to make large changes in
position for assets with wide bid-ask spreads, incurring high costs which will have a negative impact on overall profitability.

The third plot in Figure 6.6 shows the box plots for the individual daily turnover produced by each of the stocks in the portfolio, using the portfolio optimisation objective function presented in this chapter. The results from implementing this cost-minimising method show a marked decrease in the size of daily turnover for all assets in the portfolio. In contrast to the results in the second panel, we can see that the turnover for the most expensive stocks, TG and CACI, has been significantly reduced, making these stocks now the least traded assets in the portfolio.

Additionally, the daily turnover distributions have been squashed and now have much lower variability; this can be observed from the large reduction in the length of the whiskers of the plots for TG and CACI. For the entire time period spanned by the strategy, the turnover accrued by CACI and TG was attenuated by 88.2% and 86.6% respectively; for half of the trading days the turnover was reduced by more than 96.2% for CACI and 96.9% for TG. In contrast, the turnover attenuation levels for the stocks with narrower bid-ask spreads were much lower, e.g. EBAY’s turnover had a total attenuation of only 43.9%. The cost minimisation strategy causes a general shift in preference from trading in the most expensive (to trade) stocks to the three cheaper (to trade) ones; however, the strategy appears to still permit large changes in position of the most expensive stocks on occasion. These large changes are, however, generally now considered as outliers, as indicated by the increase in dots seen in the third panel.

To illustrate the impact that the reduced turnover has on a portfolio’s performance, Table 6.5 provides the strategy test results for portfolio #5. The table compares the cost minimisation strategy alongside the standard portfolio optimisation from Chapter 5 (C5). The top panel shows the results when transaction costs are not deducted from the wealth curve (gross). The bottom panel shows the true results when transaction costs are deducted (net).

The test results are consistent with, and follow the same pattern as those presented above. The top panel of the table shows that when all stocks are traded freely, and no transaction costs are considered, substantial portfolio returns are generated by the original (C5) strategy. Restricting trading turnover with the cost minimisation strategy reduces the portfolio’s gross annual return from 89.5% to 42.2%. However, in the real world case where transaction costs are deducted the results show that the C5 strategy is clearly outperformed by the cost minimisation strategy on all metrics bar the maximum drawdown, where the results are approximately equal. The consequence of restricting the turnover in this way is that the annual net portfolio return of 21% increases to 30.8% and the Sharpe ratio of 1.45 increases to 2.0.
6.6.2 Experiment 2

This experiment investigated the average behaviour of a strategy with a portfolio containing only five random stocks drawn from the pool of 37 available. As in the previous chapter, the average behaviour of the individual five stock portfolio strategies largely reflected the performance of the 35 stock strategy in Experiment 1. In general the same pattern, a decrease in gross performance but increase in net performance, was also seen in the averages, as already discussed in the results for Experiment 1. Panel A of Table 6.6 shows a comparison of the results for the C5 and C6 strategies based on the main MS-GAMLSSD model. The table shows the average annual return increased from 22.92% (20.11-25.73: 95% confidence interval) to 32.06% (28.98-35.14: 95% c.i.); the Sharpe ratio rose from 1.697 (1.512-1.883: 95% c.i.) to 2.032 (1.871-2.193: 95% c.i.). Additionally, there was an across-the-board substantial and significant improvement in performance for all other risk / return metrics except results of the maximum drawdown.
TABLE 6.6: Experiment 2: Comparison of the new optimisation method (C6) with the standard method (C5) of Chapter 5, showing the average strategy performance metrics from 100 randomly selected five-stock portfolios. Panel A reports the results for the MS-GAMLSSD based strategies both with and without transaction costs, and similarly, Panel B reports the results for the single-state SS-GAMLSSD based strategies. The blue entries represent the standard errors.

### EXPERIMENT 2

#### Panel A
Markov-switching MS-GAMLSSD | Previous Chapter (C5) | Cost Minimisation (C6) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GROSS</td>
<td>NET</td>
</tr>
<tr>
<td>Portfolio return (%)</td>
<td>50.8</td>
<td>22.92</td>
</tr>
<tr>
<td>(+/4.13)</td>
<td>(+/-2.81)</td>
<td>(+/-3.28)</td>
</tr>
<tr>
<td>Sharpe ratio</td>
<td>3.294</td>
<td>1.6974</td>
</tr>
<tr>
<td>(+/0.2014)</td>
<td>(+/-0.1851)</td>
<td>(+/-0.1613)</td>
</tr>
<tr>
<td>Sortino ratio</td>
<td>5.7125</td>
<td>2.7171</td>
</tr>
<tr>
<td>(+/0.4233)</td>
<td>(+/-0.3215)</td>
<td>(+/-0.3164)</td>
</tr>
<tr>
<td>Calmar ratio</td>
<td>9.9353</td>
<td>3.8041</td>
</tr>
<tr>
<td>(+/1.1578)</td>
<td>(+/-0.6055)</td>
<td>(+/-0.7719)</td>
</tr>
<tr>
<td>Max drawdown (%)</td>
<td>-5.09</td>
<td>-7.31</td>
</tr>
<tr>
<td>(+/0.45)</td>
<td>(+/-0.73)</td>
<td>(+/-0.65)</td>
</tr>
<tr>
<td>Turnover</td>
<td>68.18</td>
<td>-</td>
</tr>
<tr>
<td>(+/3.43)</td>
<td>-</td>
<td>(+/-1.96)</td>
</tr>
<tr>
<td>Annual Cost</td>
<td>-</td>
<td>19.7</td>
</tr>
<tr>
<td>(+//-1.82)</td>
<td>(+/0.45)</td>
<td></td>
</tr>
</tbody>
</table>

#### Panel B
Single state SS-GAMLSSD | Previous Chapter (C5) | Cost Minimisation (C6) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GROSS</td>
<td>NET</td>
</tr>
<tr>
<td>Portfolio return (%)</td>
<td>38.03</td>
<td>14.34</td>
</tr>
<tr>
<td>(+/3.51)</td>
<td>(+/-2.63)</td>
<td>(+/-3.28)</td>
</tr>
<tr>
<td>Sharpe ratio</td>
<td>2.5573</td>
<td>1.0996</td>
</tr>
<tr>
<td>(+/0.1893)</td>
<td>(+/-0.18)</td>
<td>(+/-0.17)</td>
</tr>
<tr>
<td>Sortino ratio</td>
<td>4.12</td>
<td>1.6713</td>
</tr>
<tr>
<td>(+/0.34)</td>
<td>(+/-0.272)</td>
<td>(+/-0.295)</td>
</tr>
<tr>
<td>Calmar ratio</td>
<td>6.4466</td>
<td>2.1655</td>
</tr>
<tr>
<td>(+/0.82)</td>
<td>(+/-0.433)</td>
<td>(+/-0.49)</td>
</tr>
<tr>
<td>Max drawdown (%)</td>
<td>-6.09</td>
<td>-8.84</td>
</tr>
<tr>
<td>(+/0.55)</td>
<td>(+/-0.9)</td>
<td>(+/-0.89)</td>
</tr>
<tr>
<td>Turnover</td>
<td>65.02</td>
<td>-</td>
</tr>
<tr>
<td>(+/3.29)</td>
<td>-</td>
<td>(+/-1.97)</td>
</tr>
<tr>
<td>Annual Cost</td>
<td>-</td>
<td>18.72</td>
</tr>
<tr>
<td>(+/1.78)</td>
<td>(+/0.45)</td>
<td></td>
</tr>
</tbody>
</table>
TABLE 6.7: Experiment 2: The complementary benchmarks for comparison with the results from Panel A and B in Table 6.6. These are the equal-weighted five-asset portfolios (EW) and the Standard and Poors 500 index (S&P 500). The blue entries represent the standard errors.

<table>
<thead>
<tr>
<th>Panel C</th>
<th>Return (%)</th>
<th>Sharpe</th>
<th>Sortino</th>
<th>Calmar</th>
<th>Max drawdown (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EW</td>
<td>15.91</td>
<td>1.0252</td>
<td>1.4768</td>
<td>1.5634</td>
<td>-11.3296</td>
</tr>
<tr>
<td></td>
<td>(+/-2.27)</td>
<td>(+/-0.14)</td>
<td>(+/-0.199)</td>
<td>(+/-0.26)</td>
<td>(+/-0.81)</td>
</tr>
</tbody>
</table>

Testing the results with a one-sided, two sample Welch test, under the null hypothesis that the statistics for the performance metrics from the C6 strategies are not greater than those generated by the C5 method, the null is rejected with 99% confidence for the tests on annual return, Sharpe ratio and Sortino ratio; for the Calmar ratio, the null is rejected with 95% confidence. However the improvement in maximum drawdown achieved from using the C6 method was unable to be shown to be statistically significant.

Statistical tests on the results from the 100 strategies using the SS-GAMLSSD model showed a statistically significant improvement of annual return with 99% confidence; for the Sharpe ratio and Sortino ratio the null was able to be rejected with 95% confidence, and for the Calmar ratio with 90% confidence. For the strategies using the SS-GAMLSSD model, again, there was no improvement in maximum drawdown between the C5 and C6 methods.

