CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Machine learning for financial applications: self-organising maps, hierarchical clustering and dynamic time-warping for portfolio constructive
Authors	Emerson, Sophie
Publication date	2019-12-15
Original Citation	Emerson, S. 2019. Machine learning for financial applications: self-organising maps, hierarchical clustering and dynamic time-warping for portfolio constructive. MRes Thesis, University College Cork.
Type of publication	Masters thesis (Research)
Rights	© 2019, Sophie Emerson. - https://creativecommons.org/licenses/by-nc-nd/4.0/
Download date	2026-11-02 21:14:23
Item downloaded from	https://hdl.handle.net/10468/10908



UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

MACHNINE LEARNING FOR FINANCIAL APPLICATIONS

SELF-ORGANISING MAPS, HIERARCHICAL CLUSTERING AND DYNAMIC TIME-WARPING FOR PORTFOLIO CONSTRUCTIVE

Sophie Emerson BSc

Supervised by

Dr John O'Brien, Prof Mark Hutchinson,

Head of School

Prof Joe Feller

A Thesis in Quantitative Finance

for the Degree of Master of Commerce in Fintech

University College Cork

December 2019

Abstract

This study investigates how modern machine learning (ML) techniques can be used to advance the field of quantitative investing. A broad literature review evaluated the common applications for ML in finance, and what ML algorithms are being used. The results show ML is commonly applied to the areas of Return Forecasting, Portfolio Construction, Ethics, Fraud Detection Decision Making Language Processing and Sentiment Analysis. Neural Network technology and support vector machine are identified as popular ML algorithms. A second review was carried out, focusing in the area of ML for quantitative finance in recent years finds three primary areas; Return forecasting, Portfolio construction and Risk management.

A practical ML experiment carried out as a proof of concept of ML for financial applications. This experiment was informed by the results of the broad and more focused literature searches. Two forms of ML techniques are used to analyse market return data and equity flow data (provided by State Street Global Markets) and create a portfolio from insights derived from the ML technology. The ML technologies employed are those of Self-Organising Maps and Hierarchical Clustering. The portfolios created were tested in terms of risk, profitability and stability. Stable regimes and profitable portfolios are created. Results show that portfolios obtained by analysing equity flow data consistently outperform those created by analysing return data.

Acknowledgements

I would first like to thank my supervisor Dr John O'Brien of the school of accounting and finance] at UCC. He was always available to offer focused and constructive advice; this project is of a higher quality due to his continued support throughout the year. I appreciate the many hours spent explaining difficult concepts and reading through my work, his advice has always improved the substance and integrity of whatever body of work I was working on. Many people offered their help, support and oversight throughout this year. Prof Philip O'Reilly always had an open door and offered valuable insights and assisted in many aspects of this programme. Prof Mark Hutchinson oversaw the project, provided excellent direction and helped shape what this project was to become. John McAvoy guided me through the process of the Msc and helped me keep on top of deadlines. He was always on call to provide clarity on instruction into all facets of carrying out a research project.

State Street kindly sponsored this research endeavour. In addition to providing monetary support, they also stayed in close contact throughout the year. Brian McCabe and Neasa Ní Fhátharta made time to attend and organise regular meetings to discuss this research project. Each session offering excellent suggestions and asking questions that greatly strengthened my understanding of the core concepts of this project and improving the overall quality of my work. This research would not have been possible without the instruction and collaboration with State Street Global Markets. Michael Metcalfe and Maria Veitmane provided equity flow data and information on past studies of their own which inspired and informed method utilised during this study. They generously gave their time to attend regular meetings and offered expert advice on how to improve the study.

Finally, I would like to thank my parents for their continued support of my education and ambitions. They are amazing role models, have taught me uncountable skills and given me a strong work ethic. Without their support, this year of study would not have been possible.

Table of Contents

Abstract 2
Acknowledgements 3
Table of Contents 5
Introduction 9
1.1 Motivations, Objectives and Overview 10
1.2 Structural Organisation 12
1.3 Group Literature Review Paper 12
1.4 Machine Learning 13
1.5 ML for Quantitative Investing 16
1.6 Market Regimes 17
Literature Review 18
2.1 Introduction 19
2.2 Literature Review Methodology 20
2.2.1 Broad Review of ML for Finance 20
2.2.2 Quantitative Finance Literature Review 21
2.3 Common ML Applications 22
2.4 ML Algorithms for Financial Applications 23
2.5 Recent Quantitative Investing Literature 24
2.5.1 Overview of Quantitative Investment 24
2.5.2 Quantitative Investment Strategies 25
2.5.3 Portfolio Diversification Theory 26
2.5.4 Key Insights 27
2.6 Literature Review Insights 36
2.6.1 Strategy Development & Analysis 36
2.6.2 Alternative Data 37
2.6.3 Linking ML Algorithms to Applications 37

2.6.4 Backtesting & Strategy Verification 38
2.7 Conclusions 39

Theory and Background Information 41
3.1 Market Regimes 42

3.1.1 Market Regime Literature 42
3.1.2 Original Research by SSGM 43
3.2 Machine Learning Overview 44

3.2.1 Supervised Learning 45
3.2.2 Unsupervised Learning 46
3.2.3 Semi-Supervised Learning 46
3.2.4 Reinforcement Learning 47
3.2.5 Competitive Learning 47
3.3 Machine Learning Algorithms 47

3.4 Self-Organising Maps 51

3.4.1 Introduction to Self-Organising Maps 51
3.4.2 SOMs for Data Visualisation 52
3.4.3 How SOMs Work 52
3.5 Distance Measures 54

3.5.1 Dynamic Time Warping 55

3.6 Time-Series Clustering 58

3.6.1 Clustering Algorithms 59
Chapter 4 62
ML Experiment 62
4.1 Data 63

4.1.1 Equity Flow Data 64
4.1.2 MSCI 65
4.1.3 Risk-Free Rate 65
4.1.4 Regions 66
4.5 Exploratory Analysis by SOM 67

4.6 Determining Market Regimes 68

4.7 Development of Model Portfolios 68

4.7.1 Portfolio Models 69
4.7.2 Benchmark Portfolio Models 70
4.8 Implementation of Portfolio Models 71

4.9 Portfolio Model Analysis 71

4.9.1 Total Return 71
4.9.2 Annualised Cumulative Return 72
4.9.3 Volatility 72
4.9.4 Sharpe Ratio 73
4.10 Investigating Regime Stability 73

Results and Discussion 75
5.1 Analysis of Flow Data by SOM 76

5.2 Results and Discussion 77

5.2.1 2012-2018 'Equity Flow Regime' Analysis 79
5.2.2 2012-2018 'Return Regime' Analysis 80
5.2.3 1998 – 2018 'Equity Flow Regime' Analysis 81
5.2.4 1998 - 2018 'Return Regime' Analysis 83
5.3 Regime Stability Analysis 84

5.4 Model Performance 87

5.4.1 Model Portfolio Comparison 88
5.4.2 'Equity Flow' Regime Models 2012-2018 90
5.4.3 'Return' Regime Models 2012-2018 91
5.4.4 'Equity Flow' Regime Models 1998-2018 93
Chapter 6 96
Conclusions 96
6.1 Determining Number Regimes by SOM 97

6.2 Stability Analysis of Regimes 98

6.3 Portfolio Model Performance 98

6.4 ML as a Quantitative Investing Tool 99

6.4 Closing Statement 100

References 101
Appendix I Published Literature Review 110
II. Risk 119

Strategy Development & Analysis 120

The use of Alternative Data 120

Choosing Machine Learning Algorithms 121

Backtesting & Strategy Verification 121

CONCLUSION 121
REFERENCES 122
Appendix II - Quant Award Essay Competition submission 125
Appendix III – Code Extracts 135

Chapter 1

Introduction

This chapter provides a brief overview of the primary themes and topics that this research investigates. Machine learning (ML) is defined and explained alongside the main uses for this technology. We explore how ML can be used in the context of quantitative investing and in the area of regime investing. Background information is presented to give the reader an overall sense of the context and relevance of this study. The main research questions and objectives are stated and an overview of the structure of the entire document is included in section 1.2.

1.1 Motivations, Objectives and Overview

Recent advances in ML are finding commercial applications across many industries, not least the finance industry (Dominigos, 2012). This study focuses on applications in one of the core functions of finance, the investment process. This includes return forecasting, risk modelling and portfolio construction (Abe & Nakayama 2018)(Ahmed, Atiya, Gayar,& Shishiny 2010). The study evaluates the current state of the art through an extensive review of recent literature. Themes and technologies are identified and classified, and the key use cases highlighted. Quantitative investing, traditionally a leading field in adopting new techniques is found to be the most common source of use cases in the emerging literature.

The initial objective of this study is to broadly investigate how machine learning techniques are being used in financial applications. An extensive literature review looks at the area of machine learning for financial applications. The results of this systematic literature search are presented and discussed. As the study evolved, a focus was given to the literature in the area of quantitative investing. Many examples of how ML is impacting the field of quantitative investing and found in the literature, highlights of these studies are discussed in the literature review section.

This research was kindly sponsored by State Street and was carried out in the State Street advanced research centre in UCC over a period of twelve months. Support was provided by representatives of the Global Markets sector of State Street. These industry contacts shared information how their team has used ML technology to tackle real-world problems. An interesting example is the use of Self-Organising to create an investor regime map to assist in portfolio management. A regime map describes several different states the market can be in, each state characterised by investor behaviour. Knowledge of these regimes can

assist in making portfolio management decisions. The practical part of this research consisted of carrying out an experiment of using Self – Organising Maps (SOMs) in a similar fashion to that of state street global markets using additional ML technology; hierarchical clustering and dynamic time warping.

An experiment was devised by taking inspiration from the original approach of using SOMs by State Street Global markets.; SOMs to determine an appropriate number of regimes, while hierarchical clustering and dynamic-time warping were used to cluster weekly equity flow data to create four distinct investor regimes. Regime investing is based on the principal that equity flows are stable (Garvey & Chen, 2004). If the market is in a certain regime one week then it most likely will be in the same regime in the following week. Portfolio management decisions can be informed by considering what regime the market is currently in. A portfolio model is created to test this hypothesis and investigate this approach to using ML techniques to assist in quantitative investing. A full description of the methodology of this experiment, the theory behind the technology and key terms and the results are fully discussed in later sections.

This study provides a proof of concept that ML techniques and algorithms can assist in the creation of a profitable portfolio. It shows the advantage of using regional equity flow data to inform the creation of investor regimes in contrast to solely using regional return data. An objective of this study provides a proof of concept that ML techniques can be utilised in the area of quantitative investing. In discovering helpful insights from financial data to inform investment decisions and in the creation and diversification of a portfolio.

1.2 Structural Organisation

The following sections (1.3-1.5) aim to introduce the reader to three fundamental concepts to this body of work; Machine learning, Quantitative Investing and Investor regimes. Chapter 2 describes the literature review process and been split into two sections. First the broad literature review of ML for financial applications and secondly the more focused literature review of recent studies involving ML for Quantitative investing. (Chapter 2 includes methodology, results, discussion and conclusion of the literature search.) Chapter 3 seeks to explain in detail the main concepts which this study deals with including; Investor regimes, original ML project carried out in SSGM, the main types of ML, commonly used ML clustering algorithms, distance measures, time-series clustering, self-organising maps and all related formulae associated with this project.

Chapter four provides relevant information on MSCI data, equity flow and LIBOR data sets which were using for this project. Chapter 5 presents and discusses the results of this experiment. Finally, chapter 6 highlights the main outcomes, conclusions and areas of further research.

1.3 Contribution

I am individually responsible for chapters 1 and 3-7, chapter 2 is collaborative. Chapter 2 consists of a comprehensive literature review which looks at how ML can be used to assist in financial applications. The results of this review show areas where ML is commonly applied to in finance and what ML algorithms and technologies are employed. Chapter 2 is adapted from a paper published in The International Journal of Trade, Economics and Finance. The work was presented in Lyon at the International Conference of Trade and Financial Research (ICEFR). The published paper is included in Appendix III in its original unaltered form.

The work is co-authored by myself and my two peers; Ruairi Kennedy and Luke O'Shea. I will briefly explain here how the work was delegated between the three of us. Once key word and search criteria was decided upon, we all contributed to looking for suitable papers to include in our review. Once enough was obtained, we split the papers up evenly. The financial applications present in each one was noted, and the ML algorithms employed were recorded also. We combined the three sets of results into one concept-centrix matrix and created relevant tables which can be found in chapter 2.

The same approach was taken to the second literature search that focused into the area of ML for Quantitative investing. Three areas of interest emerged from this review; return forecasting, risk management and portfolio construction. We each took one area to explore in more depth and write up discussion sections for the paper. My area was portfolio construction. I took the lead on writing the methodology while my colleagues carried out the task of writing the introduction and conclusion sections of the paper. All authors had equal roles the creation of the results section. Individuals carried out the primary work on each section and would then and it over to another teammate to add elements, proofread and improve with further iterations.

1.4 Machine Learning

ML refers to the field of study involving machines and computer programmes capable of performing useful tasks or gaining insight from data without being explicitly programmed to do so (Domingos, 2012). In recent years there has been a proliferation of ML techniques and growing interest in their applications in finance, where they have been applied to sentiment analysis of news, pattern recognition, trend analysis, risk management among a huge host of other applications (Russell and Norvig, 2009). An example of a well-known learning machine is that of IBM's Watson (Ferucci, 2010). Originally Watson

was a “question and answering machine” which came into the public’s eye when it appeared on the television show ‘Jeopardy!’, where it defeated two of the show’s most successful contestants (Reynolds, 2019).

The first widespread commercial use cases of artificial intelligence were “expert systems”, originating in Stanford in the 1960s (Domingos 2012) and popularized in the 1980s and 1990s. Expert systems were designed to solve complex problems in a specific field, in a manner similar to a subject matter expert. Original expert systems were rule-based programmes developed in languages such as LISP and Prolog. In recent years, there has been a significant drop in interest in classic expert systems, as they are superseded by systems incorporating artificial intelligence (Lindsay, Buchanan, Feigenbaum & Lederberg, 1993). AI systems are systems that replicate human thought processes (Wagner, 2017). Many of these systems are advertised today as cognitive computing systems.

‘Learning’ is a somewhat deceiving term as machines of course cannot learn in the same way a human can (Burkov, 2019). A person could observe a video of a simple game being played many times and from this learn over time how to play the game. A machine can be programmed to learn in the same way but would fail if the screen displaying the video was tilted to the side (Burkov, 2019). Machines cannot account and adjust for this change in the environment the way a human being can. So why call it ‘learning’? This catchy name serves a greater purpose; marketing. The term ‘Machine Learning’ was coined by Arthur Samuel 1959 while he worked in IBM. Samuel was an expert in the field of computer games and artificial intelligence (Domingos, 2012). The name ‘Machine Learning’ made this field of research sound new and exciting, designed to stand out from the competition (Domingos, 2012), to assist in the recruitment of the best talent in computer science and statistics. In addition to assisting to recruitment, the catchy name helped in acquiring new clients and strengthen investor confidence. A similar

strategy was carried out by IBM in 2010 with the phrase ‘Cognitive Computing’. This term refers to machines that emulates the way in which a human mind works (Dominigos, 2012). Cognitive computing is in part carried out by implementing ML techniques.

Many factors have led to the emergence of ML as a popular tool (Reynolds, 2017). It can assist in the automation of various task which can cut cost and create value for a company (Reynolds, 2017). The development of easy to use programming languages in which to create ML algorithms such as Python and R has contributed to their wide usage as well as the creation of frameworks such as TensorFlow (Burkov, 2019). Companies are constantly looking for ways to innovate traditional services and gain insight into very large datasets created from the web that may contain information about consumer sentiment, market influences and much more (Burkov, 2019).

ML techniques is revolutionising health services, IT and has a huge amount of financial applications (Wagner, 2017). Machine learning can be applied to predicting the market in many ways (Heaton, Polson & Witte, 2017) (Ritter 2017). The literature search presented as a part of this study highlights some cases where this technology has been successfully applied to real world tasks. A notable example is how machine learning is assisting advancements the field of sentiment analysis (Boiy & Moens, 2009). Sentiment analysis is the process of systematically analysing text to identify feeling or opinion towards a certain topic (Boiy & Moens, 2009). In this age of big data, vast oceans of internet chatter contain valuable information about consumer opinions on products and companies. When properly aggregated and analysed this data can be of great use.ML techniques can help make sense of the ever-growing amount of data available to us in ways not possible by traditional techniques (Sharma & Dey, 2012).

1.5 ML for Quantitative Investing

Quantitative Investing refers to the practice and field of study that involves analysing various types of data in a rigorous and systematic way (Spiegeleer, Madan, Reyners & Schoutens, 2018), using the results of the analysis to inform investment decisions. Graham and Dodd's Security Analysis, published in 1934 following the Wall Street Crash of 1929 is the seminal work on fundamental investing and remains in publication today (Graham & Dodd, 2008). It is one of the first books to distinguish investing from speculation, advocating the use of a systematic framework for analysing securities for stock selection.

An interesting type of ML technology is that of Self-Organising maps (SOMs). They are form of artificial neural network (ANN), and are an example of unsupervised learning (Kohonen, 1997). Technical terms such as these will be discussed in full detail during the literature review and Theory sections. State Street Global Markets (SSGM) utilised SOMs to assist in the area of portfolio management. This original method of using self-organising maps is discussing in greater detail in section 3.1.2. Methods and results of SSGM's work informed aspects of this research project. A primary aim of this study is to provide a proof of concept that ML techniques can be utilised in the area of quantitative investing. In discovering helpful insights from financial data to inform investment decisions and in the creation and diversification of a portfolio.