Figure 6.8 and Figure 6.9 are fan plots representing the wealth trajectories of all 100 portfolios modelled with MS-GAMLSSD. Figure 6.8 represents the results from Chapter 5 (C5) that used the original portfolio optimisation method and Figure 6.9 the results using the cost minimisation method of Chapter 6 (C6). Examining these figures side-by-side we can see a general downward shift in the gross wealth curves for the strategies using the C6 method; however, more importantly, there is a clear positive shift in the trajectory of the strategies when transaction costs are deducted. The red line in the plot is the mean average wealth curve for 100 naive equal weighted portfolios and acts as a benchmark representing the market; the average portfolio performance of the equally-weighted (EW) strategy is shown in Table 6.7, this table also shows the performance of the S&P 500.

The results can tell us that, from the list of stocks in Appendix C.2, that choosing any five at random, and opting for a naive equal weighted strategy where initial wealth is distributed
FIGURE 6.7: Wealth curves for C5 (black) and C6 (blue) strategies for the Markov-switching MS-GAMLSSD model (top) and single state SS-GAMLSSD (bottom). Dotted lines indicate the wealth curves for the relevant strategies without transaction costs deducted and the solid lines indicate the wealth processes net of transaction costs.

evenly across the five stocks, from the definition of the market as stated above, an investor would have been expected to have approximately a 50/50 chance of beating the market (as would be expected). However, choosing a five-asset portfolio which is modelled with an MS-
FIGURE 6.8: Results from Chapter 5: fan charts of the wealth curves for 100 randomly selected five-asset portfolios.
FIGURE 6.9: Results from the cost minimisation strategy: fan charts of the wealth curves for 100 randomly selected five-asset-asset portfolios.
GAMLSSD model, and operating a dynamic asset allocation strategy using the original C5 portfolio optimisation method, the results suggest an investor would be expected to beat the market 68% of the time (57.8-76.8: 95% c.i.). However using the same model and adopting the cost-minimising C6 strategy the chances of beating the market portfolio rise to 88% (79.6-93.4: 95% c.i.).

At the top and bottom end of performance, measured by annual return, if an investor were to have randomly selected a five-stock equally-weighted portfolio and held it for the year, there would be a 90% chance (82.0-94.8: 95% c.i.) that the portfolio would have been profitable that year. The high probability of profitability for a very simple buy-and-hold strategy goes to show how favourable the market conditions were for long-only portfolios during this backtest period. However, modelling the five stocks with a MS-GAMLSSD model with a dynamic long-short strategy and the original C5 method of portfolio rebalancing increased the probability of profitability to 95% (88.2-98.1: 95% c.i.). Adopting the C6 method in place of the original C5 method, however, and using the same forecasts as before for portfolio rebalancing, further increased the probability that the portfolio would be profitable to 99% (93.8-99.9: 95% c.i.). At the other end of the spectrum, we can look at the probabilities that a given portfolio would produce extremely high annual returns for the year. A naive equally-weighted buy-and-hold portfolio would have had a 1% chance (0.0-6.2: 95% c.i.) of generating an annual return in excess of 40%. Modelling a portfolio with the MS-GAMLSSD and using the C5 method for portfolio rebalancing, however, would have had a 14% (8.1-22.7: 95% c.i.) chance of generating returns in excess of 40%; however, if one had instead adopted the C6 cost minimisation method the chance would have increased to 28% (19.7-38.0, 95%: c.i.). This final set of statistics suggests that opting for the C6 method over the C5 method would have more than doubled the odds (×2.4) that a randomly drawn portfolio would achieve an extremely high return in excess of 40%. Moreover, using forecasts from the MS-GAMLSSD model and dynamically trading a portfolio with the C6 method would have increased the chance of achieving a greater than 40% annual return by 38 times that of investing in a naive equally-weighted buy and hold strategy.
6.7 CONCLUSION

In this final chapter, the issue of high portfolio turnover generated by dynamic allocation strategies encountered in the experiments of the previous two chapters was addressed, and a simple solution was presented to provide optimal portfolio allocations when considering variable transaction costs. The solution takes advantage of known information about the cost of a rebalancing at the point in time rebalancing takes place; this information is derived from the quoted bid and ask prices. The results presented for the experiments in this chapter showed clear advantages of using the turnover reduction (cost minimisation) method over the same portfolio optimisation method without cost minimisation. The portfolio strategy performance results are consistent with earlier findings (Hautsch and Voigt (2019), Olivares-Nadal and DeMiguel (2018)) that considered a portfolio optimisation using fixed (non-variable) proportional transaction costs and traditional mean-variance optimisation.

The work of this chapter differs from the previous contributions mentioned above for a number of reasons. First, this chapter presents a solution to a higher moment optimisation using a CRRA utility function. Second, the optimisation directly encompasses costs into the wealth process providing strong economic reasoning behind the optimisation objective. And finally, the costs applied were derived directly from real-world market data, and therefore the costs varied over time and across the assets in the portfolio. This allows the optimisation problem to dynamically discriminate against the more costly-to-trade stocks and suppress changes in their positions, and to instead favour those changes in allocation that present a lower cost at each point in time when the portfolio is rebalanced.

As noted in Section 6.3, with the emergence of retail trading platforms that operate on a commission-free model, the bid-ask spread is the major source of trading cost to the investor. The prices quoted by these platforms will tend to be proprietary prices and have spreads larger than the actual bid-ask spread quoted by the market, and so are a source of concern to investors. Applying the method presented here would allow investors to be able to control the costs they incur by operating what would otherwise be expensive dynamic strategies. With the same platforms offering fractional share dealing, the exact proportions of wealth as determined by the optimisation problem can now be applied exactly across stocks in a portfolio. This means that by applying this method results achievable by retail investors in the real world would be likely to reflect those presented in the results section of this chapter.
Part IV

CONCLUSIONS
Discussion

This thesis introduced a novel modelling framework designed for probabilistic forecasting of multidimensional financial time-series, specifically targeting its application in systematic portfolio management strategies. The framework’s primary function is to generate rolling one-step-ahead probabilistic forecasts of the joint distribution of returns from multiple financial assets, thereby providing the critical information for the systematic optimisation of portfolios over time. Additionally, the thesis proposed a novel optimisation approach to effectively address the significant transaction costs inherent in dynamic asset allocation strategies, enhancing their overall performance.

This final chapter will review and discuss the two core topics that underpin the contributions of this thesis: the methodology and design of the modelling framework (Section 7.1), and its application in systematic trading (Section 7.2). The discussion will critically evaluate the main findings within these areas and include suggestions for future research. The chapter will then conclude with final remarks, summarising the key contributions and broader implications of this research in the context of asset management and autonomous trading systems.
7.1 MODELLING METHODOLOGY AND DESIGN

For the development of the probabilistic forecasting models introduced in this thesis, the bivariate model of Chapter 4 and the multivariate model of Chapter 5, the Generalised Additive Model for Location, Scale and Shape (GAMLSS) distributional regression framework was extended from one that considers only univariate densities first to the bivariate case, and then to the multivariate case, respectively. For the bivariate Gaussian model of Chapter 4, this was found to be a reasonably straightforward extension. By using a Cholesky decomposition of the covariance matrix, state-dependent predictive densities could be modelled using a five-parameter GAMLSS distributional regression model. However, when using the gradient boosting method to learn the model, it became evident that this five-parameter specification presented some issues when performing hyperparameter selection, and that these issues would prove impossible to resolve if the same method was used to extend to higher dimensional distributions. Therefore, for the extension to a full multivariate framework in Chapter 5, a different approach was necessary to avoid model building intractability.

A solution to the identified intractability issue for modelling higher-dimensional time-series was found by breaking down the modelling problem into computationally manageable pieces, more specifically, by separating the joint likelihood of the multivariate distribution using a copula decomposition. This decomposition allowed the separate estimation of univariate marginal densities using a set of standard univariate GAMLSS models; by using this approach, a more complex specification could easily be implemented to model the marginal behaviour of the individual assets. To this end, a four-parameter skewed Student’s $t$ replaced the two-parameter Gaussian specification used in Chapter 4; this is well known to be a far more precise method for modelling the behaviour of financial price returns. The GAMLSS framework was also applied to model a Pair Copula Construction (PCC) density, representing the dependence structure between the individual time-series. The PCC approach offered the most flexible way possible in which to represent the time-varying dependence structure of the multivariate distribution.

The GAMLSS framework, with the previously mentioned extensions, proved to be a very effective way of modelling the probability distributions of multidimensional time-series. The use of the gradient boosting algorithm to learn the models meant that forecasts could be conditioned on a large number of exogenous variables (in the hundreds, or even thousands), making the models in this thesis stand apart from the vast majority of econometric models used to model financial time-series. As mentioned in Chapter 4, the conditional time-variation of forecasts...
achieved using GAMLSS provided a method for local time-variation; with the embedment of these models into a regime-switching framework, a further global form of time-variation was achieved, through adapting the architecture of a hidden Markov model. To the best of the author’s knowledge, there is no other time-series forecasting model that is more flexible and generalisable than the MS-GAMLSSD model introduced in this thesis, or one as capable of capturing all the statistical properties observed in financial asset price returns.

The particular architecture for regime-switching chosen for the MS-GAMLSSD model was one in which regimes were represented by different conditional multivariate densities; as such, the model switches between state-dependent models that each represent a different joint behaviour for the portfolio of assets, and can be interpreted, economically speaking, as representative of different underlying conditions in the market. However, this particular way is not the only way that regime-switching can be implemented in this context. Due to the copula decomposition of the multivariate distribution, there are a number of alternative regime-switching architectures beyond those considered in this thesis. For example, as was the case in Bernardi and Catania (2018) and Bernardi and Catania (2019), the set of univariate marginals could be static (non-switching) models, with a Markov-switching model for the dependence structure. For this particular choice, state-dependent models would be of PCC densities only. Moreover, each univariate margin model could be made to be switching with either a static or separately switching dependence model. Perhaps the most elaborate architecture would be one that switches between multivariate densities, as with the case of the MS-GAMLSSD, though with each univariate marginal also being a Markov-switching model; this would therefore be a hierarchical Markov-switching specification. To explore these various alternative architectures within the scope of this thesis was unfeasible, but could certainly be considered as a topic for future work.

In addition to the work presented in this thesis, but deemed out of scope for inclusion in it, extensive research was carried out that extended the MS-GAMLSSD model to include a conditional Markov transition density, by using the GAMLSS framework with component-wise gradient boosting. In this case, rather than treating the transition probabilities as fixed, as was done throughout this thesis, the transition density is also rendered time-varying, conditioned on exogenous variables in exactly the same way as the conditional parameters of the state-dependent models. Preliminary comparisons between this method and the MS-GAMLSSD indicated that there are benefits to pursuing this area of research further, and therefore it is suggested as future work that would look very likely to significantly improve the quality of the predictive model.
Generalised additive models, the regression framework that was used for forecasting parameters of the probability distribution, is not the only regression model that could be considered within the modelling framework used in this thesis. The MS-GAMLSSD can be adapted for the use of alternative models that fit the same criteria originally set out in the aims of the thesis. For example, gradient boosting regression trees, such as those from the packages XGBoost or LightGBM, are possible alternative candidate models that, similarly, consider high dimensional exogenous variable sets. As these models are built in a stepwise fashion, they could be used in place of the component-wise boosting GAMS. However, gradient boosting trees, being more sophisticated, would introduce increased hyperparameter selection problems, as XGBoost, for example, is well known for having a large number of tunable parameters. For this to be a viable option further work would be required to discover a way to address this issue. Alternatively, Schlosser et al. (2018) show how random forests can be used in distributional regression models, and are therefore another possible candidate to replace the boosting method used in this thesis. The authors claim there are advantages, with their method, over gradient boosting trees, as random forests are far less computationally demanding models to use. Moreover, with only one important hyperparameter to tune, random forests would be an ideal alternative to the additive models used here. But unlike the case of gradient boosting trees, in order to adapt the MS-GAMLSSD framework to incorporate random forests, substantial changes to the estimation method would have to be made; however, it still remains a possible direction for future work.

7.2 SYSTEMATIC TRADING

Chapter 4’s pairs trading experiments confirmed that the bivariate gaussian GAMLSS model was an effective method for generating rolling forecasts for the dynamic allocation of two-asset portfolios. However, due to the large variation in performances of the individually traded pairs, and high transaction costs, it was concluded that it would not be recommended that a strategy be operated in a live trading environment that considers only a single pair of assets; in contrast, running a strategy using 40 stocks, split into 20 pairs, had the effect of diversifying away the idiosyncratic risks far more effectively than the naive equal-weighting of the portfolio, and showed that even when using simple mean-variance optimisation methods the dynamic portfolio strategies were particularly effective at mitigating risk; furthermore, in spite of the high costs of operating the strategies, back-testing showed all strategies generated returns in excess of the market benchmarks.
A surprising result of the experiments of Chapter 4 was that the best-performing portfolio strategy was not based on probabilistic forecasts; instead, it was the strategy whose signals were a buy or sell decision based on directional forecasts. At a glance, this could suggest that forecasting the distribution of the portfolio’s asset returns does not provide economic benefits for this type of systematic strategy. However, this interpretation would be naive; by investigating how the portfolio returns were generated, we could see that the directional forecast strategy generated substantially less portfolio turnover than the mean-variance strategies, and therefore that strategies based on directional forecasts have significantly less operating costs and are, therefore, less expensive to run. By contrast, the two mean-variance strategies generated high turnover and were subject to high costs that affected their profitability, and led to reduced values in risk / return based performance metrics. It was discovered that the impact of portfolio turnover, and the costs incurred with the frequent rebalancing of the portfolio, can easily mask the benefits of using a high-quality statistical model alongside optimal portfolio allocations. Using this information, there were two ways in which we could progress to improve the performance of the dynamic allocation strategy. The first would be to improve the model, by improving its specification and the number of assets it can model, and the second would be to address the impact of costs. These would be the aims of the next two research chapters.

The findings of Chapter 4 were deemed to be more than sufficient to explore the extension of the modelling framework from bivariate to the full multivariate case for which, theoretically, it would be possible to consider portfolios containing an arbitrary number of assets. Therefore, the viability of the MS-GAMLSSD was tested using portfolio strategies containing five assets; this number was chosen to obtain reasonably good diversification while not requiring extensive computation time to build the models. Furthermore, in the experiments of Chapter 5, the optimisation method used for the dynamic portfolio strategy was improved, by replacing the mean-variance optimisation with one that considers higher moments of the portfolio returns distribution. Findings from the experiments showed significant performance advantage for the strategy that used the MS-GAMLSSD compared against traditional econometric models and more traditional portfolio strategies, as well as significantly outperforming the market on all measured performance indicators. It was concluded that the strategy consisting of multiple portfolios using the MS-GAMLSSD model showed sufficiently good performance in back-testing to be a viable systematic strategy for deployment in live trading.

A limitation of the experiments in Chapter 5 could be perceived as a lack of high-quality alternative methods against which to compare the MS-GAMLSSD. As was stated in this chapter’s results, the models used for comparison with the MS-GAMLSSD were, by contrast, very simple, yet well-known methods. Although able to produce the probabilistic forecasts as inputs for the portfolio strategy, these models were not specifically designed to be used for this purpose and
so unlikely to offer much in the way of competition to the MS-GAMLSSD. It was also discovered that, for most of the alternative methods, results were likely to have benefited from the bull market conditions rather than forecasting quality, evidence for which came from analysing the performances without considering trading costs. However, in lieu of any model which could be considered as a contemporary to the MS-GAMLSSD, a universally used objective benchmark for comparing performance was also included. All portfolio managers, funds, etc., will compare their performance against the market on the whole; this can be gauged by using appropriate market proxies, as was done in all three research chapters. It was therefore considered that outperformance of the market was a more informative way of validating the efficacy of the MS-GAMLSSD than comparing it against other modelling methods. This remains consistent with the goal of this thesis, which was to develop a novel class of model specifically for dynamic asset allocation strategies, and test viability and efficacy; it was not a goal to develop a method that outperformed all other possible ways of forecasting asset returns; this would not have been possible, as models similar in complexity to the MS-GAMLSSD are extremely rare in the academic literature, and it is believed that comparable classes of model would tend to be proprietary in industry, highly involved to implement, and, therefore, not possible to compare against. Additionally, after discoveries from Chapter 4, in hindsight it was perhaps an oversight not to report results for a directional forecasting model in the experiments of Chapter 5. Tests of the directional forecast strategy were in fact conducted, but as it was outperformed by the primary, MS-GAMLSSD strategy, these results were deemed not to be of value.

As noted in the introduction to this thesis, dynamic asset allocation strategies innately require substantial amounts of regular rebalancing of portfolios in order to be able to track the optimal allocation over time. It is likely that turnover generated by this rebalancing can lead to an undesirable accumulation of transaction costs that are inhibitory to a strategy’s feasibility. Significant performance impairment specifically attributable to transaction costs was confirmed in the back-testing results in both Chapter 4 and Chapter 5’s experiments; with the observed high portfolio turnover, in general a greater proportion of the revenues from the strategies was being spent to transact the rebalancing than was being realised as profits. Because of the significant impact transaction costs have on these type of strategies it was considered of utmost importance to recreate the costs in as realistic a way as possible; as such market data was used to derive an accurate approximation for transaction costs. This makes the research of this thesis stand out from the vast majority of published literature focusing on financial market trading, where it has unfortunately become common to assume that transaction costs can be fixed over time, or be omitted altogether. Only a good quality, realistic, approximation of costs is sufficient to ensure we have a methodology that can be applied with confidence to live trading accounts.
Throughout all experiments in this thesis, transaction costs were computed from market prices, which meant that, for every asset, at every point in time, the cost of opening or closing a position was unique for each observation, and was observed to vary significantly across the different assets and over time. The aim of Chapter 6 was to investigate a way of improving the performance of systematic trading strategies by taking advantage of the different prices in transactions. This was achieved by redesigning the optimisation problem to exploit knowledge of costs. In a similar spirit to the well-known profit maximisation problem in microeconomic theory, a new objective function was derived to optimise the dynamic allocation strategy. The optimisation problem became one that minimises the cost of trading while simultaneously maximising expected utility. The impact of using the novel cost minimisation method was notable, significantly improving strategy performance compared with the original method. An investigation showed that the strategy would forgo changing positions in assets which, at the time, were expensive to trade, and instead find allocations that were more cost-effective to rebalance to, yet still ones that were optimal when transaction costs were considered.