1.6 Market Regimes

A regime is a period of time characterised by a particular pattern of investor behaviour (Balcilar, Demirer, Hammoudeh, 2013). Market regimes are persistent. Identifying the current regime can assist in making investment or trading decisions (Balcilar, 2013). A regime, once identified, may be characterised by a pattern of price changes across different markets. From an investors point of view, once the current regime is identified, because of persistence, the expected regime for the next period is also known. If the regime has an expected pattern of price changes, this produces a prediction of the price movements in that period, financially useful information. In this study we define regimes in terms of equity investment flows into regional markets based on data from a leading investment bank and characterised the returns associated with each market based on the corresponding market indices. The aim of this study is to identify and characterize regimes and test whether these can be used to generate financially useful information.

Market regimes have applications in the field of investment modelling. They can provide a way to assess risk. Typically, regimes do not change from being low to high risk in a day but rather switch over a longer period of time. This study used the latest AI techniques, specifically self-organising maps, to identify four stable regimes based on equity investment flows into regional markets. This stability of regimes is key to the usefulness of market regimes. Once identified, the average equity returns for each regime were characterised based on the country or regions MSCI Index. Finally, the potential of this regimes to generate useful information was demonstrated by showing three different trading strategies produced a positive return in back-tests.

Section 3.2.1 gives a detailed explanation of this original research project. This project revisits this area, using the latest ML advances, including the latest neural network methods and ML algorithms to attempt to recreate a working portfolio model.

Chapter 2

Literature Review

The following literature review has been adapted from the paper ‘Trends and Applications of Machine learning for Financial Applications’. The work was presented in Lyon at the International Conference of Trade and Financial Research (ICEFR) and published in the associated journal; The International Journal of Trade, Economics and Finance. This paper was co-authored by my peers Ruairi Kennedy and Luke O’Shea. The workload was evenly shared between us. Details of how the work was allocated can be found in the introduction chapter in section 1.3.

Abstract — Recent advances in machine learning are finding commercial applications across many industries, not least the finance industry. This paper focuses on applications in one of the core functions of finance, the investment process. This includes return forecasting, risk modelling and portfolio construction. The study evaluates the current state of the art through an extensive review of recent literature. Themes and technologies are identified and classified, and the key use cases highlighted. Quantitative investing, traditionally a leading field in adopting new techniques is found to be the most common source of use cases in the emerging literature.

Index Terms—Machine Learning, Quantitative Finance, Portfolio Construction, Return Forecasting

2.1 Introduction

The study evaluates the current state of the art through an extensive reviewed of recent literature. Themes and technologies are identified and classified, and the key use cases highlighted. Quantitative investing, traditionally a leading in adopting new techniques is found to be the most common source of use cases in the emerging literature. This function includes return forecasting, risk modelling and portfolio construction. This literature search and review is divided into two distinct sections. The first is a broad review of machine learning in finance. This includes popular use cases for ML, commonly used ML algorithms and which algorithms are most used in specific applications. The aim was to draw connections between popular use cases in finance and current ML techniques. This search yielded information on the popular use cases and technologies.

A second, more focused, literature search was carried out in the area of ML for quantitative finance. We provide an overview of the development of the area as a background for the discussion, this includes the emergence of ML as a useful tool, common algorithms and methodologies, and a review of the evolution and theory of quantitative investing. Only recent papers were included in this search, published and soon to be published papers only. The reasoning behind this narrow time period was to attempt to evaluate where the cutting edge of ML for quantitative finance lies. This literature review has been adapted from a paper written by myself and two members of my research team; Ruairi Kenney and Luke O'Shea. The paper was published in the International Journal of Trade, Economics and Finance and presented at the 2019 international conference on Economics and Finance Research in Lyon. The paper was adapted to better fit this specific thesis. The original paper can be found in appendix I.

2.2 Literature Review Methodology

A full description of the methodology, tools and techniques used in the literature review is presented. Separate approached were given to the initial broad literature search and the second more focused review of ML for quantitative finance.

2.2.1 Broad Review of ML for Finance

Initially, a broad search was conducted to identify the major themes related to ML. This search yielded information on the popular use cases and technologies. This information informed a second, more focused investigation of relevant material. The aim was to draw connections between popular use cases in finance and current ML techniques.

As quality and scope of published research can vary widely, measures were taken to reduce the possibility of including unreliable studies in the final dataset. Before inclusion in the concept matrix, each paper was assessed on quality. This was achieved by using a variety of quality indicators including; the citation count, the quality of an institute's research activities associated with the paper, bias created from funding sources, and the impact factor of the journal. The journal Impact Factor is calculated by taking the average number of times articles from the journal published in the past two years have been cited in the year and dividing that number by the total number of articles published in the two previous years.

An appropriate search strategy was devised and carried out based on the main topics that were identified during the literature review. Search words included: image recognition, sentiment analysis, market prediction, and language processing, used in conjunction with ML. The

purpose of searching by use case was to identify which technology is widely and effectively used to accomplish the tasks in recent years. Every paper was assessed in relation to its relevance and quality.

2.2.2 Quantitative Finance Literature Review

An appropriate search strategy was devised and carried out based on the main topics that were identified during the first investigation of the literature. The arXiv and SSRN databases were searched to ensure that the most up-to-date research papers were included. However, as these are not peer reviewed papers, extra care was taken to ensure that the papers were from reputable authors, focusing on the quality of authors' previous publications. The topic phrases used in search were "portfolio management", "stock market forecasting", and "risk management". All these topic phrases were used in conjunction with the key phrase "machine learning" to return only relevant research papers. The purpose of searching by topic was to identify which technologies are widely and effectively used within each area. As we are evaluating the current state of the art, we wanted to ensure that only recent papers were included. Thus, we only included papers that were submitted in 2015 or later.

From the initial search we collected a total of 118 papers. After an initial review of abstracts, papers that were not relevant to machine learning in finance (specifically investing) were removed. Any papers that were duplicates under more than one search topic were kept under the topic that appeared most relevant. Papers were then assessed in relation to their quality using the quality indicators mentioned above. This reduced the number of papers to 55.

2.3 Common ML Applications

A concept-centric matrix was utilised initially to identify which areas commonly use machine learning techniques. Recurring concepts and themes were noted and counted across a sample of 67 papers identified. An initial list of recurring themes was identified and analysed. Some themes, such as ‘Geopolitics' were removed as they were deemed irrelevant due to the lack of research on the topic. Recurring themes are presented in Table I.

Table 1 - Common themes identified in broad ML for finance literature search

Theme	References
Return Forecasting	21
Portfolio Construction	12
Ethics	8
Fraud Detection	8
Decision Making	8
Language Processing	7
Sentiment analysis	7

As can be seen in table 1; the most common use-cases identified were return forecasting and portfolio construction. Quantitative methods were introduced to finance through the equity market and it is unsurprising that it should lead the way in incorporating the latest advances in its processes. Many of the papers above also discussed risk modelling. This led us to take return forecasting, portfolio construction, and risk modelling as our three core topics.

Many techniques used in the papers only appear once, some twice. Since the purpose of this paper is to identify the most popular machine learning techniques used in finance, specifically in the topics above,

only techniques which appeared in at least three papers were included in the table. We also decided to include RNN, although it is only mentioned explicitly in two papers, it appears implicitly more frequently as both LSTM and GRU are subsets of the technology. The results of the analysis are presented in Table 2.

2.4 ML Algorithms for Financial Applications

The most popular ML techniques identified in the papers researched are presented in Table II overhead, as well as a breakdown of the different acronyms used in the table. This information was collected from a sample set of 67 papers, all that focus in the area of ML for financial applications.

Table 2 - Common ML algorithms identified in broad ML for finance literature search

	MLP	**SVM**	**LSTM**	**GRU**	**RNN**	**CNN**	**RF**	**GPR**	**LR**
Return Forecasting	7	5	4	2	-	1	2	-	-
Portfolio Construction	7	2	3	1	1	1	4	2	1
Risk Modelling	6	2	2	1	1	1	4	3	4

MLP	Multilayer Perceptron
SVM	Support Vector Machine
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
RNN	Recurrent Neural Network (basic)
CNN	Convolutional Neural Network
RF	Random Forests/Decision Trees
GPR	Gaussian Process Regression
LR	Logistic Regression

Artificial neural networks are used in all three areas of finance studied, with a standard feedforward network (MLP) being the most common. Useful results are found from networks that range from small to very large networks (deep neural networks). There is also evidence of preferences for some techniques across different areas of industry and various applications. For example, Gaussian process regression is used in both portfolio construction and risk modelling but has not been applied to return forecasting.

2.5 Recent Quantitative Investing Literature

This section outlines the latter part of the literature review. Here we focus the search to only include recent papers discussing instances where ML has been leveraged in the area of quantitative finance. We begin with a brief overview of quantitative finance and an explanation of some key and recurring terms such as portfolio diversification theory and the leading quantitative investment strategies.

2.5.1 Overview of Quantitative Investment

A quantitative approach to market analysis gained popularity as advances in computing technology made the collection and analysis of large amounts of market data possible. Graham and Dodd's *Security Analysis*, published in 1934 following the Wall Street Crash of 1929 is the seminal work on fundamental investing and remains in publication (Graham & Dodd, 2008). It is one of the first books to distinguish investing from speculation, advocating the use of a systematic framework for analyzing securities for stock selection. This allowed the development and verification of market models on a scale not previously possible, contributing to significant advances in the understanding of financial markets, including the Capital Asset Pricing Model (CAPM), (Sharpe, 1964), (Mossin, 1966), (Lintner, 1975), (French, 2003) and Efficient Market Hypothesis (EMH) (Malkiel & Fama, 1970).

A systematic approach to portfolio construction and risk analysis was presented in Portfolio Selection (Markowitz, 1952), published in 1952. In this, Markowitz provides a mathematical definition of risk as the standard deviation of return. The approach focused on maximizing portfolio performance by optimizing the trade-off between risk and return. This was the foundation of modern portfolio theory, providing an analytical framework for the construction and analysis of investment portfolios (Becker & Reinganum, 2018).

2.5.2 Quantitative Investment Strategies

This project partly seeks to evaluate how useful ML techniques are in the creation and implementation of quantitative investment strategies. It is essential to first understand and consider the established quantitative investing strategies that are a key part of any quantitative investor's toolkit. In this section we cover some basic strategies, namely active equity, value and momentum investing. Active equity investing refers to a portfolio management strategy where the investor continues to actively buy and sell. Therefore, an active investor must frequently monitor the market to exploit profitable opportunities (Fabozzi, 2005). This strategy contrasts passive investing, where equity is purchased for its long-term appreciation value and does not involve constant buying and selling by the investor. Success of this strategy largely depends on the skill of the investor, the ability to notice trends and weaknesses in the market and overtime the overall skill of active investors has increased (Pástor, 2003).

Value and momentum are two popular trading strategies, investors often use one or the other. This paper investigates the returns on investment when used simultaneously to assess the market and find that these strategies offer a more powerful result than either on its own. A three-factor model is devised and tested. These three factors are the global

market index, value and momentum. A market index is a weighted average of investments from a section of the stock market, its calculated from the price of the selected stocks (Fabozzi, 2005).

Value investing is an investment strategy that exploits that market by purchasing securities that appear under-priced by some sort of analysis. Any data that is important for decision-making or has a logical relationship with an equity is quantitative (Fabozzi, 2005). Models are built to predict expected returns and are based upon an equities value relationship with various factors (Fabozzi, 2005). Momentum and sentiment are two widely used factors (Fabozzi, 2005).

2.5.3 Portfolio Diversification Theory

In the ML experiment section of this study we diversify a portfolio across different global regions. This can be considered regional portfolio diversification, but we can diversify a portfolio in different ways, one can also diversify by industry sector. Harry Markowitz outlines how diversification of assets can reduce risk in a portfolio (Markowitz, 1952). This degree to which diversification is effective is dependent on the level of correlation present between security returns. Modern portfolio theory states that diversification of security returns with lower correlation should yield more favourable results for an investor (Levy & Sarnat, 1970). There exists a variety of different approaches that an investor can take to diversify a portfolio, including diversification by sector and by country or region. In a 1970 paper by Levy and Sarnat discusses the high degree of correlation between security returns in a single economy and presents the benefits of diversifying assets internationally in comparison to holding assets across different industries domestically (Levy & Sarnat, 1970). For many years international diversification has been an established portfolio management strategy and has grown more popular in recent decades (Hitt, 2006).

2.5.4 Key Insights

The paper selection included ML papers published in recent years as well as papers yet to be published by established authors from reputable institutions. The most recent studies in this field were included to help evaluate the cutting edge and state of the art of the use of ML for financial applications.

2.5.4.1 Portfolio Construction

Portfolio construction is the process of combining return forecasts and risk models to create an optimum portfolio given an investor's constraints. A variety of ANN methodologies are applied to the portfolio optimisation problem, often outperforming traditional optimisation techniques (Deng & Yu, 2014) (Nakagawa, Uchida & Aoshima, 2018) (Jiang, Xu & Liang, 2017). Deep learning reappeared many times during this search in the context of portfolio construction. Deep learning refers to models that consist of multiple layers or stages of nonlinear information processing (for example, a neural network with many hidden layers) (Deng & Yu, 2014). Both hierarchical clustering and reinforcement learning were used to improve portfolio diversification.

Multiple papers discuss the method of applying Markov models to predict the performance of stocks (Fons, Dawson, Yau, Zeng & Keane, 2019) (Samo & Vervuurt, 2016). Markov models are a type of ML method that model variables that change randomly through time (Guyer, 2009). The complicated nature of the global market makes using this type of model a viable option. Markov models are relevant to the field of regime investing and are thus important to the theory of the ML experiment implanted in the later stages of this project.

2.5.4.2 ML for Portfolio Construction Studies

- The authors present a deep learning framework for portfolio design, applying their framework to the stocks in the IBB index, demonstrating that their portfolio weighted using deep learning outperformed the index (Heaton, Polson & Witte, 2017).

- The author outlines a reinforcement learning solution for a rational risk-averse investor seeking to maximize expected utility of final wealth, giving an example of a Q-learning agent exploiting an approximate arbitrage in a simulation (Ritter, 2017).

- The authors of both papers make use of hierarchical clustering algorithms for constructing diversified portfolios. The portfolios are constructed using variations of risk parity (Lopez & Prado, 2016) and equal risk contribution methods (Raffinot, 2017) which take the hierarchical correlation structure of the assets into account. The portfolios constructed are shown to have superior diversification and out-of-sample risk adjusted performance.

- The authors make use of convex analysis techniques to devise an optimal portfolio coupled with a Hidden Markov Model (HMM) used to estimate growth rates in the market model, which achieves improved results over a simple model using geometric Brownian motions (Al-Aradi & Jaimungal, 2019).

- The authors provide an overview of the financial applications of Gaussian processes and Bayesian optimisation, providing examples for forecasting the yield curve with Gaussian processes, and using Bayesian optimisation to build an online trend-following portfolio optimisation strategy (Gonzalez, Lezmi, Roncalli & Xu, 2019).

- The authors compare the use of Feature Salient Hidden Markov Models (FSHMM) and HMM for constructing factor investing portfolios. The FSHMM selects relevant factors for use from a pool of available factors, while the HMM uses the whole pool of factors. Both models outperformed benchmark portfolios, with the FSHMM portfolio showing better performance (Fons et al 2019).

- The authors use factors as inputs to deep neural network, SVM and random forest models for predicting stock returns. While their research shows the effectiveness of a deep learning model, more significantly they used Layer-wise Relevance Propagation (LRP) to determine individual factor contributions to the neural network's prediction (Nakagawa, Uchida & Aoshima, 2018).

- The authors create a non-linear multi-factor model using LSTM to estimate the non-linear function. As in the previous paper the authors make use of LRP to identify which factors contribute to the model. The performance of the LSTM model is compared to the neural network model used in (Nakagawa, Uchida & Aoshima, 2018) and gives superior returns (Nakagawa, Ito, Abe & Izumi, 2018).

- The authors examine the use of three deep reinforcement learning algorithms, Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO) and Policy Gradient (PG), in managing a portfolio of assets in the Chinese stock market. They propose the use of adversarial training methods and employ a revised PG algorithm which outperforms a Uniform Constant Rebalanced Portfolio (UCRP) benchmark (Liang, Jiang, Chen, Zhu & Li, 2018).

- The authors employ models constructed using Gaussian processes and Monte Carlo Markov Chains which learn optimal strategies from historical data, based on user-specified performance metrics (e.g. excess

return to the market index, Sharpe ratio, etc.). This approach addresses the inverse problem of Stochastic Portfolio Theory – devising suitable investment strategies that meet the desired investment objective, when initially given a user-defined portfolio selection (Samo & Vervuurt, 2016).

- The author provides an ML framework for estimating optimal portfolio weights. They apply this framework using three ML methods – Ridge and Lasso regression, and two newly introduced methods; Principal Component regression, Spike and Slab regression. All methods outperform the mean-variance, minimum-variance, and equal weight portfolios (Kinn, 2018).

- The authors propose a way to find the risk budgeting portfolio by using optimisation algorithms to find a solution to the logarithmic barrier problem. They use algorithms such as cyclical coordinate descent, alternating direction method of multipliers (ADMM), proximal operators, and Dykstra's algorithm (Richard & Roncalli, 2019).

- The authors present a financial-model-free reinforcement learning framework as a solution to the portfolio management problem. The study tests the proposed framework with the following neural networks: CNN, a basic RNN, and LSTM (Jiang, Xu & Liang, 2017).

2.5.4.3 Return Forecasting

Return forecasting refers to the practice of predicting the investment return from an asset or asset class (Fama, 1973). It is central to investment management and features highly in the literature (Fama, 1973) (Lintner, 1975) (Kahn, 2018). Many types of ANN are tested on their ability to forecast returns (Song, Zhou & Han, 2018). Deep neural networks, CNNs, LSTMs are all applied to the problem of return

forecasting (Tsantekidis et al, 2017). In one theme, the new ML technology is applied to improve forecasts made from traditional inputs, such as fundamental accounting data or technical indicators. A second approach uses ML to extract new inputs from alternative data, such as sentiment from news data. Finally, the authors predict movement at market level rather than at the level of individual securities, for example using ML to identify states.

2.5.4.4 ML for Return Forecasting

- The authors use a CNN strategy to analyse and detect price movement patterns in high-frequency limit order book data. Multilayer neural network methods and SVMs were also considered. However, they conclude the CNNs provide better performance for this task (Tsantekidis et al, 2017).

- The authors implement several ML algorithms to predict future price movements using limit order book data. They employ two feature learning methods: Autoencoders, and Bag of Features. They compare three different classifiers: SVM, a Single Hidden Layer Feedforward Neural Network (SLFNN), and an MLP. The results from the MLP are better than the other classifiers (Nousi et al, 2018).

- The authors introduce a novel Temporal Logistic Neural Bag-of-Features approach, that can be used to tackle the challenges that come with data of a high dimensionality, in this case high-frequency limit order book data (Passalis et al 2019).

- The authors train a deep neural network on reported fundamental data from publicly traded companies (revenue, operating income, debt etc.). A value investing factor strategy based on forecasted fundamental data outperforms a traditional value factor investing approach with a

compounded annual return of 17.1% vs 14.4% for a standard factor model (Alberg & Lipton, 2017).

- The authors create a simple buy-hold-sell strategy to predict direction of movement for 43 CME listed commodities and FX futures based on an ANN trained on a multitude of features for each instrument designed to capture co-movements and historical memory in the data (Dixon, Klabjan & Bang, 2017).

- The authors use a random forest model to predict the direction of stock prices based on price information and a number of momentum indicators (Relative Strength Index, Moving Average Convergence Divergence, Stochastic Oscillator, Williams %R, On Balance Volume, and Price Rate of Change). The algorithm is shown to outperform existing algorithms found in the literature (Khaidem, Saha & Dey, 2016).

- The authors provide a sentiment analysis dictionary which they use to predict stock movements in the pharmaceutical market sector. With this model they achieve an accuracy of 70.59% (Shah, Isah & Zulkernine, 2018).

- The authors present a methodology to define, identify, classify and forecast market states. They use a Triangulated Maximally Filtered Graph network to filter information, and simple logistic regression for predicting market states. They compare five models, with a Gaussian Mixture Model as their baseline. All five models outperform the baseline in terms of risk/return significance (Procacci & Aste, 2018).

- The authors compare five ANN models for forecasting stock prices: a standard neural network using back propagation, a Radial Basis Function (RBF), a General Regression Neural Network (GRNN), SVM

Regression (SVMR), and Least Squares SVM Regression (LS-SVMR) (Song, Zhou & Han, 2018).

- The authors use 25 risk factors as inputs to ML stock returns prediction models. Results show that deep neural networks generally outperform shallow neural networks, and the best networks also outperform representative machine learning models (Abe & Nakayama, 2018).

- The author employs ANNs to predict product demand for weather sensitive products in Walmart stores around the time of major weather events (Taghizadeg, 2018).

- The authors implement a Gaussian Naïve Bayes Classifier for prediction based on sentiment analysis of Twitter data. The data used was obtained from Twitter and pertained to the 2014 FIFA world cup (Le, Ferrara & Flammini, 2015).

2.5.4.5 Risk Management

Three different themes are identified under the broad heading of risk. The first attempts to employ ML to improve traditional measures of risk used in the mean variance framework (Wang & Ni, 2019) (Goudenége, Molent & Zanette, 2019). The second theme looks for companies at risk of default or bankruptcy (Hisano, Sornette & Mizuno 2018) (Zhang, Luo & Lu 2018) (Chow, 2018). Techniques such as natural language processing are used to identify words that indicate higher risk. The final theme uses ML to develop hedging strategies (Buehler, Gonon & Teichmann & Wood, 2019).

2.5.4.6 ML for Risk Management Applications

- The authors use k-means clustering to construct risk models by clustering stock returns (Kakushadze & Yu, 2016). They demonstrate that this ML approach outperforms statistical risk models (Kakushadze & Yu, 2017). in quantitative trading applications (Kakushadze & Yu, 2019).

- The authors present a framework for hedging a portfolio of derivatives in the presence of market frictions such as transaction costs, market impact, liquidity constraints or risk limits (Buehler et al, 2019).

- The authors show how Gaussian Process Regression can assist in pricing and hedging a Guaranteed Minimum Withdrawal Benefit (GMWB) Variable Annuity with stochastic volatility and stochastic interest rate (Goudenége, Molent & Zanette, 2019).

- The authors show that machine learning can be as effective as other existing algorithms at solving difficult hedging problems in moderate dimension. They use techniques such as a modified LSTM neural network to calculate their hedging strategies (Fecamp, Mikael & Warin, 2019).

- The authors aim to explore the optimal model for business risk prediction. They attempt to do this using XGBoost, and by simultaneously examining feature selection methods and hyper-parameter optimization in the modelling procedure (Wang & Ni, 2019).

- The authors try to predict daily stock volatility using news and price data. Their model, which utilizes a Bidirectional Long Short-Term Memory (BiLSTM) neural network and stacked LSTM's, outperforms

the well-known Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model in all sectors analysed (financial, health care, etc.) (Sardelich & Manandhar, 2018).

- The authors exploit a heterogeneous information network of 35,657 global firms to improve the predictive performance for firms likely to be added to a blacklist. Blacklists are used to keep track of entities that have unacceptable problems, such as financial or environmental issues. Blacklists help keep portfolios profitable and "green". Their model consists of a simple MLP with thirty hidden units (Hisano, Sornette & Mizuno 2018).

- The authors estimate corporate credibility of Chinese companies using a CNN and natural language processing. They use Latent Dirichlet Allocation to summarise the text of news articles and use a CNN to extract the most important words from each topic (Zhang, Luo & Lu 2018).

- The authors compare different strategies for solving a variation of the multi-armed bandit problem. In their version of the problem, the learner can pull several arms simultaneously, or none at all. This could easily be applied to assist in investment decisions. Out of the strategies compared, Bayes-UCB-4P and TS-4P perform the best (Achab, Clémençon & Garivier, 2018).

- The author compares several ML algorithms: Logistic Regression, K-Dimensional Tree (K-D Tree), SVM, Decision Trees, AdaBoost, ANN, and Gaussian Processes (GP) for forecasting business failures (corporate bankruptcy). The techniques used are: Principal Component Analysis (PCA), Linear Discriminate Analysis (LDA), Isometric Feature Mapping (ISOMAP), and Kernel PCA. On the Korean dataset, all models perform similarly. K-D Tree, SVM, and GP perform best over all of the

dimensionality reduction methods used. On the Polish dataset, the linear regression model performs the best (Chow, 2018).

2.6 Literature Review Insights

The results of the literature search demonstrate that there is a wide range of ML techniques being successfully applied to many areas in the development of quantitative investing strategies, outperforming traditional benchmarks, previously used techniques and algorithms in many cases. Algorithms that assume a linear relationship between data can result in reduced accuracy. Authors argue for the use of more advanced mathematical models and ML techniques such as unsupervised learning that are capable of modelling complex non-linear relationships in financial systems.

2.6.1 Strategy Development & Analysis

Taking factor investing as an example of this, (Harvey & Liu, 2017) and (Harvey, Liu & Zhu, 2016) make use of statistical algorithms to show that many factors discovered over the last number of years can be considered inaccurate or invalid. In the aptly named paper, Taming the Factor Zoo, a double selection LASSO ML method was used to analyse the contribution and usefulness of individual factors amongst the large number available today (Feng, Giglio & Xiu). LASSO (Least Absolute Shrinkage and Selection Operator) is a regression analysis method capable of reducing the dimensionality of a large sample while selecting variables significant to the final result (Belloni, Chernozhukov & Hansen, 2014). In (Abe & Nakayama, 2018) the author uses twenty-five factors as model inputs, comparing the use of shallow and deep neural networks, as well as SVMs and random forests for predicting stock returns, finding the deep neural networks (more layers) superior to the other methods.

2.6.2 Alternative Data

The use of ML for the analysis and application of alternative data for example, sentiment analysis, supply chain data etc. has opened up opportunities for new investment strategies. As seen in Table I, sentiment analysis was identified as a popular use case for ML. (Becker & Reinganum, 2018) provides a thorough overview of the growth of big data and sentiment analysis research over the last 30 years, highlighting the use of techniques such as NLP, SVMs and ANNs for the analysis of news, conference calls, reports, and social media activity. They concluded that to date, sentiment information has provided short-term, easy to exploit insights, but long-term persistent insights are hard to achieve (falling in line with EMH). (Kahn, 2018) acknowledges the effectiveness of big data for the modern fundamental investor. This sentiment is echoed in (Lopez de Prado, 2016) where the author makes reference to the recently emerged term “quantamental" – describing a fundamentally leaning investor who manages their portfolio based on data-driven insights provided by ML algorithms. Examples of ML and alternative data being applied together in the results section mainly fall under return forecasting or risk modelling, where decisions may be made based on good or bad news (Shah et al, 2018), weather (Taghizadeh, 2017), or social media sentiment (Ferrara & Flammini, 2015).

2.6.3 Linking ML Algorithms to Applications

Many factors contribute to the choice of ML algorithms, given the wide range available of these algorithms that are available to researchers and businesses. These factors include accuracy, training time, linearity, number of parameters, the number of features and the structure of the data (Barga, Fontama & Tok, 2015). Model training times can also vary hugely between algorithms, making some algorithms more appealing than others when under time constraints. Many algorithms assume a linear relationship between input and output (linear regression, logistic regression, SVMs). This can result in reduced accuracy when dealing

with non-linear problems. The number of features can be overwhelming for some algorithms. It's important to consider the structure of the data and the specific problem, as some algorithms are better suited for certain problems and data structures (Harrington, 2012).

The results of the literature search demonstrate that there is a wide range of ML techniques being successfully applied to many areas in the development of quantitative investing strategies, outperforming traditional benchmarks, previously used techniques and algorithms in many cases. Algorithms that assume a linear relationship between data can result in reduced accuracy. (Lopez de Prado, 2016) highlights this issue in terms of many of the econometric models employed by finance academics and investment managers. The author argues for the use of more advanced mathematical models and ML techniques such as unsupervised learning that are capable of modelling complex non-linear relationships in financial systems.

2.6.4 Backtesting & Strategy Verification

While ML techniques can provide superior performance, financial data is notorious for having a low signal-to-noise ratio, which can lead to the detection of false patterns and results. Backtesting protocols have been proposed to tackle this (Arnott, Harvey & Markowitz, 2012). ML solutions have also been applied to this problem. In (Lopez de Prado & Lewis, 2018) the authors present an unsupervised learning strategy which makes use of a modified k-means clustering algorithm to extract the number of uncorrelated trials from a series of backtests, which can be used in estimating the probability of false positives and estimating the expected value of the maximum Sharpe ratio. While in (Barga, Fontama & Tok, 2015) the authors use a machine learning strategy for backtesting and the evaluation of automated trading strategies which is trained on a number of performance and risk metrics, demonstrating that this strategy outperforms standard metrics such as Sharpe ratio out-of-

sample. The Sharpe ratio is fully explained, and relevant formulae are presented in later sections as this measure of portfolio performance is utilised to evaluate a model portfolio created later in this research.

The development of new backtesting strategies and protocols is welcome and necessary, especially when taking into account recent “black box” criticisms by leading deep learning researchers regarding a lack of testing and reproducibility in the field of ML. In their acceptance speech after winning the “test-of-time” award at NIPS, the leading AI conference, the authors of (Recht & Rahimi, 2017) compared much of recent ML research to “alchemy”, highlighting a situation where algorithms were being created and trained using trial and error methods, with the researchers unable to explain the fundamental operation. They later published a paper highlighting instances of this (Sculley, Snoeck, Wiltschko & Rahimi, 2018).

2.7 Conclusions

Here we discuss the key outcomes of the literature review. Including the popular ML use cases and ML algorithms used in the current day, and what algorithms are best suited to specific use cases. These research questions are answered using results of the broad literature search. We go on to look at the main outcomes from the second more focused literature search into the use of ML in the field of quantitative finance.

As the previous section discusses, ML offers an opportunity for more complex financial analysis than was previously possible. The literature shows that quantitative investors have embraced new tools and techniques as they have emerged (Kahn, 2018), (Becker & Reinganum, 2018). Varieties of ML methods have been applied to areas of quantitative finance– the most popular methods are MLPs, followed by SVMs, and LSTM. ML has been applied to problems in areas such as

return forecasting, portfolio construction, and risk modelling. These ML methods utilize traditional financial data, as well as making use of new types of alternative data. Big data is providing new datasets that need to be analysed and ML techniques are capable of modelling complex (non-linear) relationships and analysing new data.

(Lopez de Prado, 2016) notes the recent trend of traditional hedge funds hiring an increasing proportion of STEM graduates for portfolio construction positions, as they possess the required mathematical skillset for performing complex analysis and computer modelling. An understanding of machine learning, as well as the languages (Python, R, etc.) and frameworks (e.g. TensorFlow) needed to construct complex models could certainly be considered advantageous for any quantitative investor looking for an edge.

Chapter 3

Theory and Background Information

The purpose of this section is to provide some essential theory and background information that is relevant to the practical experiment portion of this research where a portfolio model is devised using ML technologies. We begin by considering the integral concept of a ‘regime’. A look at how SSGM defined regimes in their 2004 research is explored and an intuitive explanation to this investment tool is provided. In dept details of the technology employed in this study are given in addition to any relevant derivations and equations. But before that we take a step back and give a brief overview as to what machine learning is and the main types of learning that can be achieved. We give some greater detail of widely used ML algorithms including artificial neural networks, which are later applied in the practical experiment portion of this research in the form of Self-Organising Maps. A key step in the process of devising a model portfolio is the clustering of equity flow data. Background and details are given pertaining to the area of time-series clustering in section 3.5.

3.1 Market Regimes

Portfolio managers have many tools and techniques at their disposal. Considering what “Market Regime”, sometimes referred to as “Investor Regimes”, the market resides in can be helpful investment decisions (Hamilton, 2005)(Garvey & Chen, 2004). A “Market Regime” refers to a state the market is in, where equity is moving and what regions are exhibiting high, low or neutral returns. For example, a regime may be a period of time where investors are buying heavily in Asian markets while selling in European ones. Knowing this information allows one to make more informed decisions about portfolio diversification, i.e. what regions to buy and sell assets in. An established principle used by investor is that if the market resides in a certain regime in a given week then it is most likely to be in the same regime in the following week (Garvey & Chen, 2004). We test this hypothesis in the ML experiment section of this study.

A regime is defined as a time period where there is a partticular pattern of equity flow across the different global regions. This term is synonymous with that of ‘investor regime’, a phrase used by State Street Global Markets (SSGM) in their 2004 whitepaper. This research is discussed in detail in section 3.1.2. Understanding the methodology and results of this original body of work gives context to decisions and avenues taken during this research. During the discussion section of this document we briefly compare the results of this study to that of the original research.

3.1.1 Market Regime Literature

The existence of Market Regimes is not a new concept. Sherwood and Hamilton describe regimes as ‘Markov processes’, which is also known as a Markov Chain, in two 1990 papers on the subject (Sherwood,

Hamilton et al, 1990)(Hamilton, 1990). A Markov describes a sequence of possible events, where the probability of an event occurring depends only on what event occurred previously (Geyer, 1992). In the context of Hamilton's study, each event in question refers to the market residing in a certain regime.

(Hamilton, 2005) discusses 'Regime switching models'. The presence of regimes implies the ability of asset returns to change significantly from one time period to another. Hamilton backs up this theory by pointing out that abrupt changes are a prevalent feature of financial data and so abrupt changes will also appear in asset prices (Hamilton, 2005).

3.1.2 Original Research by SSGM

The original research conducted at SSGM involved applying cluster analysis to portfolio flows in order to identify market regimes and estimate how long they will last for (Garvey & Chen, 2004). This study identifies investor regimes and assesses their stability, but not their duration. The cluster analysis techniques used in the original study are SOMs and K-means clustering algorithm. SOMs were also utilised in the analysis portion of this study but as an exploratory analysis technique rather than a clustering one (Garvey & Chen, 2004).

A key assumption that the original study by SSGM shares with this project is that if the market resides in a certain regime one week then it will most likely be in the same regime in the next week. Later in the ML experiment portion of this project, we show this to be true when using equity flow data to determine the regimes. but not true when return data is using to determine regimes. The reasoning for this is that equity flows are generally persistent/stable while return data is not so.

3.2 Machine Learning Overview

An excellent source of information on the basics of Machine learning was found in the resource ‘The hundred-page machine learning book’ published by Andriy Burkov in 2019. This piece of work provided clear and up to date information about the basics of learning, expert systems, ML algorithms and all other relevant aspects to this field of study. Machine learning (ML) is a subfield of Artificial Intelligence (AI) that uses statistical techniques, hardware and software to create computer models that have the ability to learn from a dataset, this allows the models to perform specific tasks without being explicitly programmed to do so (Burkov, 2019). The catchy name made the technology appear to be cutting edge which encouraged research into this field in addition to assisting in acquiring the best talent in hiring and was used to impress new and existing clients.

Although variations of ML have long been around, the discipline has developed rapidly in recent years (Burkov, 2019). Many factors have combined to derive this development. Increased computer power has made real time processing feasible for many complex tasks, increase connectivity has driven innovation and automation in the delivery of traditional tasks and services, the potential to extract useful information from the vast amounts of data generated via the internet (Big Data) has led to novel analytic methods. Alongside this, the development of easy to use programming languages, such as Python and R, and ML focused frameworks such as TensorFlow, has contributed to the wide investigation of ML applications in industry (Burkov, 2019). It has already found commercial application across multiple industries from automated trading systems in the finance industry to the health sector where ML algorithms assist decision making in fertility treatments (Anway, Cupp, Uzumcu, Skinner, 2005). The success of these applications is driving commercial research into further applications.