A major strength of the cost minimisation method is its simplicity; only a minor alteration to the higher moment optimisation objective function was required to accommodate knowledge of the rebalancing costs. The results showed that this had a major effect on improving a strategy’s performance, and has potential to improve profitability in live trading situations where turnover and costs have previously prevented strategies from being feasible. Chapter 6, however, left an unanswered question, one that can be taken up in further studies of the higher-order moment optimisation, namely what is the impact of the tuning parameter on further improving portfolio performance? To recap, the tuning parameter $\gamma$ controls the influence of the cost component within the optimisation objective function, and as a form of regularisation, can be used to constrain the amount of portfolio rebalancing, see Figure 6.3 for an illustration. In the case of all back-testing experiments, this parameter was set to one, therefore did not have an effect on the allocations; however, it is very likely that different predictive models respond very differently to market data and have different properties, such as agility and responsiveness, and therefore would be expected to produce different amounts of portfolio turnover during the strategy. This means that there should be a benefit to tuning this parameter to suit the particular predictive model being used. Bespoke tuning of this parameter can easily be achieved with model selection using a time-series validation method. A suggested objective function for model selection would be a portfolio performance metric such as Sharpe or Sortino ratio. It would be of interest to study how tuning this parameter would affect the results of Chapter 6, and this is suggested as further work.
7.3 CONCLUDING REMARKS

Very many published works in computational finance appear to give impressive results. However, these tend to be in the restricted context of the paper in which these results appeared, with many potential, undisclosed or unsuspected, sensitivities to the nuances of the datasets and assumptions in experimental methodology. To be of real value in the wider world, a systematic trading system, as has been developed throughout this thesis, needs to be robust to these details. The experiments of this thesis have aimed to operate without restrictive assumptions, and thus the methods developed have the necessary robustness for deployment in practice, as a viable method of automation for the management of portfolios of financial assets. From the findings in this thesis, we can conclude that the methods introduced meet the standards for practical application set out in the research aims, and therefore that dynamic asset allocation strategies, as introduced in this thesis, can be deployed, with confidence, to live trading.
Part V

APPENDIX
Cryptocurrency Pairs Study

a.1 CRYPTOCURRENCY PAIRS STUDY

Bitcoin, the original cryptocurrency, had its conception in a white paper by Nakamoto (2008). The original motivation behind Bitcoin was to introduce a novel payment system that could easily facilitate electronic transactions, but would also have many advantages over making payments with physical cash. Bitcoin was later joined by Ethereum in 2016 and, to date, has been followed by the issuance of over 4,000 other variants, known as *altcoins*. Despite the vast number of new coins issued, there exist only a handful of major cryptocurrencies considered as serious currency. However, through the many exchanges that emerged to enable transactions, cryptocurrencies have become readily-tradable assets and have attracted an unprecedented attention by speculative investors.¹

Through the use of blockchain technology, these currencies rely on principles of cryptography to ensure the security of transactions; as a result of this architecture they can be decentralised, meaning they are not regulated nor controlled or monitored by any third party or governing body. A result of decentralisation through blockchain is that the values of cryptocurrencies are not backed up by any assets, governmental promises or credibility of issuing body, and therefore these assets, traditionally speaking, have no real intrinsic value. A consequence of

¹ This study was undertaken in late 2018; since the time of writing, the industries around digital assets, and the digital assets themselves, have evolved significantly, and many of the statements in the introduction to this appendix would nowadays be considered outdated
decentralised currencies is that prices of these currencies have no mechanisms available to ensure their stability, and as a result we have seen some uniquely dysfunctional behaviour in cryptocurrency markets over the past few years.

The lack of any intrinsic value behind these currencies means the price of major cryptocurrency pairs is only determined by supply and demand, i.e. the demand users have for using them on a transactional level, and also demand from speculative investors. It is the latter of these two demand sources that in the recent past has spurred an overly enthusiastic increase in interest in the major cryptocurrencies. The impact of speculative behaviour has seen the creation and inevitable bursting of an economic bubble in these markets, a bubble that was so inflated that at its 2018 pre-crash peak Bitcoin reached 2,300% of its initial price, with some other major coins exceeding this growth.

It has been suggested in Hale et al. (2018) that the 'crypto bubble' came to an end after the opening of Bitcoin futures markets on 17th December 2017. With these futures contracts available to speculators, the more pessimistic of them were able to act upon their negative expectations for the currencies’ future prices and enter into short positions more easily than before. For the first time there was significant downside resistance to Bitcoin’s price growth, which inevitably triggered a major sell-off in Bitcoin. This had an effect across other major cryptocurrency markets resulting in the ‘crypto crash’.

In this study we focus on the two major players in the cryptocurrency market, namely Bitcoin (BTC) and Ethereum (ETH). The time series used are the raw OHLC (open, high, low, close) prices, along with trading volume for BTC/USD and ETH/USD currency pairs, all prices being denominated in U.S. Dollars. The observations are sampled every hour on the hour for 24 hours a day. The dataset spans from 1st June 2017 to 9th September 2018 and is split into a training period and test period by a partition on the 1st July 2018, allowing for 104 days and 5 hours of testing. In total there are 10485 observations, with 7919 used for training and 2501 in the test set. All data was sourced from the Kraken exchange.
Figure A.1 is a plot of the closing mid-price of BTC/USD and ETH/USD. The test region is marked right of the dashed line. It is clear from this figure there is an extreme behaviour of prices in the training region. We can see the last few months of the bubble, displaying exponential price growth during this period; following this, from the start of 2018, the crash occurs as a sharp and highly volatile decline which, despite an eventual decrease in volatility, continues into the test region. The test region alone shows a drop in value for BTC/USD of 15.85%, and an even more serious loss for ETH/USD of 66.94%, more than two-thirds of its value.

It is clear from observing the price time series in Figure A.1 that the behaviour of the BTC/USD market changes significantly after 17/12/2017, transitioning from a bull regime to a bear regime, with the ETH/USD market joining BTC in the crash 27 days after this initial date. Here, this date is used as a cut-off point, and a summary examination of the empirical distributions is presented.
Figure A.2 shows the empirical density of BTC/USD and ETH/USD; the histogram contains all data from training and test sets superimposed with density plots for the price returns before and after the high points of each series, 17/12/2017 for BTC, and 14/01/2018 for ETH. The Normal density is included to illustrate how far the empirical distribution deviates from normality; leptokurtosis and positive skew are present in both. The latter is extremely interesting as it goes against what we can regularly observe in the distributions of other asset classes. Usually in financial returns data a negative skew / positive mean is observed over the long run; here we see a positive skew in both bull and bear regimes with the skew decreasing for BTC/USD during the crash, but substantially increasing for ETH/USD. Table A.1 shows the mean, variance, skewness and kurtosis of the empirical distributions for both assets’ pre- and post-crash high points in their corresponding time-series.
Furthermore, asymmetry in correlation between the two currencies is observed. Figure A.3 shows a scatter plot of the data superimposed with contours of the fitted density. The left image shows data from before the Bitcoin price’s high point, i.e. the period of growth prior to the crash, and the right hand plot is for the period of sell-off in both markets. It can be observed that (on average) the two assets are more highly correlated during the sell-off period than they are during the growth period, which is consistent with the stylised facts found in more traditional currency exchange rates (Patton (2006)).

**A.1.1 Joint Probabilistic Forecasting of Cryptocurrency Time-series**

The bivariate MS-GAMLSS from Chapter 4 is applied to model the time-varying joint distribution of price returns for Bitcoin and Ethereum to U.S. Dollar exchange rates. Both in- and out-of-sample forecasts of volatility and correlation are presented, and compared to standard methods of modelling these quantities. As discussed previously, the training set includes the period leading up to the ‘crypto crash’ of 2018, where exponential growth can be observed in both currencies, along with the subsequent period of prolonged sell-off in both markets. For this experiment we look at hourly price returns data, with the same features generated from OHLCV as defined above.

For the in-sample forecasts, we have the opportunity to see how the MS-GAMLSS can track changes in the markets during a period of extraordinary behaviour, the period leading up to the crypto crash, where exponential growth is observed, and the subsequent period of sell-off in the market after the bubble has burst. For the out-of-sample forecasts, statistical tests are used to
A.1 CRYPTOCURRENCY PAIRS STUDY

compare the quality of forecasts from the MS-GAMLSS with the dynamic conditional correlation model of Engle (2002) and the asymmetric extension (Cappiello, Engle, and Sheppard (2006)). Both benchmark models use an AR(1) mean model and GARCH(1,1) for variance, and assume a Student’s t error distribution function.

A.1.2 Empirical Examination of Volatility and Correlation

In the in-sample plots of Figure A.4 the red marker identifies the point in time where futures markets for Bitcoin became available. From this point on, it has been suggested that pessimistic speculation within these markets put excessive downward pressure on Bitcoin’s price and sparked the beginning of the crash. The squared volatility series shows that, leading up to the marked point, several large volatility spikes punctuate periods of relative low volatility. The spike in September can be seen in both BTC/USD and ETH/USD markets and caused the two to become highly correlated for a time. The coinciding price series for this period show an increase in price of both currencies, in excess of the underlying long-term trend, which was then followed by a price correction. Prior to this period, we saw cryptocurrencies becoming mainstream through publicity from news reports etc. This may well have been the cause of the increase in price growth in September, as new speculators entered the market looking to recreate gains of previous cryptocurrency market participants.