This section provides explanations of the main types of machine learning; supervised, unsupervised, semi-supervised, reinforcement and competitive learning. They are some of the ways in which ML can be achieved. Supervised learning generates a function that maps inputs to outputs based on a set of training data. The algorithm infers a function linking each set of inputs with the expected, or labelled, output in the training set.

3.2.1 Supervised Learning

In Supervised learning, the dataset used in the algorithm must be labelled data (Burkov, 2019). Each element of the data set can have several different features. For example, a single datapoint might refer to a person and each person might have attributes such as gender, height, weight ect. Another perspective; is that each data object in the dataset is a vector with dimension equal to the number of features.

Each $[(x_i y_i)]_i = N$ is a labelled example where x_i is the feature vector and is it's label. In a supervised system, the label y_i can be one of a finite number of things (Burkov, 2019). A supervised learning algorithm takes a feature vector x as input and outputs information that allows deducing the label for this feature vector (Burkov, 2019).

The main research areas in supervised learning are regression and classification (specifying the category or class to which something belongs), this approach is often used in developing predictive models (Burkov, 2019). Regression techniques predict continuous responses using algorithms such as linear regression, decision trees and Artificial Neural Networks (ANNs). Classification techniques predict discrete responses using algorithms such as logistic regression, Support Vector Machines (SVMs) or K-Nearest Neighbors (KNN).

Supervised learning is employed in the ML experiment portion of this project. Labelled weekly equity flow data is fed in the hierarchical clustering algorithm. These weeks are clustered into groups, these groups are the 'investor regimes'. Characterising these regimes in terms of return data, these insights allowed for the creation of model portfolios.

3.2.2 Unsupervised Learning

Unsupervised learning finds hidden patterns in and draws insights from unlabelled data. In unsupervised learning, the dataset is a collection of unlabelled examples $\{(x_i)\}_i$=N. In general, an unsupervised learning algorithm takes in a vector x as input and either transforms it into another vector or into a value that can be used to solve a practical problem (Burkov, 2019). An intuitive explanation would be that these algorithms are programmed to notice patterns in the data. Unsupervised learning provides inputs to models, but does not specify an expected set of outcomes, the outcomes are unlabelled (Burkov, 2019).

During the ML experiment portion of this study, an unsupervised ML technology known as Self-Organising Maps (SOM) was used to explore the equity flow dataset, provided by contacts in State Street Global Markets (SSGM). Results of this work informed an appropriate number of investor regimes to search for. This technology in explained in detail in section 3.3.

3.2.3 Semi-Supervised Learning

A hybrid system, semi-supervised learning, combines supervised and unsupervised learning, using both labelled and unlabelled data to train models (Burkov, 2019). This is useful where there is limited data, or the process of labelling data could introduce biases. In real life, it may not be practical to label large quantities of data because of the cost associated with getting people to label large quantities of data. Semi-

supervised learning occurs when most of the training data is unlabelled, only a few of the data points are labelled (Burkov, 2019).

3.2.4 Reinforcement Learning

Reinforcement learning lies between supervised learning and unsupervised learning. It operates through continuing interactions between a learning system and the environment (Haykin, 1994). Reinforcement learning enables algorithms to learn by trial and error, based on feedback from past experiences. Like unsupervised learning, it does not require labelled data (Burkov, 2019).

3.2.5 Competitive Learning

An interesting class of unsupervised learning system are those which are based on competitive learning. Competitive learning is where output neurons compete amongst themselves to be activated (Burkov, 2019). Only one neuron can be activated at any one time. The use of competitive learning for practical applications is demonstrated later in this study as it is the type of learning used by the ML algorithm known as Self-Organising Maps. Self-Organising maps are a type of Artificial Neural Network and are used as an exploratory analysis tool during the practical portion of this study (Kohonen, 1990).

3.3 Machine Learning Algorithms

Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task (Burkov, 2019). We previously discussed the main types of learning that a machine can be programmed with, here we look at some common ML

algorithms that are used in research and industry. ML algorithms are essentially a collection of instructions and mathematical equations which compute information about a data set to perform an objective (Burkov, 2019).

3.3.1 Artificial Neural Networks

Artificial neural networks (ANNs) have become a key technology in the development of ML. They were first proposed over 75 years ago, inspired by the workings of the human brain (Haykin, 1994). They are a collection of algorithms that replicate the process of a biological brain at the neuron level. The human mind is incredibly adept at information processing, possessing the ability to recognise patterns, control motion of the body and many other tasks more efficiently than any computer in existence today (Haykin 1994). These algorithms can be utilized to implement both supervised and unsupervised learning.

They are highly useful tools in modelling complex systems across a multitude of different fields. ANNs have been implemented in medicine, cyber security, quantitative finance and many more. These systems maintain certain advantages when creating a programme or device capable of performing tasks that vary with each iteration of use. An example of this would be facial recognition software. Variables such as lighting, skin tone, angle of sight and more can be taken into account by the ML technology. One must devise a programme that can handle these changes and learn from them, ANNs are suited to such a task. They can be adapted to a huge variety of different problems and are effective at modelling non-linear data (Haykin, 1994). Multiple neuron ‘learn’ from input data in parallel, thus they are potentially fast at carrying out many complex computations (Haykin, 1994). As such, ANNs are often implemented in the analysis of big data, a highly relevant field in recent times.

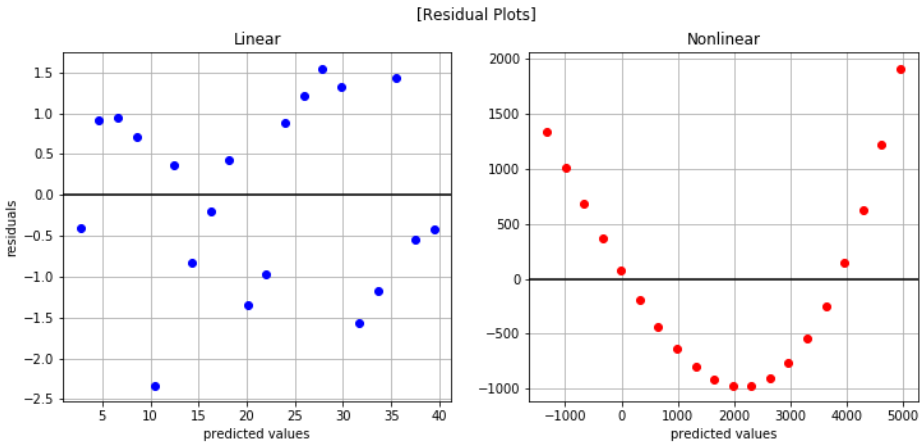


Figure 1 - Linear data (left) and non-linear data (right) displayed in scatter plots

ANN's consist of layers of neurons. There are three different types of layers in a basic neural network. The input layer, the hidden layer and the output layer (Haykin 1994). There are just one input and output whereas there can be many hidden layers.

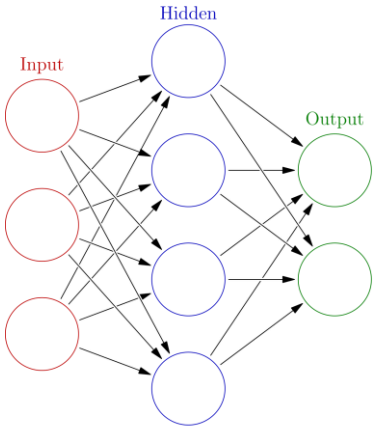


Figure 2 - Depiction of basic neural network setup with a single hidden layer.

Each input neuron represents some independent variable that has an influence over the output of the neural network (Haykin, 1994). The hidden layer is the layer which is responsible for extracting the required features from the input data. Different problems require different numbers of hidden layers depending on the complexity of the problem (Haykin, 2009).

Neural networks are often referred to as Multilayer Perceptrons (MLP). A perceptron is a single layer neural network, ie a simple mathematical algorithm which models the way in which a nerve cell receives signals from other cells (Haykin, 1994). Neural Networks come in a huge variety of different types. There are all based upon the same principle of interconnected neurons as discussed previously. In some cases information can be sent back through the neurons, number of hidden layers may differ, the outputs of an ANN may affect it's inputs and other factors distinguish different types of ANNs (Haykin, 1994).

A Single-Layer Neural Network is one in which the input layer of neurons projects directly onto the output layer of neurons (Burkov, 2019), whereas in a Multi-Layer Neural Network there is at least one hidden layer located between the input and output layer where the information is processed (Burkov, 2019). In a Feedforward Neural Network information can pass from the input layer of neurons along the hidden layer to the output layer but not back in the reverse direction which is possible in a Recurrent Neural Network (RNN) (Burkov, 2019). An artificial neural network algorithm known as Self-Organising Maps is implemented during this study to explore a data set of equity flow data provide by State Street Global markets.

3.4 Self-Organising Maps

The exploratory analysis tool employed to extract the number of investor regimes, is a form of artificial neural network called the Self-Organising Map (SOM). Here we discuss an overview of the technology; how it works, some minor background and highlight some popular applications of SOMs.

3.4.1 Introduction to Self-Organising Maps

A self-organising map is a type of artificial neural network (Kohonen, 1990). They can reduce the dimensionality of data thus making them useful for visualization (Kohonen, 1990). Prof Teuvo Kohonen developed this data analysis technique in the 1980's and was coined the Kohonen map or Kohonen network. An established use for this technology is in the area of exploratory analysis, in examining the structure and finding patterns in large datasets (Kaski, 1997). An effective exploratory analysis tool is essential for analysts working on large and complicated data sets. Analysis of these data sets can sometimes be difficult and time-consuming (Kaski, 1997).

One perspective is that the core purpose of SOMs is to reduce the dimensionality of data (Kohonen, 1990). This technique is a branch of unsupervised learning that can take high dimension data and transform it into a two-dimensional representation of the input data (Kohonen, 1990). High dimension data is when there are many columns of data, ie many variables. Instead of having to deal with hundreds of rows and columns the data is processed into a simplified map; known as a self-organizing map.

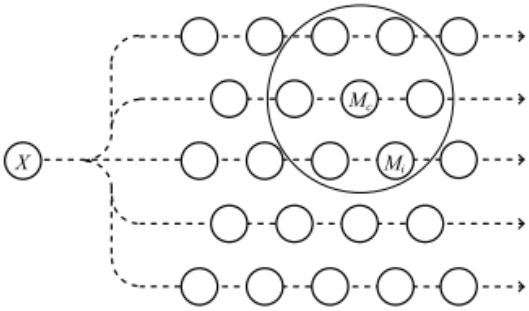


Figure 3 – Nodes of Self-Organising Map

In contrast to multilayer perceptrons (MLP), which are used much more often, the SOMs neurons have a position on a regular grid.

3.4.2 SOMs for Data Visualisation

SOMs can be helpful in the area of data visualisation (Kohonen, 1990). Data science is more than just building machine learning models; it's also about explaining the models and using them to drive data-driven decisions. Displaying data in an informative and visually appealing way can play a very important role of presenting data in a powerful and credible way. They can create two dimensional visualisations of data that has many variable (ie a high dimensionality). This is known as reducing the dimensionality of data as mentioned in section 3.4.1.

SOMs are used in this way during the ML experiment portion of this research. They are used to create visualisations of equity flow data to assist in deciding upon an appropriate number of investor regimes.

3.4.3 How SOMs Work

Like many ANNs, SOMs operate in two modes; training and mapping. The training part of the operation involves building the map using input examples. The mapping part of the operation automatically classifies a

new input vector (Kohonen, 1990). The map space is pre-defined before the training process. The space consists of nodes arranged in a rectangular or hexagonal grid; the dimensions of this grid are pre-set.

Training can in general is considered either batch or online (Haykin, 1994). Online machine learning is used when data becomes available in a sequential order to determine a mapping from data set corresponding labels. The difference between online learning and batch learning (or "offline" learning) techniques (Haykin, 1994), is that in online learning the mapping is updated after the arrival of every new data point in a scale fashion, whereas batch techniques are used when one has access to the entire training data set at once.

The goal of learning in the self-organizing map is to cause different parts of the network to respond similarly to certain input patterns (Kohonen, 1990). All the neurons in the network are originally set to be random values (Kohonen, 1990). The algorithm proceeds iteratively. On each training step a data sample x from the input space is selected. The learning process is competitive, meaning that we determine a winning unit c on the map whose weight vector is closest in magnitude to the input sample x (Burkov, 2019).

This form of training is known as competitive learning. When a training example is fed to the network, its Euclidean distance to all weight vectors is computed. The Euclidean distance between two time series, V and W (Danielsson, 1980);

$$V = v_1, v_{2,} \dots \dots v_n \quad (1)$$

$$W = w_1, w_{2,} \dots \dots w_n \quad (2)$$

by the following formula

$$E(v, w) = \sum_{i=1}^{n} (v_i - w_i)^2 \quad (3)$$

The map space is pre-defined before the training process. The space consists of nodes arranged in a rectangular or hexagonal grid; the dimensions of this grid are pre-set (Kohonen, 1990). SOMs can be helpful in the area of data visualisation. Data science is more than just building machine learning models; it's also about explaining the models and using them to drive data-driven decisions. Displaying data in an informative and visually appealing way can play a very important role of presenting data in a powerful and credible way (Haykin 1994).

3.5 Distance Measures

Once one chooses what clustering algorithm they wish to use for their data, they must also choose an appropriate distance measure. A distance measure defines what is the measure of similarity or dissimilarity between two data points (Bassevile, 1989). An example which highlights the importance of a distance measure is if you wished to write a program which calculated the time taken to get to a destination in a car. The most standard measure of similarity is that of the Euclidean distance (Danielsson, 1980). This measure, in terms of the car example, would calculate the 'bird's eye' view distance between two points on a map. The mathematical expression for the Euclidean distance is given in equation (3).

In real life, this is not a good measure of the distance a car must travel, as it must stay on roads and in some cases roads way have one-way systems. To implement this programme, one must choose a more appropriate distance measure. Shown below is an illustration of how the dynamic time warping distance measure compares two time series, by examining their shapes.

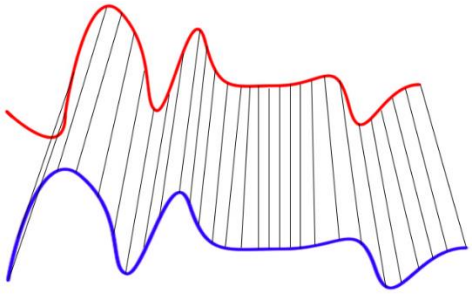


Figure 4 – Dynamic Time Warping comparison of two time-series

Each distance measure has different specifications of what defines a cluster, so a certain clustering distance measures might be preferred depending on what types of clusters one wishes to obtain. Time-Series data can pose some challenges, partly due to factors like large size and dimensionality. A first important issue is to decide whether clustering must be governed by a "shape-based" or "structure-based" dissimilarity concept (Dominigos, 2012). The distance measure chosen for the experiment in this study is that of dynamic time warping.

3.5.1 Dynamic Time Warping

Dynamic Time Warping is a clustering algorithm that considers the shapes of multiple time series and clusters together time series that have similar shapes (Oates, Firoiu & Cohen, 1999). An advantage of dynamic time warping is that having dates out of sync across time series will not affect the results of clusters obtained (Oates & Firoiu, 1999). This can prove particularly useful in international financial time-series data where time zones can cause differences in date/times of close of market. Dynamic time warping simply considers the overall shape of the time series when measuring similarity. This can save time in data cleaning, getting date and times to match up exactly across many time-series can be time consuming and thus costly for companies.

This technique sought to solve the problem of finding patterns in temporal series, i.e. time series. It was developed to fix the issue of time series being out of phase with one another when trying to compare them. An example of an early use case of the Dynamic Time Warping distance measure was by Berndt and Clifford in 1994 (Berndt & Clifford, 1994). Here it was used in the development of speech recognition technology when trying to compare various audio samples that are out of sync with one another.

Dynamic Time Warping is a distance/ similarity measure between two time series (Berndt & Clifford, 1994). It was first proposed by Berndt and Clifford in 1994. To understand how this technology works we first consider a nxn matrix. Each element D_{ij} of the matrix D is the difference between x_i and y_i ie

$$d(x_i, y_i) = | x_i, - y_i | = (x_i - y_i)2 \quad (4)$$

DTW is a 'shape-based' clustering algorithm (Berndt, Clifford 1994). The algorithm clusters together time series that have similar shapes. Consider time series S and T (equations 5 and 6 respectively);

$$S = s_1, s_{2,} \dots \dots s_n \quad (5)$$

$$T = t_1, t_{2,} \dots \dots t_n \quad (6)$$

The DTW measure computes the difference between a point on S to every other point on T. It then iteratively does this to every point on S and creates a matrix out of these values. The warping path is the path taken to get from the lower left matrix entry up to the upper right matrix entry (Berndt, Clifford 1994).

The DTW algorithms clusters together time series that are computed to have the smallest warping path, W, between them. This can be expressed as;

$$W = w_1 w_2 \dots w_k \quad (7)$$

$$w_k = (i, j) \quad (8)$$

Where i is an index from time series S, and j is an index from time series T. A warping path W is a contiguous set of matrix elements which defines a mapping between x and y that satisfies the following conditions (Oates & Firoiu, 1999).:

Boundary conditions: w_1 = (1, 1) and w_k = (m, n) where k is the length of the warping path (The first and last point of the path is predefined).

Continuity: if w_i = (a, b) then $w_i - 1$ = (a_0, b_0) where $a_0 \leq 1$ and $b_0 \leq 1$. (The warping path is smooth).

Monotonicity: if w_i = (a, b) then $w_i - 1$ = (a_0, b_0) where $a_0 - 1 \geq 0$ and $b_0 - 1 \geq 0$ (The warping path is either always increasing or decreasing).

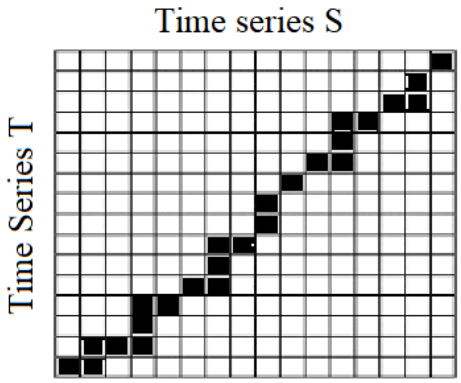


Figure 5 – Distance matrix and Warping path (in black) of time series S and T

An example of a difference matrix is shown in figure 5. The dynamic time warping algorithm aims to find the best warping path linking the bottom left corner of the matrix to the top right corner. Once this path

is identified, a measure of fit for this path is then calculated. This score indicates the similarity between two time series and is in the range [0,1]. This allows the programme to compare similarity between many time series, for example stock prices, and cluster series which exhibit more similar behaviour (Oates & Firoiu, 1999)..

3.6 Time-Series Clustering

A time series is defined as a sequence of values measured at successful time intervals. Clustering of time-series data is mostly utilized for discovery of interesting patterns in time-series datasets. This practice has many real-world applications in the medical and financial industry, as well as many more. Time-series data consists of continuous real-valued data points taken at equal interval in time. Methods of time-series can be split into two categories; frequency-domain methods and time-domain methods (Liao, 2005).

There are two main techniques for time series data clustering; Correlation-based Online Clustering and Shape-based Off-line Clustering. In correlation selection of time series are clustered in real-time based on the correlations among the different time series. This method can be helpful when clustering financial markets. A short window of history is used for the clustering process. These methods often need to be performed in real-time, as the streams are evolving over time (Liao, 2005). When clustering time series one must choose and appropriate algorithm and distance measure.

3.6.1 Clustering Algorithms

Many different algorithms exist that perform clustering (Liao, 2005). Each algorithm has different specifications of what defines a cluster, so a certain clustering algorithm might be preferred depending on what types of clusters one wishes to obtain. Time-Series data can pose some challenges due it's large size and dimensionality. Some important things to consider when clustering time-series data, are the distance measure, the prototype extraction function, the clustering algorithm itself and the cluster evaluation.

"In many cases, algorithms developed for time-series clustering take static clustering algorithms and either modify them in some way to account for time-series. A static clustering algorithm is one that clustered a collection of single data points (Liao, 2005), ie a non-time-series dataset. Aspects of a static time-series clustering algorithm that could be modified include the distance measure/similarity definition or the prototype extraction function. Another path people take is to modify the time-series dataset in some way so that is resembles a static time-series (Liao, 2005).

Very common approaches to time-series clustering are partitioning and hierarchical cluster, these are explained in detail during the following sections. Clustering itself may be shape-based, feature-based or model-based. Some widely used clustering algorithms are discussed in the following sections; 3.52-3.55. The clustering algorithm used for the experiment in this study was that of hierarchical clustering algorithm.

3.6.1.1 K-means Clustering

K-means clustering is a type of unsupervised learning, which is used when you have unlabelled data (i.e., data without defined categories or groups) (Likas, Vlassis & Verbeek, 2003). This algorithm iteratively sorts the data into k groups. Rather than defining groups before looking at the data, clustering allows you to find and analyse the groups that have formed organically (Likas, Vlassis & Verbeek, 2003). This may be helpful to cluster markets without predetermining the groups. K-Means has the advantage that it's relatively fast, as all it's really doing is computing the distances between points and group centres (Likas, Vlassis & Verbeek, 2003).

3.6.1.2 DBSCAN

DBSCAN is a density based clustered algorithm similar to mean-shift, but with a couple of notable advantages (Birant & Kut, 2007). DBSCAN begins with an arbitrary starting data point that has not been visited. The neighborhood of this point is extracted using a distance epsilon ε (Birant & Kut, 2007). If there are a sufficient number of points within this neighbourhood, the clustering process starts, and the current data point becomes the first point in the new cluster. Otherwise, the point will be labelled as noise (Birant & Kut, 2007).

3.6.1.3 Gaussian Mixture Models

A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters (Rasmussen, 2007). One can think of mixture models as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centres of the latent Gaussians. Each Gaussian distribution is assigned to a single cluster (Rasmussen, 2007). Each data point is

assigned to a cluster, the closer the data point is to the centre of the gaussian distribution the higher the probability (Rasmussen, 2007).

3.6.1.4 Hierarchical Clustering

Hierarchical clustering relies using clustering techniques to find a hierarchy of clusters, where this hierarchy resembles a tree structure, called a dendrogram (Johnson, 1967). It allows one to see how different sub-clusters relate to each other, and how far apart data points are. Hierarchical clustering either falls into the top-down or bottom-up category (Johnson, 1967). These similarities do not imply causality (Karypis, 1999). Bottom-up algorithms treat each data point as a single cluster at the outset and then successively merge (or agglomerate) pairs of clusters until all clusters have been merged into a single cluster that contains all data points (Karypis, 1999). Bottom-up hierarchical clustering is therefore called hierarchical agglomerative clustering or HAC. Divisive clustering uses a top-down approach, wherein all data points start in the same cluster (Karypis, 1999). You can then use a parametric clustering algorithm like K-Means to divide the cluster into two clusters. For each cluster, you further divide it down to two clusters until you hit the desired number of clusters (Karypis, 1999).

Chapter 4

ML Experiment

This study looks at how modern clustering techniques can be used to cluster equity flows to define investor regimes. We wish to show by way of experiment that ML can be used to extract useful investment information. The ML techniques employed are hierarchical clustering and dynamic time warping. Intuitive explanations and definitions of these terms will be provided. The ML technology is used to create market regimes and these regimes inform the creation of portfolio models. The same strategy is carried out again but this time clustering the regional return data to define regimes. We examine how profitable the resulting investment model is by comparing it to a risk-free rate. One can compare the use of equity flow data and return data when defining regimes and investigate the validity of using this modern technology to inform international portfolio management.

The first task was to choose an appropriate number of investor regimes to look for. The exploratory analysis tool used was the SOM. Once the optimal number of regimes were determined, the weekly equity flow data was clustered using the hierarchical clustering algorithm and the dynamic time warping distance measure. Four investor regimes were determined and characterised by average weekly returns and stability. A portfolio model was constructed based on the analysis of the four regimes.

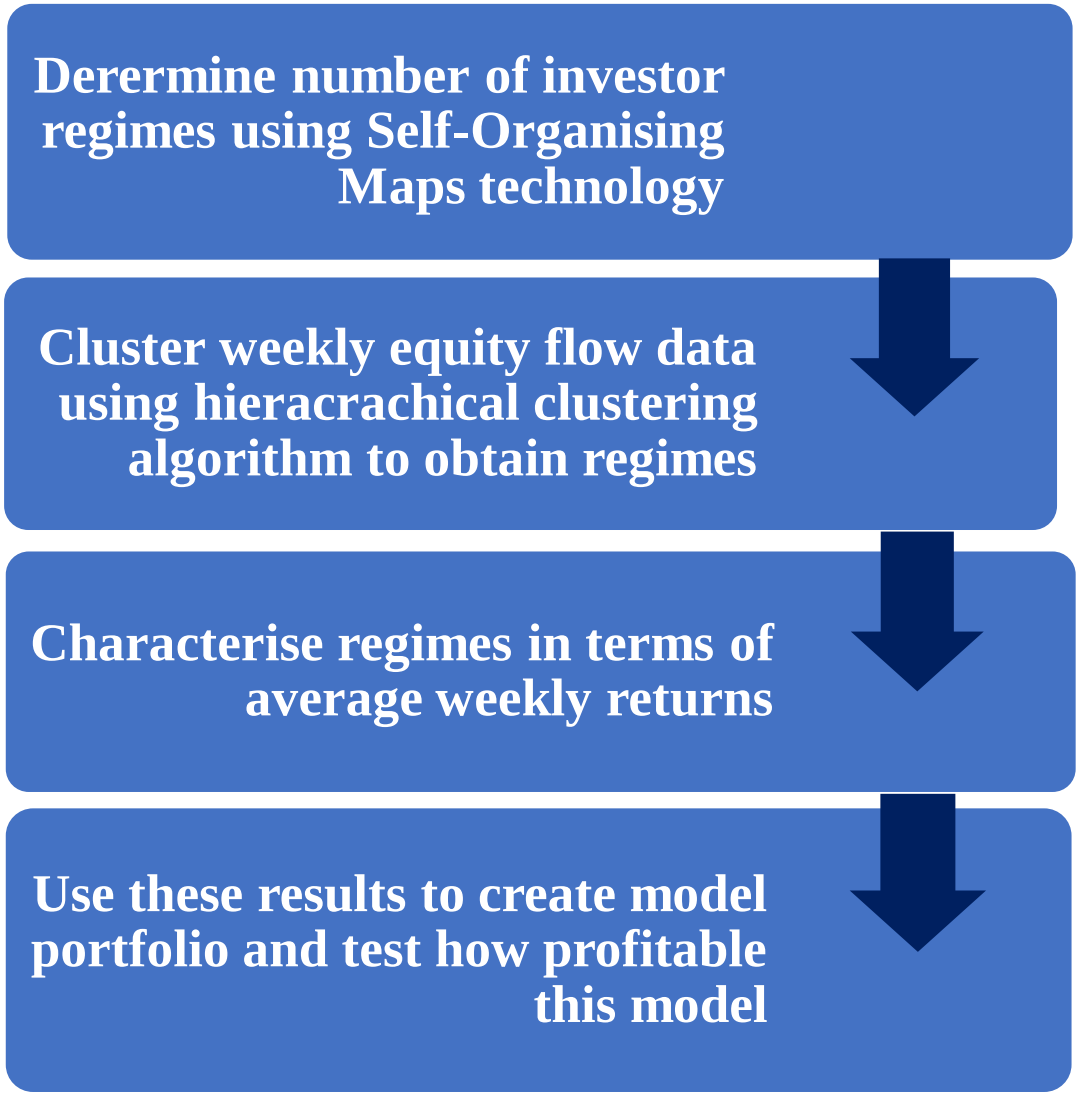


Figure 6 - Overview of Proof of Concept ML experiment steps

4.1 Data

The details of the data sets used during this study are outlined here. The MSCI index was used to obtain the total weekly returns in each region. The second data set used in this study is that of the equity flow data, provided by State Street Global markets. The period selected for analysis was that between 2012/01/07 to 2018/08/18. The data set was originally larger, starting in the year 1998. A full analysis was carried out on the equity flow and return data, where four regimes were determined. Analysis of these regimes showed that two of the regimes

predominantly occurred before 2012 and the other two regimes occurred mostly after 2012. This indicted a structural break in the dataset before and after 2012. It was hypothesized that using data from 2012 onwards may result in creating a model that better fits the current market better and yield a better model.

4.1.1 Equity Flow Data

The primary data set used in this study is that of the equity flow data. This was provided by State Street Global markets. The data are derived from data held by State Street Bank & Trust (SSB). SSB the largest mutual fund custodian in the US and hold roughly 40% of the industry's funds under custody (Froot, 2001). An approximate estimate to the quantity of assets under custody by SSB is $6 trillion. The period selected for analysis was from 7th of January 2012 to 18th of August 2018. The dataset was originally in daily format but was transformed to weekly in the statistical software; R. A breakdown of the regions can be found in section 4.4.

Equity Flow is a measure of how much money is flowing in or out of a country. The flow data provided by State Street consists of two types of data: active flows; and total flows. The data consists of data for 44 countries and 9 regions. The data is represented daily from the 31st March 1998 to the 31st December 2018 (inclusive). To calculate the flows, first, a benchmark flow is calculated empirically. Benchmark flows result from allocating capital by buying or selling at benchmark proportions, i.e. capital is allocated according to capitalization weights across a manager's existing positions. Active flows represent deviations from benchmarks. Total flows are obtained by summing the observed flows and benchmark-implied flows.

4.1.2 MSCI

The MSCI Index is a measurement of investment performance in a particular area, it is the industry's accepted gauge of global stock market activity. The weekly MSCI data was downloaded from Bloomberg.com. The MSCI index was used to obtain the total weekly returns in each region using the following formula.

The formula for calculating returns is as follows:

$$r = \frac{P_t - P_{t-1}}{P_{t-1}} \tag{9}$$

Where P_t is the price at time t.

4.1.3 Risk-Free Rate

The risk-free rate used in this study is the LIBOR dollar rate. It is the average interest rate at which leading banks borrow funds from other banks in the London market. It is a widely used global "benchmark" or reference rate for short term investments. LIBOR is an acronym for; The London Interbank Offered Rate (Jamishidian, 1997).

The London Interbank Offered Rate is the average interest rate at which leading banks borrow funds from other banks in the London market (Jamishidian, 1997). It is the most widely used reference rate for short term interest rates. The rate is calculated and published by the Intercontinental Exchange. It's based on five currencies: the US dollar; the euro; the British Pound; the Japanese yen; and the Swiss franc. It's calculated for seven different time periods: overnight; one week; one month; two months; three months; six months; and 12 months (Jamishidian, 1997).

4.1.4 Regions

Countries were split up into the following regions during analysis. The data in question was supplied to the research team with these pre-set groups

Table 3 - Breakdown of Global region used in this study

Europe	EE	CC	Asia+	UK	US	Japan
France	Czech Republic	Australia	Hong Kong	UK	US	Japan
Austria	Hungary	Canada	Malaysia			
Belgium	Israel	Norway	Indonesia			
Denmark	Russia	New Zealand	Singapore			
Finland	Turkey		Thailand			
Greece			Taiwan			
Ireland			South Korea			
Italy			Egypt			
Netherland						
Portugal						
Sweden						
Spain						
Germany						
Switzerland						

4.5 Exploratory Analysis by SOM

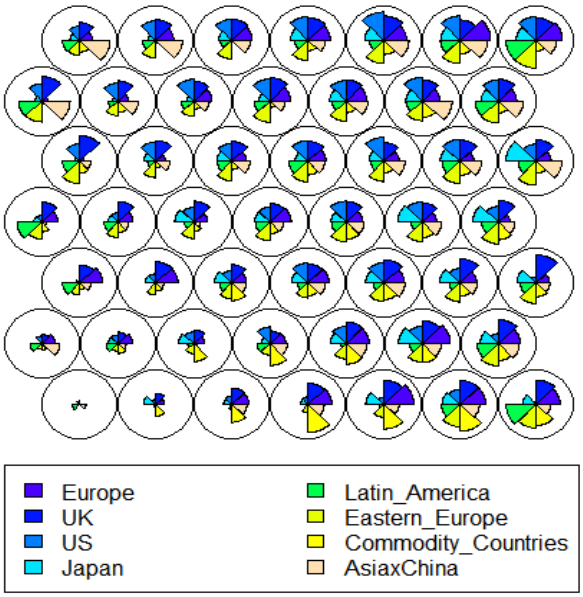


Figure 7 – Example of SOM created using Equity Flow dataset

Shown in figure 7 is a SOM produced in Rstudio using the kohonen package. This analysis technique allows one to get a general sense of the overall structure of the dataset. The SOM grid consists of many circular nodes, one can set the desired number of nodes depending on the size and nature of the dataset. The SOM presented overhead (figure 8) consist of 400 nodes (20x20). Inside each node, there are eight wedges of varying size, each wedge representing the magnitude of equity flow. There is approximately 350 data points of average weekly equity flow from 2012 to the present day. The above SOM was trained with a dataset of similar magnitude to the number of nodes it possesses. For example, in the bottom right of both figures one can observe that each of these nodes correspond to weeks during this period where the equity flows across all regions are of a large positive magnitude. Each node roughly corresponds to a week during this period. A disadvantage to a SOM of this size is that is can cause the nodes to be difficult to read.

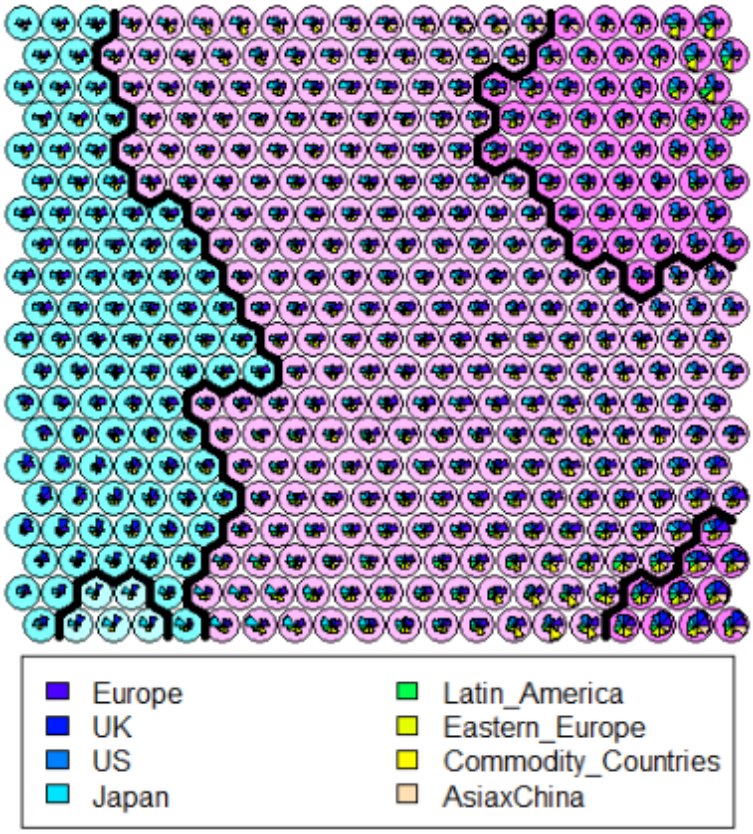


Figure 8 - 20x20 SOM created using Equity flow dataset showing four distinct regions

4.6 Determining Market Regimes

Weekly Equity flow data was imported into the statistical software tool Rstudio. It was here that the hierarchical clustering algorithm was implemented on the weekly data. The purpose of this was to cluster the weeks into four groups to form the four investor regimes.