Following September’s spike, there is a relatively tranquil month for both BTC/USD and ETH/USD, with correlation decreasing, leading up to a few weeks prior to the 17/12 shock, where we can see a marked increase in volatility in the BTC/USD market with lesser effects in ETH/USD. After 17/12, however, both markets see a series of three large volatility spikes indicating an increase in activity. The three spikes coincide exactly; however, they have different effects on prices. Post 17/12, during the first spike, the BTC price undergoes a sharp drop-off, whereas ETH makes large gains. This translates into the correlation between the two decreasing into negative. This behaviour can also be observed in the price time series Figure A.1. It appears that as money leaves BTC/USD it is transferred into ETH/USD.

This behaviour persisted for approximately four weeks, after which ETH/USD had made large gains from the decline in BTC/USD, until 14/01/2018 (vertical black line in left column in Figure A.4), where the second spike in volatility occurs; this date is also when ETH posted its highest price against the dollar. Unlike the first spike, which was far more pronounced for the BTC/USD market, the second spike affects ETH/USD the most; this volatility spike is a consequence of a sharp sell-off in the currency, and signifies the bursting of the ETH/USD bubble and the beginning of the crash.
It is this date, 14/01/2018, that the crypto crash can be said to have commenced. As is common during market downturns and crashes, markets become highly correlated with each other; after 14/01 this can be observed in the in-sample correlation plot in Figure A.4. From this point on, the BTC/USD and ETH/ISD markets are very highly correlated, with a coefficient in excess of 0.8, and are both crashing in price. The third spike signifies the end of an attempted price rally by both currencies; the rally fails and the sell-off continues over the next few months. During this time of downward trend, the two currencies remain highly correlated.

The correlation asymmetry, where we observe stronger correlation during periods of market downturns than in equivalent periods of growth, is very pronounced in this example. From the negative price correction in September to the crash, BTC and ETH have stronger correlation when their prices are in decline. The dynamic conditional correlation model (DCC) cannot account for such behaviour, nor can a single state bivariate normal GAMLSS. These findings are an argument for the use of the ADCC (asymmetric DCC) or an MS-GAMLSS to model these time series.

### A.1.3 Forecast Evaluation

In this subsection evaluation methods are applied to assess whether the obtained series of volatility forecasts coincide with a corresponding series of ‘true’ realised values. The diagnostic test is the Mincer–Zarnowitz volatility regression, which can be used to assess the quality of the forecasted conditional variances.

For a general forecasting problem, the Mincer–Zarnowitz regression is

\[
\hat{y}_{t+1} = \alpha + (1 + \beta)\hat{\sigma}_{t+1}^2 + \epsilon_{t+1},
\]

(A.1)

where \(\hat{y}_{t+1|t}\) is the conditional estimate of the quantity we wish to test and \(y_{t+1}\) its realised value. It is easy to see from Equation (A.1) that both coefficients equalling zero will give a good indication that the forecasted series is close to the realised values. For a series of volatility forecasts we set \(\hat{y}_{t+1|t} = \hat{\sigma}_{t+1|t}^2\); however, for the dependent variable it is not clear what to use as the true value of volatility. Along with correlation, volatility is inherently a latent quantity; as such there is not an observable ‘true’ value to use for the dependent variable in the above regression, and hence, in lieu of any ‘true’ values, a proxy must be used for the volatility series.

A simple solution is to use the series of squared returns as the proxy for the dependent variable. This choice, however, has many negatives; primarily it assumes a mean of zero for the returns series but, also, it essentially means we are representing the ex-post variance at each timepoint,
FIGURE A.4: The in- and out-of-sample squared volatility and correlation forecasts for the BTC/USD and ETH/USD time series. The series are smoothed with a five-day simple moving average for readability. The MS-GAMLSS forecasts are compared with the DCC, with the asymmetric extension to the DCC (ADCC), and with sample variance and correlation estimates (calculated using a simple moving average).

with an estimate computed using just a single observation, making the squared returns proxy an extremely noisy one. A preferred solution would be to use the ‘realised’ variance (Shephard
TABLE A.2: Mincer–Zarnowitz regression summary (in-sample); p-values for the $a = 0, b = 1$ hypothesis are included only in the cases where it cannot be rejected with $> 90\%$ confidence.

<table>
<thead>
<tr>
<th>IN-SAMPLE</th>
<th>BTC</th>
<th>ETH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>DCC</td>
<td>-3.78E-05</td>
<td>1.310</td>
</tr>
<tr>
<td>ADCC</td>
<td>9.28E-07</td>
<td>0.971</td>
</tr>
<tr>
<td>MS-GAMLSS</td>
<td>-7.50E-06</td>
<td>1.04</td>
</tr>
</tbody>
</table>

p-value 0.067 p-value 0.0127

TABLE A.3: Mincer–Zarnowitz regression summary (out-of-sample); p-values for the $a = 0, b = 1$ hypothesis are included only in the cases where it cannot be rejected with $> 90\%$ confidence.

<table>
<thead>
<tr>
<th>OUT-OF-SAMPLE</th>
<th>BTC</th>
<th>ETH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>DCC</td>
<td>-8.29E-05</td>
<td>2.290</td>
</tr>
<tr>
<td>ADCC</td>
<td>-4.10E-05</td>
<td>1.960</td>
</tr>
<tr>
<td>MS-GAMLSS</td>
<td>-1.36E-05</td>
<td>1.09</td>
</tr>
</tbody>
</table>

and Sheppard (2010)), constructed with a large number of high-frequency observations; for a definition of realised variance and a review of other possible proxies for volatility, see Andersen et al. (2006). However, the high-frequency data required to create a volatility proxy using realised variance was unavailable at the time for these markets, and so the tests were performed using the less-preferred series of squared returns.

Testing the hypothesis, $a = 0, b = 1$ is a test for the accuracy of the forecast with respect to the volatility proxy. For the out-of-sample forecasts, this test produced disappointing results for all three models; see Table A.3. The hypothesis was rejected with 90% confidence in all cases. For the in-sample forecasts, we reject the hypothesis for forecasts generated by the DCC and ADCC models, but cannot reject the hypothesis for the MS-GAMLSS forecasts with 93% confidence for the in-sample BTC/USD and 98% confidence for ETH/USD.

Comparing the regression coefficients for the competing models, out of sample the GAMLSS forecasts produce slope coefficients that are closer to 1 than those from the ADCC and DCC models. In most cases we can see $b > 1$ which, according to Andersen et al. (2006), is indicative
that the proxy is more volatile than the forecasted series; this leads us to assume that perhaps the squared returns series may be too noisy to act as a valid proxy. For the MS-GAMLSS it may suggest that a known drawback to prediction within Markov-switching frameworks is causing there to be too much uncertainty within the predictive distribution used to mix forecasts from component models, such that the resulting forecasts will have a tendency to underestimate highs and overestimate lows; this behaviour can be observed in the MS-GAMLSS forecasts in Figure A.4.

When a forecast series is found to be sub-optimal, as in the cases above, then, from the results of the Mincer–Zarnowitz regression, forecasts can be calibrated, i.e. shifted and scaled in accordance with the estimates of the regression coefficients.

While all the models found this test difficult, it should be noted as a positive result that the forecasts generated by the MS-GAMLSS have much better regression fits than those from the ADCC and DCC models. Therefore, notwithstanding the biases of these forecasts, a greater proportion of the variation is explained by the forecasts from the MS-GAMLSS model than those generated by the ADCC or DCC models. For perspective, it is documented in Andersen et al. (2006) that when using the squared returns series as a volatility proxy, one should typically expect ‘unspectacular’ $R^2$ values of around 5% to 10%. The $R^2$s of 32% and 36% for the MS-GAMLSS method for out-of-sample volatility forecasts for BTC and ETH, respectively, clearly exceed this range.
Appendix: Feature Space Generation

The data used in the experiments of Chapters 5 and 6 consist of multiple data sets falling into two categories. The first category contains the asset-specific datasets used for estimation of the marginal models and the second category contains the datasets used for pair-wise relationship modelling. All datasets contain a high-dimensional set of covariates generated by extracting features from raw market data. This was done by performing operations on the time-series of OHLCV market price data.

For each asset we have the set \( \{O_t, H_t, L_t, C_t, V_t\}_{t=1}^T \) of time-series of open, high, low, and closing prices, as well as the time-series of trade volume occurring between the opening and closing times.

The features extracted from the market price data fall into a number of sub-types; these are

- Moving average cross overs
- Bollinger band cross overs
- Candlestick functions
- Volatility, correlation and relative betas
- Technical indicators

In this appendix the feature extraction method is explained.