4.7 Development of Model Portfolios

The four clusters of weeks were obtained, next was to characterise these clusters (regimes) in terms of stability and average weekly returns. Functions in Rstudio was used to obtain the average weekly returns in each region in each regime. The purpose of this was to use the results

of the average regional weekly returns to inform several model portfolios, these are explained in full in the following section.

4.7.1 Portfolio Models

Three different trading strategies were used to test the signals from our model. These trading strategies were selected to match the most common strategies used in financial markets. Traditional fund managers are limited to holding long positions only (Investments, Bodie, Kane & Markus), a long only portfolio replicates this strategy. Hedge funds are allowed more flexible strategies, combining long and short positions which allows them to take advantage of both long and short signals (Risks and Portfolio Decisions Involving Hedge Funds, Agarwal, Naik). They can approach this in two ways. First, taking a market neutral strategy, where long positions match short positions, this is denoted as the hedge fund strategy, and looks at the relative value of signals. The second is a long/short, which looks at the absolute value of each signal. We use a long only strategy to replicate their decisions. Three models were tested during this portion of the research and are outlined here. The long/short model is one which will either buy/sell or do nothing across all regions in a certain period. Utilisation of the long/short model has been an established investment strategy for many years (Grinold, Kahn, 2000). The long only model either buys or does nothing across all regions in a certain period. This model is another example of a commonly used investment strategy in the financial industry (Huij, Lansdorp, Blitz, Vliet, 2014). These first two modes are considered to be more traditional than the last type of investment model we consider: The Hedge fund model (Fung, Hsieh, 2004). Hedge fund models are described as being more diverse and dynamic than more traditional investment strategies (Fung, Hsieh, 2004). The hedge fund model implemented in this study is more dynamic than the long/short or long only models. This is because it takes long, short, or neutral positions in different regions in the same time period. Whereas the first two models

take long, short, or neutral positions across all regions in a certain time period.

Long/Short Model - This model either takes a long, neutral, or short position across all the regions each week.

Long Only Model - This model takes only long or neutral position in specific regions each week.

Hedge Fund Model - This model takes neutral, long or short positions in specific regions each week.

4.7.2 Benchmark Portfolio Models

The model portfolios informed by the results of the regime analysis are compared to two benchmark portfolio models. These benchmarks are the 'Buy and Hold' strategy and the LIBOR. The 'Buy and Hold' strategy involves equally buying assets in each region each week and holding those assets. In recent years, all regions have experienced positive returns overall. In other words, the market has continually gone up more so than down. Because of this the 'Buy and Hold' strategy is difficult to outperform. We display this benchmark portfolio in the results and graphs, but it is more appropriate to compare the portfolio models to the dollar LIBOR benchmark (Jamisidan, 1997).

Buy and Hold Strategy - This model takes long positions in every region, every week.

Risk-Free Rate - This model steadily increases at the rate of the LIBOR dollar rate.

4.8 Implementation of Portfolio Models

The statistical software Rstudio was the primary tool used in all aspects of data handling and analysis. Rstudio is a free to use and download software that provides a wide range of functions for data analysis. Rstudio is an integrated development environment (IDE) that allows one to develop programs in R, the programming language. One can install a huge number of packages into Rstudio that include an extensive set of functions for classical and modern statistical data analysis. Among many packages employed during this study, some main ones were the kohonen, dtwclust, datetime and ggplot packages. Kohonen provides helpful functions for the implementation of Self-Organising maps. Dtwclust contains a multitude of helpful functions for performing time-series clustering. DTW refers to dynamic time warping, the distance measure employed in the ML experiment portion of this study. The datetime package makes the cleaning and handling of time-series data. Finally, the ggplot was essential in the creation of many charts and visualisations representing the output of the analysis. Excel was used in conjunction with Rstudio. Excel is helpful for simple data cleaning and carrying out quick calculations and creating visualisations of the results. Important code extracts from Rstudio are presented in appendix III.

4.9 Portfolio Model Analysis

The primary indicators of model performance are described in detail here. Any relevant formulae are provided here in addition to arguments that validate the use of these measures.

4.9.1 Total Return

This term is also referred to as total cumulative return. Total return is a measure of an investment's overall performance (Kakushadze, 2017). It

is the actual rate of return of an investment over a given evaluation period. Total return accounts for capital gains, interest gained, dividends and distributions realized over time. Total return is the amount of value an investor earns from a security over a specific period when all distributions are reinvested. This value is expressed as a percentage.

A cumulative return on an investment is the aggregate amount that the investment has gained or lost over time, independent of the time period involved. Presented as a percentage, the cumulative return is the raw mathematical return of the following calculation (Kakushadze,2017).:

$$Total\ Return\ \% = \frac{(Current\ Value)-(Original\ Value)}{(Original\ Value)} x\ 100 \qquad (10)$$

4.9.2 Annualised Cumulative Return

An annualized total return is the geometric average amount of money earned by an investment each year over a given time period. It is calculated as a geometric average to show what an investor would earn over a period of time if the annual return was compounded (Kakushadze, 2017).

$$Annualised\ Cumulative\ Return = [(1 + r_1)(1 + r_2) \dots (1 + r_{n-1})(1 + r_1)]^{\frac{1}{n}} - 1 \qquad (11)$$

4.9.3 Volatility

Volatility is a statistical measure of the dispersion of returns for a given security or market index. In most cases, the higher the volatility, the riskier the security. Volatility can either be measured by using the standard deviation or variance between returns from that same security or market index. This study using the standard deviation as the volatility

measure for returns. Volatility refers to the amount of uncertainty or risk related to the size of changes in a security's value (Aizenman 1995)

$$Volatility = Standard\ Deviation = \sqrt{\frac{\sum_1^n (r_n - \mu)}{n}} \quad (12)$$

Where r_n is the return in week n, μ is the mean return over all weeks and n is the number of weeks.

4.9.4 Sharpe Ratio

The Sharpe ratio allows investors to compare the return of an investment to its risk. In general, the higher the Sharpe ratio, the more attractive the portfolio is to an investor. The recognized Sharpe ratio is

$$S = \frac{R_p - R_f}{\sigma_p} \quad (11)$$

Where R_p is the return of the portfolio, R_f is the risk-free rate and σ_p is the standard deviation of the portfolio's excess return (Sharpe, 1994).

4.10 Investigating Regime Stability

Probability matrices, also known as transition matrices, display the probability of transitioning from one state to another. They consist of a square matrix that gives the probabilities of various states changing from one to another staying in the same state (Aguiar, 1998). There are many acceptable names for this concept besides probability and transition matrices, they are also commonly referred to as Markov and

Stochastic matrices. Andrey Markov developed this concept in the early 20th century (Aguiar, 1988). To examine regime stability in this study we employ the use of 'Right stochastic matrices. In this type of stochastic matrix, the entries down each column sum to one. In contrast from a 'left stochastic matrix' where the entries across each row sum to one. A 'Doubly stochastic matrix' is a matrix where both the entries down each column and the entries across each row sum to one. This is helpful to known when reading the matrices presented later in this study. Table 4 overhead is an example of a transition matrix representing the probability of the model changing from one regime to another. We start by looking at the first column of the matrix which displays numbers one to four, these represent the four regimes. The second row of this matrix (highlighted in blue) represents the probability of regime one switching to any of the three regimes or staying in regime ne in the following week. For example, if the model resides in regime 1 some week, then there is a 0.11 chance of switching to regime two the following week.

Table 4 - Probability matrix of regimes made with obtained using flow data from 2012-2018

	1	**2**	**3**	**4**
1	0.779	0.110	0.000	0.110
2	0.190	0.660	0.050	0.100
3	0.059	0.235	0.706	0.000
4	0.321	0.196	0.000	0.482

Probability matrices are used in this study to illustrate the differences between using equity flow data to and return data to define market regimes. The two probability matrices shown in this section demonstrate the stability of the market regimes found created by first the equity flow data and secondly by utilising the return data.

Chapter 5

Results and Discussion

In this section we look at the results of the creation of a model portfolio by using self-organising maps, hierarchical clustering and dynamic time warping. The results include examples of the many SOMs created in Rstudio, bar charts of the average weekly returns by regimes and performance of the portfolio models created based on the analysis of these regimes.

5.1 Analysis of Flow Data by SOM

The first step of the analysis process involved the equity flow dataset. Exploratory analysis was primarily carried out using SOM's, as discussed previously, this has been established as a useful tool for this stage of analysis. This technique allows an analyst to get an overview of the dataset in question. Shown below is an example of a self-organising map produced using the total flow dataset, its smaller size makes it easier to read.

Shown overhead are four examples of SOM's produced in Rstudio, using the Kohonen package to analyse the equity flow data. This analysis techniques allows one to get a general sense of the overall structure of the dataset. The SOM grid consists of many circular nodes, one can set the desired number of nodes depending on the size and nature of the dataset. The above grids consist of 400 nodes (20x20). Inside each node, there are eight wedges of varying size, each wedge representing the magnitude of equity flow. There is approximately 350 data points of average weekly equity flow from 2012 to the present day. The above SOM was trained with a dataset of similar magnitude to the number of nodes it possesses. For example, in the bottom right of both figures one can observe that each of these nodes correspond to weeks during this period where the equity flows across all regions are of a large positive magnitude. Each node roughly corresponds to a week during this period. A disadvantage to a SOM of this size is that is can cause the nodes to be difficult to read.

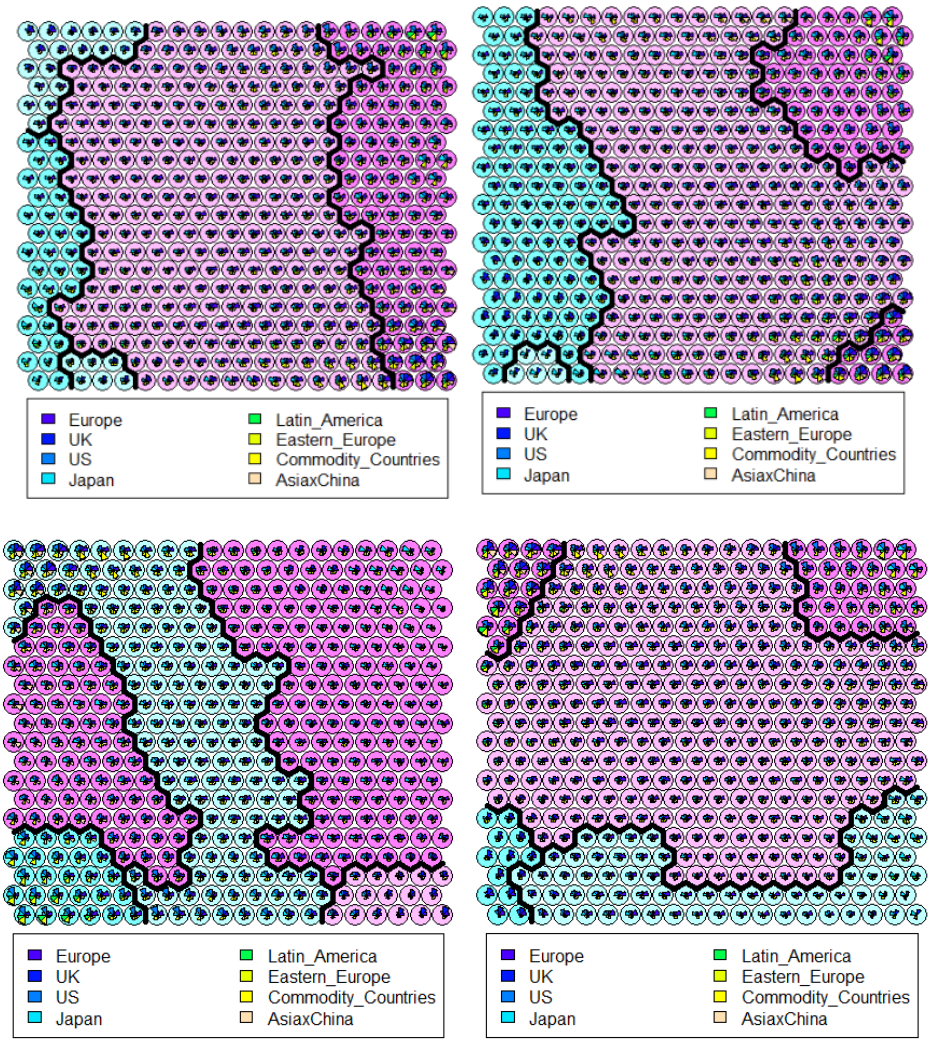


Figure 9 – Four SOM's created by equity flow data

This exploratory analysis ultimately helped to determine a good choice for the number of clusters. It gives an overall sense of the dataset and thus make further analysis more informed and straightforward.

5.2 Results and Discussion

The method of defining and characterising investor regimes regime was repeated a number of times, each time with some change of in variable. The process was carried on by clustering equity flow data to define regimes and then on return data to define regimes. This allowed for the creation of two sets of portfolio models; one set informed by regimes defines by equity flow and the other set informed by regimes defined by

returns. In this section we compare the performance of models informed by equity flows and those informed by returns. We find that in general models informed by equity flow data are more profitable than those created using returns. We examine the stability of the regimes defines using equity data and regimes defines by returns and find that the regimes defined by equity are significantly more stable.

In addition to defining regimes on different data sets, the process is repeated using two time periods. The first time-period examined in this section is from 2012 to 2018 and the second considered is from 1998 to 2018. We compare the performance of models created using data from the two time - periods. We observe again that the regimes defined using equity flow data are more stable than those defined by clustering returns.

5.2.1 2012-2018 'Equity Flow Regime' Analysis

This section presents the regional average weekly returns of the regimes obtained by clustering the total weekly equity flows from the period of January 2012 to July 2018. The average returns in each of the four regimes are found and inform the three investment models.

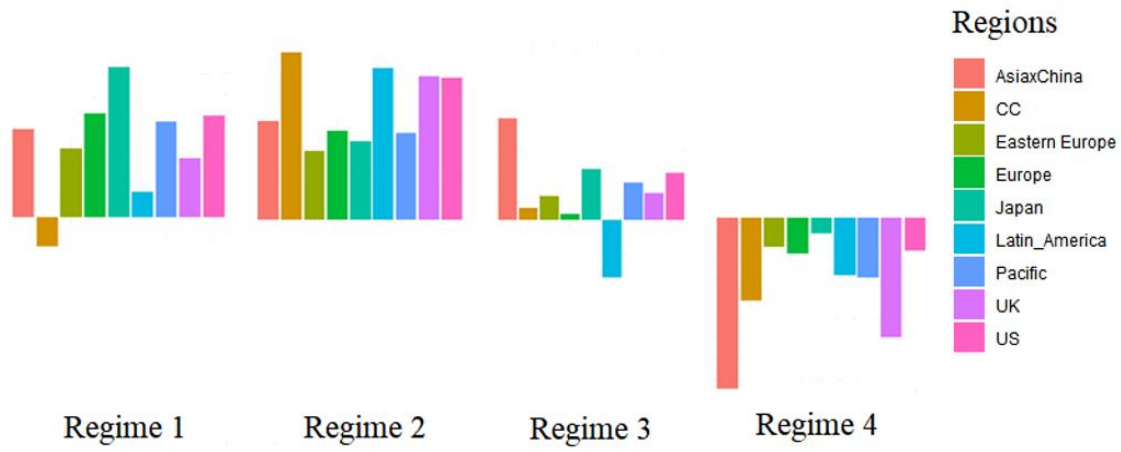


Figure 10 - Relative return levels across the four regimes created by clustering equity flows during 2012-2018

In the above plots, one can see that regime 1 and 2 are characterized by positive returns, regime 3 is generally low positive returns and returns are on average negative during regime 4 across all regions. In the 'long/short' model, the model takes a long position during regime 1 and 2, a neutral position during regime 3 and a short position in regime 4.

In the 'long only' model, the model takes a long position in all regions except the commodity countries and Latin America, in which it takes a neutral position. During Regime 2 the model takes a long position in all regions. In Regime 3, the model takes a long position in Asia, Japan, Pacific, UK and US. It takes a neutral position in all other regions. For regime 4, the model is neutral in all regions.

For the ‘Hedgefund’ model, the model takes a long position in all regions except the commodity countries (neutral) and Latin America (short) during regime 1. In regime 2 the model takes a long position in all regions. In regime 3, the model takes a long position in Asia, Japan, Pacific, UK and US. It takes a neutral position in all other regions except in Latin America where it takes a short position. For regime 4, the model takes a short position in all regions.

5.2.2 2012-2018 ‘Return Regime’ Analysis

The results from obtained by clustering the weekly returns from the period of January 2012 to July 2018 are presented and discussed here. The average returns in each of the four regimes are found and inform the three investment models.

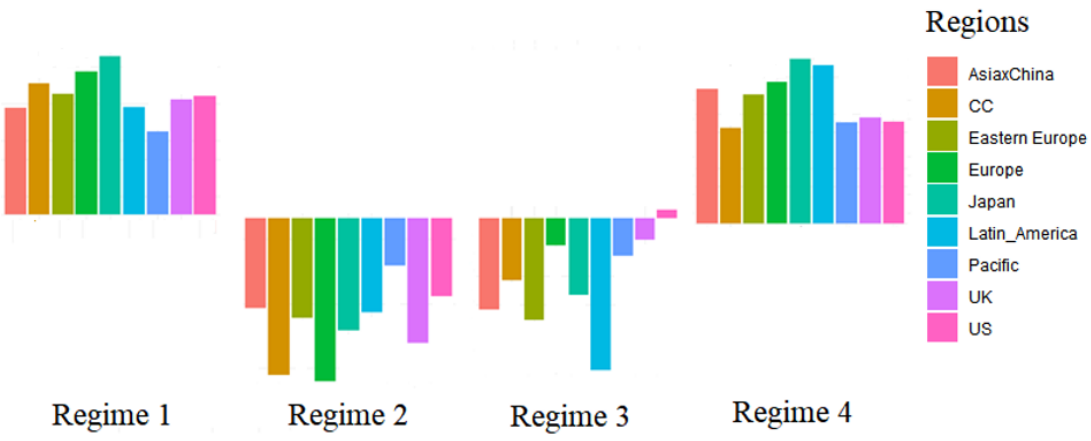


Figure 11 - Relative return levels across the four regimes created by clustering returns during 2012-2018

In the above plot (figure 11), one can see that regime 1 and 2 are characterized by positive returns, regime 3 is generally low positive returns and returns are on average negative during regime 4 across all regions. In the ‘long/short’ model, the model takes a long position during regimes 1 and 4, and a short position during regimes 2 and 3.

In the ‘long only’ model, the model takes a long position across all regions during regime 1 and 4. During regimes 2 and 3 the model takes a neutral position across all regions.

For the ‘Hedgefund’ model, the model takes a long position in all regions during regime 1. In Regime 2 the model takes a short position in all regions. In Regime 3, the model takes a short position in all countries except Europe, pacific, UK and US which it takes a neutral position in. The model takes a long position in all regions during regime 4.

5.2.3 1998 – 2018 ‘Equity Flow Regime’ Analysis

Here we examine the results from obtained by clustering the total weekly equity flows from the period of January 1998 to July 2018. The average returns in each of the four regimes are found and inform the three investment models.

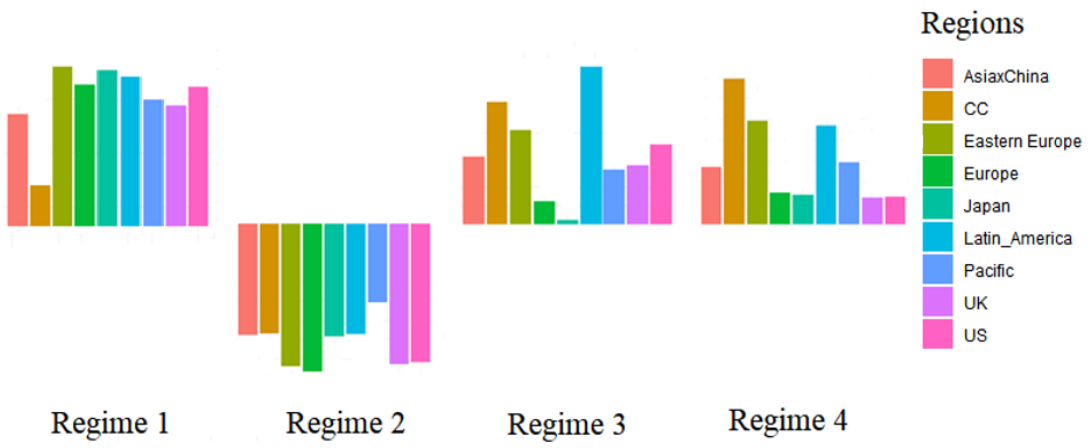


Figure 12 - Relative return levels across the four regimes created by clustering equity flows during 1998-2018

In the above plot (figure 12), one can see that regime 1 and 2 are characterized by positive returns, regime 3 is generally low positive returns and returns are on average negative during regime 4 across all

regions. In the ‘long/short’ model, the model takes a long position during regimes 1, 3 and 4, and a short position during regime 2.

In the ‘long only’ model, the model takes a long position across all regions during regime 1 and 4. During regimes 2 and 3 the model takes a neutral position across all regions.

For the ‘Hedgefund’ model, the model takes a long position in all regions during regime 1. In Regime 2 the model takes a short position in all regions. In Regime 3, the model takes a short position in all countries except Europe, pacific, UK and US which it takes a neutral position in. The model takes a long position in all regions during regime 4.

5.2.4 1998 - 2018 'Return Regime' Analysis

Here we examine the results from obtained by clustering the total weekly equity flows from the period of January 1998 to July 2018. The average returns in each of the four regimes are found and inform the three investment models.

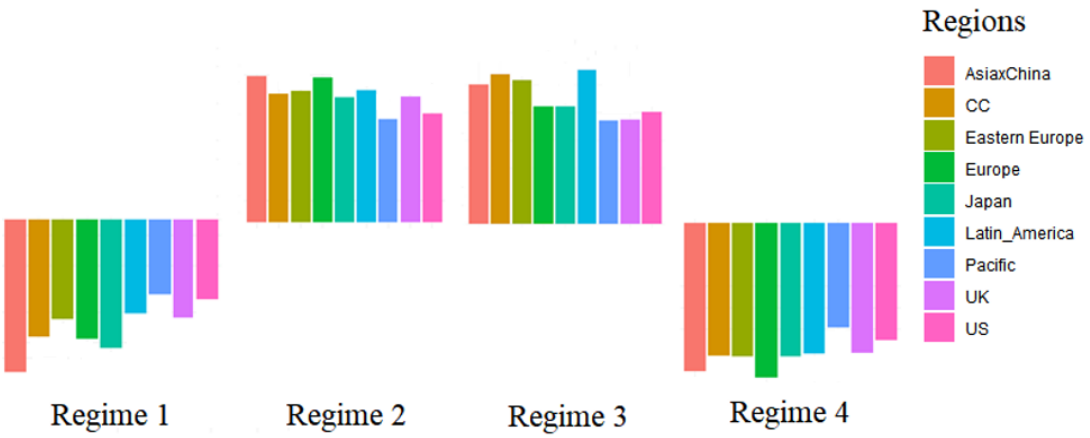


Figure 13 - Relative return levels across the four regimes created by clustering returns during 1998-2018

In the above plot (figure 13), one can see that regime 1 and 2 are characterized by positive returns, regime 3 is generally low positive returns and returns are on average negative during regime 4 across all regions. In the 'long/short' model, the model takes a long position during regime 3 and 4 and a short position during regime 1 and 4.

In the 'long only' model, the model takes a long position during regime 3 and 4 and a neutral position during regime 1 and 4.

The regimes created during this iteration have average weekly returns that are either positive across all regions or negative across all regions. Therefore, the hedge fund model worked out to be the same strategy as the long/short model, and hence is not included here.

5.3 Regime Stability Analysis

Section 6.2 presented the results of the four iterations of creating investor regimes to inform a set of model portfolios. We saw the average weekly regional returns across the investor regimes, and they inform the creation of the model portfolios. In this section we compare the stability of each set of investor regimes. We begin by looking at the stability of the regimes created in the time period of 2012-2018. First looking at the stability of the investor regimes obtained by clustering equity flow data. The probability matrix for these regimes is shown below in table 5.

Table 5 - Probability matrix of clusters made with obtained using flow data from 2012-2018

	1	**2**	**3**	**4**
1	0.779	0.110	0.000	0.110
2	0.190	0.660	0.050	0.100
3	0.059	0.235	0.706	0.000
4	0.321	0.196	0.000	0.482

Consider table 5, the highest numbers in this table are along the diagonal, from the top left corner to the bottom right corner. This indicates to us that these regimes are relatively stable, as from week to week the model is more likely to stay in the same regime rather than switch to a different regime. This is an important result as a primary hypothesis stated in the beginning of this study was that if the market resides in a certain regime one week then it will most likely still reside in that same regime in the following week. The model portfolios were created based on this assumption. Table 6 present the transition matrix of the investor regimes obtained by clustering weekly regional returns rather than equity flow.

Table 6 - Probability matrix of clusters made with obtained using return data from 2012-2018

	1	2	3	4
1	0.152	0.261	0.326	0.261
2	0.126	0.336	0.263	0.274
3	0.139	0.278	0.306	0.278
4	0.115	0.229	0.364	0.292

In Table 6 above, we see that the numbers along the diagonal (top left to bottom right) are no larger than all other numbers in the table. This result indicates that these regimes are not stable, that the model is as likely to switch regimes than to stay in a certain regime. We see later how this result effects how profitable the model informed by these regimes is compared to the model portfolio created based of regimes obtained by clustering equity flow data.

The following two transition matrices presented represent the stability of investor regimes created by clustering equity flow data from 1998 to 2018 (table 7), and clustering return data from 1998 to 2018 (table 8).

Table 7 - Probability matrix of clusters made with obtained using flow data from 1998-2018

	1	2	3	4
1	0.747	0.038	0.120	0.016
2	0.357	0.554	0.054	0.038
3	0.287	0.023	0.606	0.085
4	0.050	0.008	0.231	0.712

Look to table 7 above. Like table 5, the highest numbers in table 7 are along the diagonal (top left to the bottom right). This result demonstrates that these regimes are relatively stable. Table 8 presents

the transition matrix of the investor regimes obtained by clustering weekly regional returns rather than equity flow.

Table 8 - Probability matrix of clusters made with obtained using return data from 1998-2018

	1	**2**	**3**	**4**
1	0.343	0.153	0.305	0.199
2	0.327	0.129	0.361	0.184
3	0.395	0.110	0.361	0.134
4	0.269	0.274	0.238	0.220

Lastly, we consider table 8. This table displays the probabilities of regimes changing from one to another each week when the regimes were obtained by clustering weekly return data from 1998 to 2018. Similarly, to table 5, we see that the numbers along the diagonal (top left to bottom right) are not significantly larger than all the other numbers in the table. This indicates that these regimes are not particularly stable, as they are as likely to switch into a different regime each week as to stay in the same regime.

The main takeaway from this set of results, is that regimes created by clustering regional equity flows are more stable than regimes created by clustering regional returns. In section 6.4 we investigate how profitable portfolios informed by both sets of regimes are. We will go on to draw connections between how profitable a portfolio model is and how stable the underlying regimes are in that model.

5.4 Model Performance

Here the results of the model portfolios are presented and discussed. We begin by looking at the main result of this experiment. We compare a hedge fund style model portfolio that was created by clustering returns to one that was created by clustering equity flows. Section 6.4.1 illustrates that the model created by clustering equity flows outperforms the model created by clustering returns. This primary result was presented in an essay submitted to the 2019 CFA quant award competition. This essay is presented in appendix II. This result highlights the benefit of creating regimes using equity flow rather than return data in addition to showcasing ML as a useful toll in the field of quantitative finance.

In section 6.4.2 we look at the results of all portfolio models informed by regimes clustered using equity flow data from 2012 to 2018. Then we look at the models informed by regimes obtained by clustering return data in the same time period. Section 6.4.1 presents the results of the model informed by the set of regimes obtained by clustering equity flow data from 2012 to 2018. Lastly, we look at the model results where the underlying regimes were created by clustering returns in the same time period.

These results will allow comparisons to be made between models that rely on equity flow data vs return data. Conclusions can also be drawn of the differences in using the two time periods to create investment regimes. We evaluate a model's performance based on the following criteria; total returns, annual cumulative return, volatility and Sharpe ratio.

'Equity Flow' regime models refer to models informed by the regimes obtained by clustering weekly regional equity flow data. While 'Return'

regime models refer to models informed by the regimes obtained by clustering weekly regional return data.

5.4.1 Model Portfolio Comparison

The following two graphs highlight the difference in performance of the two hedge fund style models informed by regimes created using equity flow and return data in the time period of 2012 to 2018.

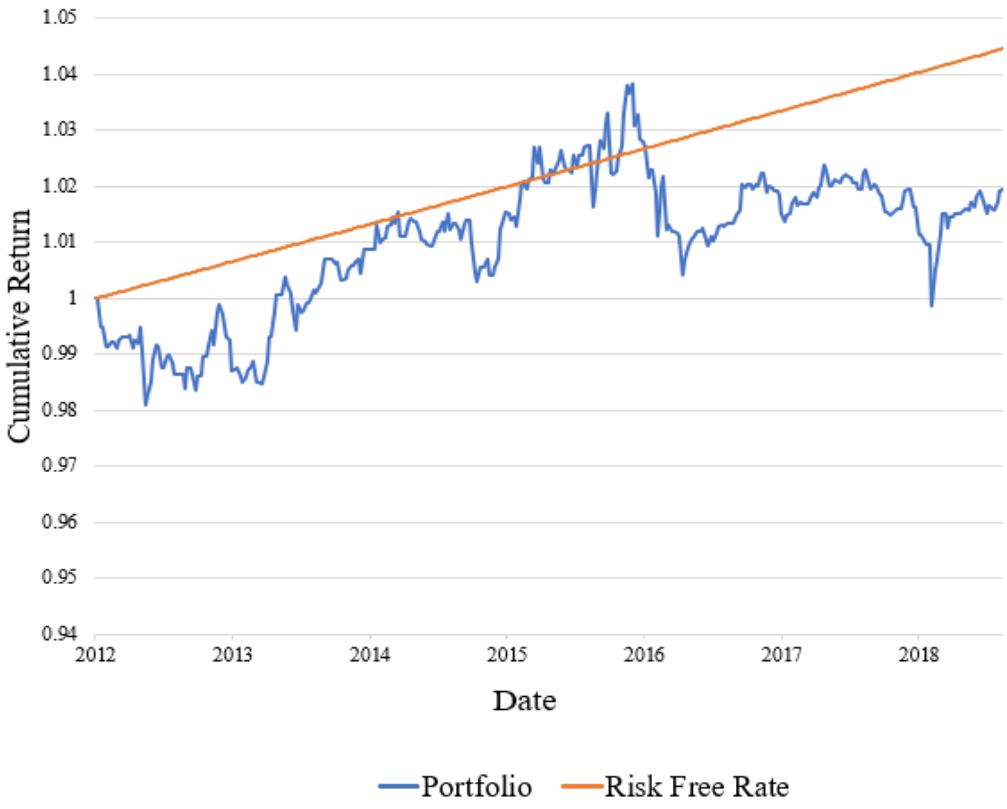


Figure 14 - Cumulative returns plot of hedge fund model informed by 'return regimes' in the period 2012 to 2018.

Figure 18 above shows the results of the hedge fund style model informed by 'return regimes' (ie. Regimes informed by clustering returns). The blue line represents this portfolio and we see that is does not outperform the LIBOR rate. This indicates that this model portfolio is not profitable for an investor. We now compare this to figure 19.

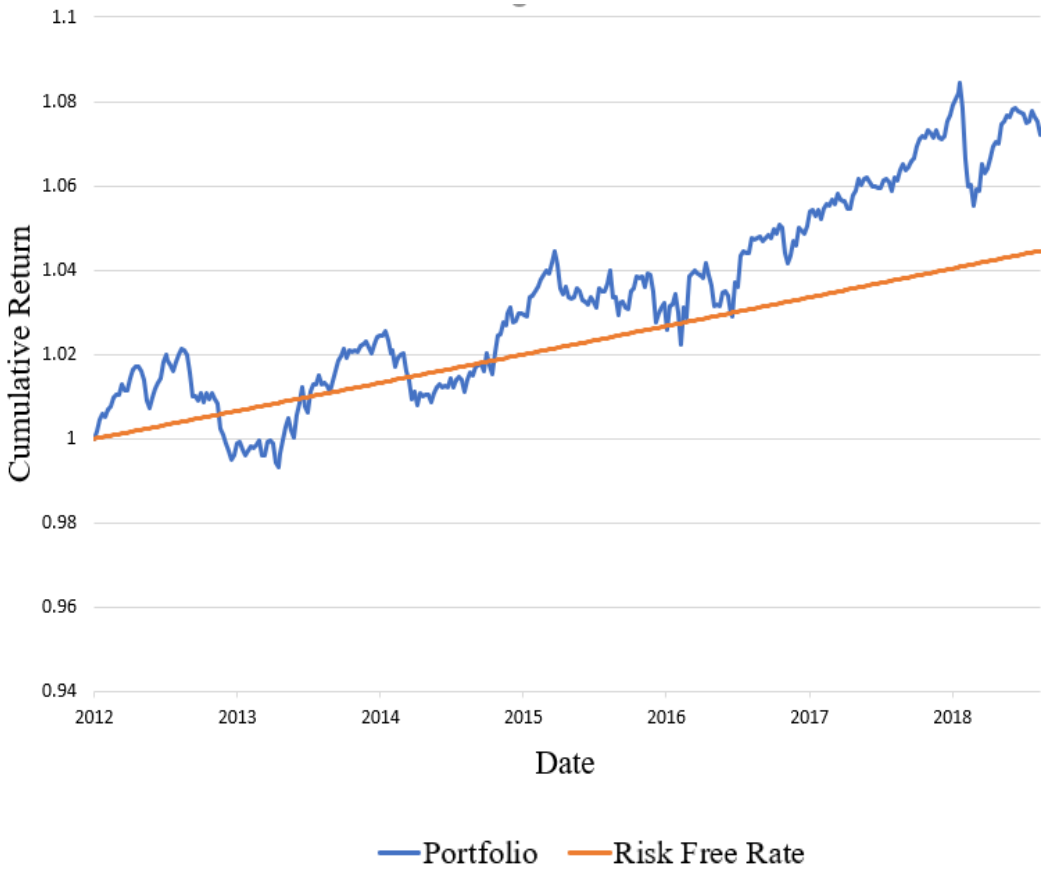


Figure 15 - Cumulative returns plot of hedge fund model informed by 'Equity flow regimes' in the period 2012 to 2018.