For a \( D \) dimensional time-series, there are a total of \( 20(D - 1) + 202 \) features for each dimension. For the experiments in this chapter this equated to 282 for each financial asset.
## b.1 FEATURES

### MOVING AVERAGE CROSSOVERS

| Number of features: 34 | Inputs: $C_t$. | Parameters:  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Computation</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>--------------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXO_sl_1 (10)</td>
<td>$\log \left( \frac{\text{SMA}(C_t, s)}{\text{SMA}(C_t, l)} \right), \forall s &lt; l.$</td>
<td>Simple moving average crossovers with short $s$ and long $l$ lookbacks.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXO_sl_2 (10)</td>
<td>$\log \left( \frac{\text{EMA}(C_t, s)}{\text{EMA}(C_t, l)} \right), \forall s &lt; l.$</td>
<td>Exponential moving average crossover</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXO_sl_3 (10)</td>
<td>$\log \left( \frac{\text{EMA}(C_t, s)}{\text{EMA}(C_t, l)} \right) - \log \left( \frac{\text{EMA}(C_{t-1}, s)}{\text{EMA}(C_{t-1}, l)} \right), \forall s &lt; l.$</td>
<td>Exponential moving average crossover rate of change</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXO_sl_4 (4)</td>
<td>$\log \left( \frac{C_t}{\text{EMA}(C_t, s)} \right), \forall s.$</td>
<td>Current price crossover exponential moving average</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### BOLLINGER BAND CROSSOVERS

| Number of features: 32 | Inputs: $C_t$. | Parameters:  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Computation</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>--------------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBXO_sl_1 (10)</td>
<td>$\log \left( \frac{\text{BBupper}(C_t, s)}{\text{EMA}(C_t, l)} \right), \text{for } s &lt; l.$</td>
<td>Long upper Bollinger band crossover short exponential moving average</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Computation</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBXO_s_2</td>
<td>[ \log \left( \frac{BB_{upper}(C_t, s)}{C_t} \right) \text{, for all } s. ]</td>
<td>Current price crossover short upper Bollinger band</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBXO_s_3</td>
<td>[ \log \left( \frac{C_t}{BB_{lower}(C_t, s)} \right) \text{, for all } s. ]</td>
<td>Current price crossover long lower Bollinger band</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBXO_sl_4</td>
<td>[ \frac{BB_{upper}(C_t, l) - EMA(C_t, s)}{BB_{upper}(C_t, l) - BB_{lower}(C_t, l)} \text{, for } s &lt; l ]</td>
<td>Short exponential moving averages position relative to trading range</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBXO_sl_5</td>
<td>[ \frac{BB_{upper}(C_t, s) - C_t}{BB_{upper}(C_t, s) - BB_{lower}(C_t, s)} \text{, for all } s. ]</td>
<td>Current price position relative to trading range</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### CANDLESTICK FUNCTIONSS

Number of features: 46

**Inputs:** \(O_t, H_t, L_t, C_t, V_t\)

**Parameters:** \(l \in \{8, 16, 32\}\).

<table>
<thead>
<tr>
<th>Name</th>
<th>Computation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNF_l_1</td>
<td>[ \log \left( \frac{C_t}{\min(L_{t-l-1})} \right) \text{, for all } l ]</td>
<td>Current price in relation to the lowest price over the past (l) days.</td>
</tr>
<tr>
<td>CNF_l_2</td>
<td>[ EMA \left( \log \left( \frac{C_t}{\min(L_{t-l-1})} \right) , l \right) \text{, for all } l ]</td>
<td>Moving average of the above.</td>
</tr>
<tr>
<td>CNF_l_3</td>
<td>[ \log \left( \frac{C_t}{\max(H_{t-l-1})} \right) \text{, for all } l ]</td>
<td>Current price in relation to the highest price over the past (l) days.</td>
</tr>
<tr>
<td>CNF_l_4</td>
<td>[ EMA \left( \log \left( \frac{C_t}{\max(H_{t-l-1})} \right) , l \right) \text{, for all } l ]</td>
<td>Moving average of the above.</td>
</tr>
<tr>
<td>CNF_l_5</td>
<td>[ \frac{SMA(C_t, l)}{\min(L_{t-l-1})} ]</td>
<td>Average closing price in relation to the lowest price over the past (l) days.</td>
</tr>
<tr>
<td>CNF_l_6</td>
<td>[ \frac{SMA(C_t, l)}{\max(H_{t-l-1})} ]</td>
<td>Average closing price in relation to the highest price over the past (l) days.</td>
</tr>
<tr>
<td>CNF_l_7</td>
<td>( \log \left( \frac{\min(L_{t-1:t})}{\max(H_{t-1:t})} \right) )</td>
<td>Difference between highest and lowest prices over the past ( l ) days.</td>
</tr>
<tr>
<td>CNF_l_8</td>
<td>( \frac{H_t - C_t}{H_t - L_t} )</td>
<td>Closing price in relation to session’s trading range.</td>
</tr>
<tr>
<td>CNF_l_9</td>
<td>( SMA \left( \frac{H_t - C_t}{H_t - L_t}, t \right) )</td>
<td>Closing price in relation to session’s trading range (smoothed).</td>
</tr>
<tr>
<td>CNF_l_10</td>
<td>( \frac{\max(H_{t-1:t}) - C_t}{\max(H_{t-1:t}) - \min(L_{t-1:t})} )</td>
<td>Closing price in relation to past ( l ) sessions’ trading range.</td>
</tr>
<tr>
<td>CNF_l_11</td>
<td>( SMA \left( \frac{\max(H_{t-1:t}) - C_t}{\max(H_{t-1:t}) - \min(L_{t-1:t})}, t \right) )</td>
<td>Closing price in relation to past ( l ) sessions’ trading range (smoothed).</td>
</tr>
<tr>
<td>CNF_l_12</td>
<td>( \log \left( \frac{\max(H_{t-1:t})}{\min(L_{t-1:t})} \right) )</td>
<td>Spread of trading range over lookback period.</td>
</tr>
<tr>
<td>CNF_l_13</td>
<td>( \log \left( \frac{H_t}{L_t} \right) )</td>
<td>Previous session and average trading rage.</td>
</tr>
<tr>
<td>CNF_l_14</td>
<td>( SMA \left( \log \left( \frac{H_t}{L_t} \right), t \right) )</td>
<td>Previous session and average trading rage (smoothed).</td>
</tr>
<tr>
<td>CNF_l_15</td>
<td>( \log \left( \frac{C_t}{C_{t-j}} \right) )</td>
<td>Price rate of change</td>
</tr>
<tr>
<td>CNF_l_16</td>
<td>( \log \left( \frac{V_t}{V_{t-j}} \right) )</td>
<td>Volume rate of change</td>
</tr>
<tr>
<td>CNF_l_17</td>
<td>( \log \left( \frac{V_t}{SMA(V_t, l)} \right) )</td>
<td>Last session’s volume relative to average over lookback period.</td>
</tr>
<tr>
<td>CNF_l_18</td>
<td>( \log (V_t) )</td>
<td>Volume (log).</td>
</tr>
<tr>
<td>Name</td>
<td>Computation</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>VCOB_1_1 ( (D-1)^4 )</td>
<td>( wcorr \left( \log \left( \frac{C^d_t}{C^{d-1}_t} \right), \log \left( \frac{C^l_t}{C^{l-1}_t} \right), t \right), ) for all ( u = D - d ) and for all ( l ).</td>
<td>A vector of time-weighted rolling correlations of an asset’s return with the returns of all other assets in a portfolio.</td>
</tr>
<tr>
<td>VCOB_1_2 ( 4 )</td>
<td>( wvar \left( \log \left( \frac{C^d_t}{C^{d-1}_t} \right), t \right), ) for all ( l ).</td>
<td>Time-weighted rolling variance.</td>
</tr>
<tr>
<td>VCOB_1_3 ( (D-1)^4 )</td>
<td>( \beta_t := wbeta \left( \log \left( \frac{C^d_t}{C^{d-1}<em>t} \right), \frac{1}{D} \sum</em>{u=1}^{D} \log \left( \frac{C^{</td>
<td>u</td>
</tr>
<tr>
<td>VCOB_1_4 ( (D-1)^4 )</td>
<td>( \beta_t - \beta_{t-1}, ) for all ( l ).</td>
<td>Change in market beta over ( l ) time-steps.</td>
</tr>
<tr>
<td>VCOB_1_5 ( 4 )</td>
<td>( \log \hat{\sigma}<em>{t+1}, ) where ( \hat{\sigma}</em>{t+1} := \text{GARCH}(1, 1) ).</td>
<td>GARCH(1,1) forecast for volatility.</td>
</tr>
<tr>
<td>VCOB_1_6 ( (D-1)^4 )</td>
<td>( \log \hat{\rho}^{</td>
<td>d,u</td>
</tr>
<tr>
<td>VCOB_1_7 ( (D-1)^4 )</td>
<td>( \log \hat{\rho}^{</td>
<td>d,u</td>
</tr>
<tr>
<td>Name</td>
<td>Computation</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>TIXO_sl_1 (10)</td>
<td>( MACD(s, l), \forall s &lt; l. )</td>
<td>Moving average convergence: difference between two moving averages of slow and fast periods ((s, l)).</td>
</tr>
<tr>
<td>TIXO_sl_2 (10)</td>
<td>( \frac{MACDSL(s, l, n)}{MACD(s, l)} )</td>
<td>Numerator is the ( MACD ) signal line: the moving average of ( MACD(s, l) ) of past ( n ) periods.</td>
</tr>
<tr>
<td>TIRSI_s (4)</td>
<td>( RSI(s) = 100 \left[ 1 - \frac{100}{SMA(C_{up}^t, s) - SMA(C_{down}^t, s)} \right], \forall s. )</td>
<td>Relative strength index: compares the days that stock prices finish up ( C_{up}^t ) against those periods that stock prices finish down ( C_{down}^t ).</td>
</tr>
<tr>
<td>TIOS_FASTK_s (4)</td>
<td>( FASTSOK_t(s) := \frac{C_t - \min(L_{t-s}^t)}{\max(H_{t-s}^t) - \min(L_{t-s}^t)}, \forall s )</td>
<td>Fast stochastic oscillator: The percentage measure of the last close price in relation to the highest high and lowest low of the last ( s ) periods (true range).</td>
</tr>
<tr>
<td>TIOS_FASTD_sl (10)</td>
<td>( FASTSOD(s, l) := SMA(FASTSOK(s), l), \forall s, l. )</td>
<td>Moving average of ( FASTSOK_t(s) ).</td>
</tr>
<tr>
<td>TIOS_SLOWK_s (4)</td>
<td>( SLOWSOK(s) := SMA(FASTSOK(s), 3), \forall s. )</td>
<td>Moving average of ( FASTSOK_t(s) ), with 3-period smoothing.</td>
</tr>
<tr>
<td>Feature</td>
<td>Formula</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>TIOS_SLOWD_s (4)</strong></td>
<td>$SLOW SOD(s, 3)$ $\implies$ $SMA(SLOW SOK(s), 3), \forall s.$</td>
<td>Double smoothed $FAST SOK_t(s),$ with another 3-period smoothing.</td>
</tr>
<tr>
<td><strong>TIOS_FASTR_sl (10)</strong></td>
<td>$FAST SOK(s)$ $\implies$ $FAST SOD(s,l)$</td>
<td>Fast stochastic oscillator relative to its average.</td>
</tr>
<tr>
<td><strong>TIOS_SLOWR_s (10)</strong></td>
<td>$SLOW SOK(s)$ $\implies$ $SLOW SOD(s,l)$</td>
<td>Slow stochastic oscillator relative to its average.</td>
</tr>
<tr>
<td><strong>TIOS_WILL_s (4)</strong></td>
<td>$WILLIOS(s) := \frac{\max(H_{t-s} - C_t) - C_t}{\max(H_{t-s} - L_{t-s}) - \min(L_{t-s} - C_t)} \forall s.$</td>
<td>Williams indicator: the calculation is similar to the stochastic oscillator with a scale from 0 to 1.</td>
</tr>
<tr>
<td><strong>TIMFI_s (4)</strong></td>
<td>$MFI(s) = 100 - \frac{100}{1 - \frac{SMA(C_t \cdot V_t,s)}{SMA(C_t \cdot V_t,s)}} \forall s.$</td>
<td>Money flow index: measures the strength of money flow (MF) in and out of a stock. Same as RSI but with volume weighted prices.</td>
</tr>
<tr>
<td><strong>TIVOL_CHAKIN_s (4)</strong></td>
<td>$\log \left( \frac{EMA(H_{t-s} - L_{t-s})}{EMA(H_{t-s} - L_{t-s}, s)} \right)$</td>
<td>Chaikin volatility: evaluates the changes in the range between high and low prices.</td>
</tr>
<tr>
<td><strong>TIOS_ADLPIVOT_s (4)</strong></td>
<td>$\frac{ADL_t(s) - ADL_{t-s}(s)}{C_t - C_{t-s}} \times \text{sign}(C_t - C_{t-s}), \text{where } ADL(s) \implies \sum_{n=0}^{s-1} MFM_{t-n} \times V_{t-n} \text{ and } MFM_t := \frac{2C_t - H_t - L_t}{H_t - L_t}, \forall s.$</td>
<td>Accumulation/distribution line where (MFM) is the money flow multiplier. Momentum diverging with price warns of a pivot (sign reveals the price direction).</td>
</tr>
</tbody>
</table>
U.S. Equities Details

The tables below list the details of the companies whose stock was traded in the experiments of this thesis. The equities pool was randomly drawn from stocks that have been listed and continuously traded on the NYSE, AMEX and NASDAQ exchanges from 2006 to at least 2018.

c.1 CHAPTER 4 COMPANIES

The list of U.S. equities traded in Chapter 6.

<table>
<thead>
<tr>
<th>TICKER</th>
<th>COMPANY NAME</th>
<th>SECTOR</th>
<th>INDUSTRY</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRAI</td>
<td>CRA InternationInc. (CRAI)</td>
<td>Industrials</td>
<td>Consulting Services</td>
</tr>
<tr>
<td>TXN</td>
<td>Texas Instruments Incorporated (TXN)</td>
<td>Technology</td>
<td>Semiconductors</td>
</tr>
<tr>
<td>WHR</td>
<td>Whirlpool Corporation (WHR)</td>
<td>Consumer Cyclical</td>
<td>Furnishings, Fixtures &amp; Appliances</td>
</tr>
<tr>
<td>ACIW</td>
<td>ACI Worldwide Inc. (ACIW)</td>
<td>Technology</td>
<td>Software, Infrastructure</td>
</tr>
<tr>
<td>AMSC</td>
<td>American Superconductor Corporation (AMSC)</td>
<td>Industrials</td>
<td>Specialty Industrial Machinery</td>
</tr>
<tr>
<td>BIG</td>
<td>Big Lots Inc. (BIG)</td>
<td>Consumer Defensive</td>
<td>Discount Stores</td>
</tr>
<tr>
<td>Ticker</td>
<td>Company Name</td>
<td>Industry</td>
<td>Sub-Industry</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------</td>
<td>------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>RTX</td>
<td>Raytheon Technologies Corporation (RTX)</td>
<td>Industrials</td>
<td>Aerospace &amp; Defense</td>
</tr>
<tr>
<td>PKI</td>
<td>PerkinElmer Inc. (PKI)</td>
<td>Healthcare</td>
<td>Diagnostics &amp; Research</td>
</tr>
<tr>
<td>AEIS</td>
<td>Advanced Energy Industries Inc. (AEIS)</td>
<td>Industrials</td>
<td>Electrical Equipment &amp; Parts</td>
</tr>
<tr>
<td>CAKE</td>
<td>The Cheesecake Factory Incorporated (CAKE)</td>
<td>Consumer Cyclical</td>
<td>Restaurants</td>
</tr>
<tr>
<td>ALB</td>
<td>Albemarle Corporation (ALB)</td>
<td>Basic Materials</td>
<td>Specialty Chemicals</td>
</tr>
<tr>
<td>KFRC</td>
<td>Kforce Inc. (KFRC)</td>
<td>Industrials</td>
<td>Staffing &amp; Employment Services</td>
</tr>
<tr>
<td>CTAS</td>
<td>Cintas Corporation (CTAS)</td>
<td>Industrials</td>
<td>Specialty Business Services</td>
</tr>
<tr>
<td>CRVL</td>
<td>CorVel Corp. (CRVL)</td>
<td>Financial Services</td>
<td>Insurance Brokers</td>
</tr>
<tr>
<td>GT</td>
<td>The Goodyear Tire &amp; Rubber Company (GT)</td>
<td>Consumer Cyclical</td>
<td>Auto Parts</td>
</tr>
<tr>
<td>CR</td>
<td>Crane Co. (CR)</td>
<td>Industrials</td>
<td>Specialty Industrial Machinery</td>
</tr>
<tr>
<td>FRT</td>
<td>Federal Realty Investment Trust (FRT)</td>
<td>Real Estate</td>
<td>REIT, Retail</td>
</tr>
<tr>
<td>FICO</td>
<td>Fair Isaac Corporation (FICO)</td>
<td>Technology</td>
<td>Software, Application</td>
</tr>
<tr>
<td>GBCI</td>
<td>Glacier Bancorp Inc. (GBCI)</td>
<td>Financial Services</td>
<td>Banks, Regional</td>
</tr>
<tr>
<td>HNI</td>
<td>HNI Corporation (HNI)</td>
<td>Industrials</td>
<td>Business Equipment &amp; Supplies</td>
</tr>
<tr>
<td>HIBB</td>
<td>Hibbett Inc. (HIBB)</td>
<td>Consumer Cyclical</td>
<td>Apparel Retail</td>
</tr>
<tr>
<td>IEX</td>
<td>IDEX Corporation (IEX)</td>
<td>Industrials</td>
<td>Specialty Industrial Machinery</td>
</tr>
<tr>
<td>IIVI</td>
<td>II-VI Incorporated (IIVI)</td>
<td>Technology</td>
<td>Scientific &amp; Technical Instruments</td>
</tr>
<tr>
<td>JBHT</td>
<td>J.B. Hunt Transport Services Inc. (JBHT)</td>
<td>Industrials</td>
<td>Integrated Freight &amp; Logistics</td>
</tr>
<tr>
<td>KIM</td>
<td>Kimco Realty Corporation (KIM)</td>
<td>Real Estate</td>
<td>REIT, Retail</td>
</tr>
<tr>
<td>LGND</td>
<td>Ligand Pharmaceuticals Incorporated (LGND)</td>
<td>Healthcare</td>
<td>Biotechnology</td>
</tr>
<tr>
<td>Ticker</td>
<td>Company Name</td>
<td>Sector</td>
<td>Industry</td>
</tr>
<tr>
<td>--------</td>
<td>--------------</td>
<td>--------</td>
<td>----------</td>
</tr>
<tr>
<td>MCD</td>
<td>McDonald’s Corporation (MCD)</td>
<td>Consumer Cyclical</td>
<td>Restaurants</td>
</tr>
<tr>
<td>NWBI</td>
<td>Northwest Bancshares Inc. (NWBI)</td>
<td>Financial Services</td>
<td>Banks, Regional</td>
</tr>
<tr>
<td>MDLZ</td>
<td>Mondelez International Inc. (MDLZ)</td>
<td>Consumer Defensive</td>
<td>Confectioners</td>
</tr>
<tr>
<td>MINI</td>
<td>Mobile Mini Inc. (MINI)</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>MOS</td>
<td>Mosaic Company (The) (MOS)</td>
<td>Basic Materials</td>
<td>Agricultural Inputs</td>
</tr>
<tr>
<td>LH</td>
<td>Laboratory Corporation of America Holdings (LH)</td>
<td>Healthcare</td>
<td>Diagnostics &amp; Research</td>
</tr>
<tr>
<td>OGE</td>
<td>OGE Energy Corp (OGE)</td>
<td>Utilities</td>
<td>Utilities, Regulated Electric</td>
</tr>
<tr>
<td>PLCE</td>
<td>Children’s Place Inc. (The) (PLCE)</td>
<td>Consumer Cyclical</td>
<td>Apparel Retail</td>
</tr>
<tr>
<td>ROP</td>
<td>Roper Technologies Inc. (ROP)</td>
<td>Technology</td>
<td>Software, Application</td>
</tr>
<tr>
<td>SJM</td>
<td>J.M. Smucker Company (The) (SJM)</td>
<td>Consumer Defensive</td>
<td>Packaged Foods</td>
</tr>
<tr>
<td>SKX</td>
<td>Skechers U.S.A. Inc. (SKX)</td>
<td>Consumer Cyclical</td>
<td>Footwear &amp; Accessories</td>
</tr>
<tr>
<td>TG</td>
<td>Tredegar Corporation (TG)</td>
<td>Industrials</td>
<td>Metal Fabrication</td>
</tr>
<tr>
<td>WRE</td>
<td>Washington Real Estate Investment Trust (WRE)</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>AAP</td>
<td>Advance Auto Parts Inc W/I (AAP)</td>
<td>Consumer Cyclical</td>
<td>Specialty Retail</td>
</tr>
</tbody>
</table>
The list of U.S. equities traded in Chapter 5 and Chapter 6.

<table>
<thead>
<tr>
<th>TICKER</th>
<th>COMPANY NAME</th>
<th>SECTOR</th>
<th>INDUSTRY</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAP</td>
<td>Advance Auto Parts Inc W/I (AAP)</td>
<td>Consumer Cyclical</td>
<td>Specialty Retail</td>
</tr>
<tr>
<td>ACN</td>
<td>Accenture plc Class A (Ireland) (ACN)</td>
<td>Technology</td>
<td>Information Technology Services</td>
</tr>
<tr>
<td>ADTN</td>
<td>ADTRAN Inc. (ADTN)</td>
<td>Technology</td>
<td>Communication Equipment</td>
</tr>
<tr>
<td>AEIS</td>
<td>Advanced Energy Industries Inc. (AEIS)</td>
<td>Industrials</td>
<td>Electrical Equipment &amp; Parts</td>
</tr>
<tr>
<td>AMAT</td>
<td>Applied Materials Inc. (AMAT)</td>
<td>Technology</td>
<td>Semiconductor Equipment &amp; Materials</td>
</tr>
<tr>
<td>BIG</td>
<td>Big Lots Inc. (BIG)</td>
<td>Consumer Defensive</td>
<td>Discount Stores</td>
</tr>
<tr>
<td>CACI</td>
<td>CACI International Inc. Class A (CACI)</td>
<td>Technology</td>
<td>Information Technology Services</td>
</tr>
<tr>
<td>CAKE</td>
<td>The Cheesecake Factory Incorporated (CAKE)</td>
<td>Consumer Cyclical</td>
<td>Restaurants</td>
</tr>
<tr>
<td>CR</td>
<td>Crane Co. (CR)</td>
<td>Industrials</td>
<td>Specialty Industrial Machinery</td>
</tr>
<tr>
<td>CRAI</td>
<td>CRA International Inc. (CRAI)</td>
<td>Industrials</td>
<td>Consulting Services</td>
</tr>
<tr>
<td>CRVL</td>
<td>CorVel Corp. (CRVL)</td>
<td>Financial Services</td>
<td>Insurance Brokers</td>
</tr>
<tr>
<td>CTAS</td>
<td>Cintas Corporation (CTAS)</td>
<td>Industrials</td>
<td>Specialty Business Services</td>
</tr>
<tr>
<td>CTB</td>
<td>Cooper Tire &amp; Rubber Company (CTB)</td>
<td>Consumer Cyclical</td>
<td>Auto Parts</td>
</tr>
<tr>
<td>EBAY</td>
<td>eBay Inc. (EBAY)</td>
<td>Consumer Cyclical</td>
<td>Internet Retail</td>
</tr>
<tr>
<td>FICO</td>
<td>Fair Isaac Corporation (FICO)</td>
<td>Technology</td>
<td>Software—Application</td>
</tr>
<tr>
<td>FRT</td>
<td>Federal Realty Investment Trust (FRT)</td>
<td>Real Estate</td>
<td>REIT—Retail</td>
</tr>
<tr>
<td>Company</td>
<td>Description</td>
<td>Industry</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>GBCI</td>
<td>Glacier Bancorp Inc. (GBCI)</td>
<td>Financial Services</td>
<td></td>
</tr>
<tr>
<td>HIBB</td>
<td>Hibbett Inc. (HIBB)</td>
<td>Consumer Cyclical</td>
<td></td>
</tr>
<tr>
<td>HNI</td>
<td>HNI Corporation (HNI)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>IEX</td>
<td>IDEX Corporation (IEX)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>IIVI</td>
<td>II-VI Incorporated (IIVI)</td>
<td>Technology</td>
<td></td>
</tr>
<tr>
<td>JBHT</td>
<td>J.B. Hunt Transport Services Inc. (JBHT)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>KBH</td>
<td>KB Home (KBH)</td>
<td>Consumer Cyclical</td>
<td></td>
</tr>
<tr>
<td>KFRC</td>
<td>Kforce Inc. (KFRC)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>KIM</td>
<td>Kimco Realty Corporation (KIM)</td>
<td>Real Estate</td>
<td></td>
</tr>
<tr>
<td>LH</td>
<td>Laboratory Corporation of America Holdings (LH)</td>
<td>Healthcare</td>
<td></td>
</tr>
<tr>
<td>MCD</td>
<td>McDonald’s Corporation (MCD)</td>
<td>Consumer Cyclical</td>
<td></td>
</tr>
<tr>
<td>MDLZ</td>
<td>Mondelez International Inc. (MDLZ)</td>
<td>Consumer Defensive</td>
<td></td>
</tr>
<tr>
<td>MINI</td>
<td>Mobile Mini Inc. (MINI)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>MOS</td>
<td>Mosaic Company (The) (MOS)</td>
<td>Basic Materials</td>
<td></td>
</tr>
<tr>
<td>NWBI</td>
<td>Northwest Bancshares Inc. (NWBI)</td>
<td>Financial Services</td>
<td></td>
</tr>
<tr>
<td>PKI</td>
<td>PerkinElmer Inc. (PKI)</td>
<td>Healthcare</td>
<td></td>
</tr>
<tr>
<td>PLCE</td>
<td>Children’s Place Inc. (The) (PLCE)</td>
<td>Consumer Cyclical</td>
<td></td>
</tr>
<tr>
<td>ROP</td>
<td>Roper Technologies Inc. (ROP)</td>
<td>Technology</td>
<td></td>
</tr>
<tr>
<td>RTN</td>
<td>Raytheon Company (RTN)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>SJM</td>
<td>J.M. Smucker Company (The) (SJM)</td>
<td>Consumer Defensive</td>
<td></td>
</tr>
<tr>
<td>TG</td>
<td>Tredegar Corporation (TG)</td>
<td>Industrials</td>
<td></td>
</tr>
<tr>
<td>TXN</td>
<td>Texas Instruments Incorporated (TXN)</td>
<td>Technology</td>
<td>Semiconductors</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------</td>
<td>------------</td>
<td>----------------</td>
</tr>
<tr>
<td>WHR</td>
<td>Whirlpool Corporation (WHR)</td>
<td>Consumer Cyclical</td>
<td>Furnishings, Fixtures and Appliances</td>
</tr>
<tr>
<td>WRE</td>
<td>Washington Real Estate Investment Trust (WRE)</td>
<td>Real Estate</td>
<td>REIT-office</td>
</tr>
</tbody>
</table>
Bibliography