Figure 19 above shows the results of the hedge fund style model informed by 'equity regimes' (ie. Regimes informed by clustering equity flow). In this case we see that the model portfolio outperforms the LIBOR rate. This indicates that this model portfolio is profitable for an investor. This result shows that the model informed by the 'equity regimes' is more profitable than the portfolio model informed by the 'return regimes' (ie. Regimes informed by clustering returns).

5.4.2 'Equity Flow' Regime Models 2012-2018

The following models were informed by the characterisation of the regimes obtained by clustering equity flow data from 2012 to 2018.

Table 9 - Cumulative returns plot of all models informed by regimes created by clustering equity flows from 2012-2018.

	LONG/SHORT MODEL	LONG ONLY MODEL	HEDGE FUND MODEL	BUY AND HOLD MODEL	LIBOR RATE
TOTAL RETURN%	6.010	10.69	7.21	14.20	4.46
ANNUAL RETURN%	0.90	1.15	1.05	2.02	0.66
VOLATILITY%	2.26	1.77	1.64	2.27	-
SHARPE RATIO	0.11	0.50	0.24	0.60	-

These results in table 9 show that all models result in a positive Sharpe ratio. As mentioned previously, it is difficult to outperform the Buy and hold strategy since the market has been steadily increasing during this time period. Therefore, it is perhaps more appropriate to compare the models to the LIBOR rate, LIBOR is taken to be the risk-free rate in for this study. The long only model performs best when compared to the Hedge fund and long/short model. Cumulative returns for these portfolio models are depicted in figure 14 overhead.



Figure 16 - Cumulative returns plot of all models informed by regimes created by clustering equity flows from 2012-2018 plot.

5.4.3 'Return' Regime Models 2012-2018

These regimes are referred to as 'Return' Regimes as they were obtained by clustering regional returns during the period 2012-2018.

Table 10 - Cumulative returns plot of all models informed by regimes created by clustering returns from 2012-2018.

	LONG/SHORT MODEL	LONG ONLY MODEL	HEDGE FUND MODEL	BUY AND HOLD MODEL	LIBOR RATE
TOTAL RETURNS%	2.48	9.27	1.96	14.20	4.46
ANNUAL RETURN%	0.37	1.35	0.29	2.02	0.66
VOLATILITY%	1.90	1.62	1.79	2.27	-
SHARPE RATIO	-0.15	0.43	0.02	0.60	0

The results in table 10 show that the portfolio models created relying solely on return data perform worse than the models created by clustering equity flow data. The Sharpe ratio is negative at -0.153 for the long/short model compared to 0.105 in the long/short model informed by the 'equity flow' regimes. The long only model had better results with a Sharpe ratio of 0.426 but was still outperformed by the long only model informed by the equity flow regimes which was calculated t have a Sharpe ratio of 0.498. The hedge fund model has a barely positive ratio of 0.018, significantly worse than the model informed by 'equity flow' regimes which had a ratio of 0.240. These results indicate that regimes obtained by clustering equity flow data work as better market indicators than regimes obtained by clustering return data.

Figure 15 works as a good comparison to figure 14. We can see that the Annual Cumulative returns of the Long/Short, long only and hedge fund models are lower than those of the models informed by the 'equity flow' regimes.



Figure 17 - Cumulative returns plot of all models informed by regimes created by clustering returns from 2012-2018.

5.4.4 'Equity Flow' Regime Models 1998-2018

The following models were informed by the characterisation of the regimes obtained by clustering equity flow data from 1998 to 2018.

Table 11 - Cumulative returns plot of all models informed by regimes created by clustering equity flows from 1998-2018

	LONG/SHORT MODEL	LONG ONLY MODEL	HEDGE FUND MODEL	BUY AND HOLD MODEL	LIBOR RATE
TOTAL RETURN%	192.83	145.11	244.36	260.84	12.55
ANNUAL RETURN%	6.16	5.27	7.12	7.40	0.66
VOLATILITY %	15.60	12.58	12.11	15.59	-
SHARPE RATIO	0.35	0.123	0.53	0.43	0

The results in table 11 show that all models result in a positive Sharpe ratio. The returns generated by these models are visualised in figure 16 overhead. The hedge fund model works well, exceeding the buy and hold strategy which a mentioned previously, it is difficult to do when the market is increasing as it has in recent years.

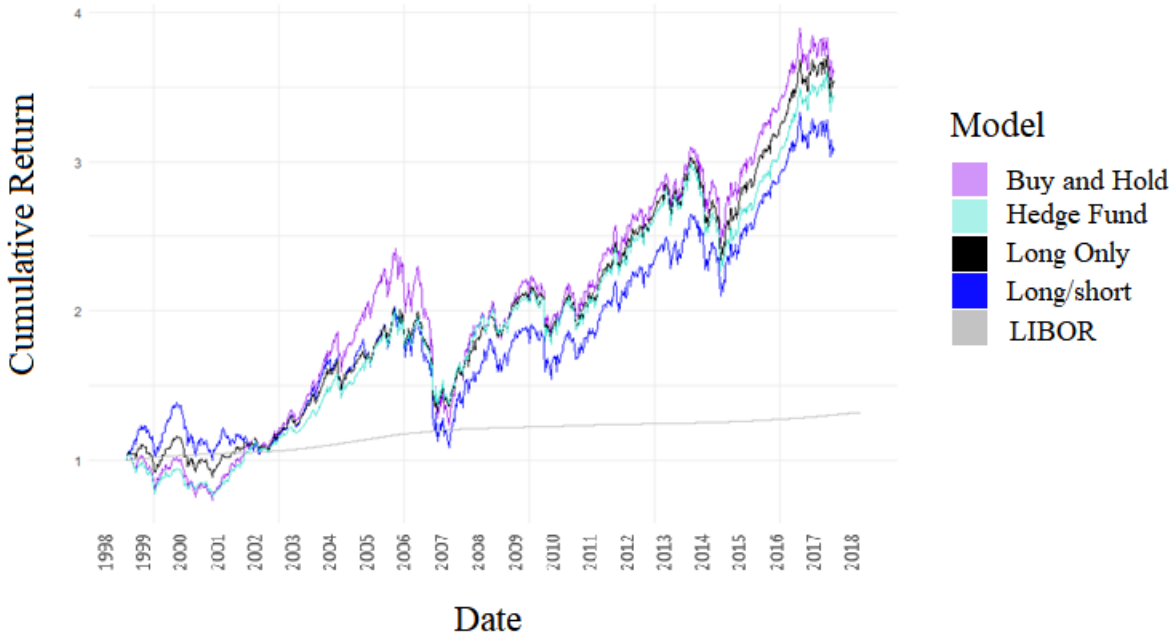


Figure 18 - Cumulative returns of all portfolio models and benchmarks, regimes obtained by clustering equity flows from 1998 to 2018.

5.4.5 'Return' Regime Portfolio Models 1998-2018

The following models were informed by the characterisation of the regimes obtained by clustering equity flow data from 1998 to 2018.

Table 12 - Cumulative returns plot of all models informed by regimes created by clustering returns from 1998 – 2018.

	LONG/SHORT MODEL	LONG ONLY MODEL	BUY AND HOLD MODEL	LIBOR RATE
TOTAL RETURNS %	192.57	145.11	260.84	12.55
ANNUAL RETURN %	6.15	5.11	7.40	0.66
VOLATILITY %	15.60	10.90	15.59	-
SHARPE RATIO	0.35	0.41	0.43	-

The hedge fund model informed by the 'equity flow' regimes can be compared to the long/short model informed by the 'return regimes'. The

regimes created by the return data have either positive or have negative returns across all regions each week, thus the model either takes a long or short position equally across all regions each week. The Sharpe ratio of the long/short model informed by 'return regimes' was calculated to be 0.35 while the hedge fund model informed by the 'equity flow regimes' was found to have a Sharpe ratio of 0.53. Once again we see that the models informed by the 'equity flow regimes' result in better performing hedge fund style portfolio models.

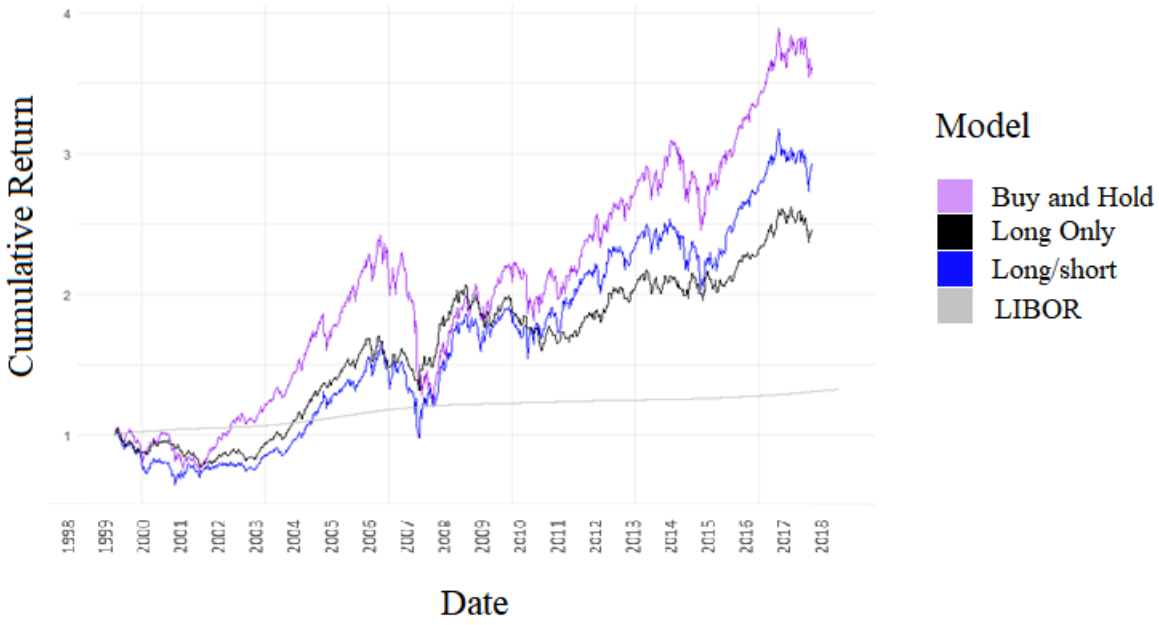


Figure 19 - Cumulative returns plot of all models informed by regimes created by clustering returns from 1998 – 2018.

Chapter 6

Conclusions

This section reiterates the original thesis statement and shows how the results of this experiment answer the research questions. The first question this study sought to answer was ‘Where does the cutting edge of machine learning for financial applications lie?’. More broadly; what applications ML is widely used to enhance and what ML technologies are being used to do so. A systematic literature search and review was carried out. The conclusions and main insights of this literature review are detailed and discussed in the literature review, sections 2.5-2.6.

The second question this study investigated was ‘Can ML techniques and technologies be used to devise a profitable portfolio?’ Using the results from this research, an ML proof of concept experiment was devised and implemented. The Ml technologies used were that of SOMs, hierarchical clustering and dynamic time warping. We look at the resulting portfolio models created by these technologies. The ML technologies were used on both equity flow data and MSCI data. The resulting portfolios from both data sets.

This study wished to investigate the validity of SOM technology as a useful exploratory analysis tool to determine an appropriate number of investor regimes to search for moving forward in the experiment. We determine these investor regimes and first compare the stability of the regimes determined using equity flow data and the regimes determined using return data. We compare portfolio models informed by ‘equity’ regimes to those informed by ‘return’ regimes. We conclude by discussing this experiment as a proof of concept for the use of ML technology to create a profitable and reliable portfolio and to assist in the field of quantitative investing.

6.1 Determining Number Regimes by SOM

The self-organising map technology was a useful and effective tool in the exploratory analysis of the equity flow data. By repeatedly using this technology and changing the number of clusters one can quickly and easily determine what is appropriate number of clusters to use when implementing the more sophisticated clustering algorithm of hierarchical clustering with dynamic time warping distance measure. Many SOMs were produced in order to determine the optimal number of regimes to look for. An example of one of these maps produced using the equity flow data is given in figure 3 where four distinct regions can be observed. Thus, for the next portion of the study, four regimes were determined and analysed to inform the portfolio.

This work showcases SOMs as a useful exploratory analysis tool. Figures 7 and 8 are example of SOMs created in Rstudio during the process of the ML experiment. These visualisations are easy to read and intuitive to read. They assisted in getting an overall sense of the data and in the selection of an appropriate number of investor regimes.

6.2 Stability Analysis of Regimes

Stability analysis of the four investor regimes were carried out using transition matrices. These matrices give the probabilities of regimes switching from one to another. To evaluate the stability, we look at the magnitude of the probabilities from the top left corner to the bottom right corner and compare these numbers to all other entries in the matrix.

The probability matrix of market regimes produced from equity flow data, shows that these clusters are more stable than clusters produced from clustering return data. These results show some of the benefits of using equity flow data to inform portfolio management decisions and strengthens the hypothesis that equity flows are more persistent and stable in comparison to return data. This result is seen across the iteration involving the time period from 2012 to 2018 and the longer time period from 1998 to 2018.

6.3 Portfolio Model Performance

The results of this study showed that the portfolio model results in higher returns when the equity flow data is used to create the market regimes. Consider the iteration of the ML experiment using data in the time period from 2012-2018. The model created by clustering equity flow data was calculated to have a Sharpe ratio of of 0.24 while the

model created using the analysis of return data is shown to have a Sharpe ratio of -0.21. This result indicates that the model created using the equity flow data outperforms the model informed using only the return data.

6.4 ML as a Quantitative Investing Tool

The potential to use modern machine learning techniques to create profitably investment models has been shown during this study. Furthermore, this study shows the advantages of determining investor regimes using equity flow data in comparison to using return data. This is shown in the fact that the resulting portfolio is more profitable and less risky in addition to the regimes being more reliable. Self-organising maps were helpful in the exploratory analysis of large financial datasets and assisted in the selection of an appropriate number of market regimes to define going forward in the research. Hierarchical clustering used in conjunction with dynamic time warping were successfully implemented to inform a regime portfolio.

The results of the literature search and ML experiment shows how ML has exciting potential for quantitative managers. During the literature discussion, we saw how this technology significantly expands their analytical toolkit. Advantages include a wide variety of languages and software that have been developed specifically for the creation and implantation of ML such as python, R and many more. The literature review noted that many authors noted the ability of ML to model nonlinear relationships and how this aspect is helpful in the area of quantitative investing.

The literature search also highlights some disadvantages to the use of ML for data analysis. Many sources point out that implementing ML technology on data can lead to overfitting. ML can often provide useful insights but the ‘black box’ nature of some algorithms can make it difficult for investors to back up their decisions. A ‘black box’ refers to

refers to the fact that ML algorithms take in data and output insights without explanation, and the algorithms are often so complex it is hard to determine the inner working of the technology and the reasoning behind decisions.

6.4 Closing Statement

In conclusion, the main takeaways from this study are as follows; The broad literature search results show that ML is commonly applied to the areas of Return Forecasting, Portfolio Construction, Ethics, Fraud Detection Decision Making Language Processing and Sentiment analysis. The more focused literature search that looked at ML applications for quantitative finance in very recent papers showed high interest in areas of return forecasting, portfolio construction, and risk modelling. A proof of concept experiment presented how ML techniques can be used in the construction of a portfolio. This experiment further showed the benefits of using equity flow data over return data in the construction of a model portfolio, when using regime investing.

References

M. Abe and H. Nakayama, "Deep Learning for Forecasting Stock Returns in the Cross-Section," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2018, pp. 273-284: Springer.

M. Achab, S. Clémençon, and A. Garivier, "Profitable Bandits," presented at the Proceedings of *The 10th Asian Conference on Machine Learning, Proceedings of Machine Learning Research*, 2018. Available: http://proceedings.mlr.press

Agarwal, V., & Naik, N. Y. (2004). Risks and portfolio decisions involving hedge funds. The Review of Financial Studies, 17(1), 63-98.

Aguiar, R. J., Collares-Pereira, M., & Conde, J. P. (1988). Simple procedure for generating sequences of daily radiation values using a library of Markov transition matrices. *Solar Energy*, *40*(3), 269-279.

A. Al-Aradi and S. Jaimungal, "Active and Passive Portfolio Management with Latent Factors," arXiv preprint arXiv:1903.06928, 2019.

J. Alberg and Z. C. Lipton, "Improving Factor-Based Quantitative Investing by Forecasting Company Fundamentals," arXiv preprint arXiv:1711.04837, 2017.

Anway, M. D., Cupp, A. S., Uzumcu, M., & Skinner, M. K. (2005). Epigenetic transgenerational actions of endocrine disruptors and male fertility. *science*, *308*(5727), 1466-1469.

Ahmed, N. K., A. F. Atiya, N. E. Gayar and H. El-Shishiny (2010). "An Empirical Comparison of Machine Learning Models for Time Series Forecasting." *Econometric Reviews* **29(5**-6): 594-621.

R. Arnott, C. R. Harvey, and H. Markowitz, "A Backtesting Protocol in the Era of Machine Learning," *The Journal of Financial Data Science*, vol. 1, no. 1, pp. 64-74, 2019.

Aizenman, J., & Marion, N. (1995). *Volatility, investment and disappointment aversion* (No. w5386). National bureau of economic research.

Balcilar, M., Demirer, R., & Hammoudeh, S. (2013). Investor herds and regime-switching: Evidence from Gulf Arab stock markets. *Journal of International Financial Markets, Institutions and Money*, 23, 295-321.

R. Barga, V. Fontama, and W. H. Tok, "Introducing Microsoft Azure Machine Learning," *in Predictive Analytics with Microsoft Azure Machine Learning*: Springer, 2015, pp. 21-43.

Basseville, M. (1989). Distance measures for signal processing and pattern recognition. *Signal processing*, *18*(4), 349-369.

Y. L. Becker and M. R. Reinganum, The Current State of Quantitative Equity Investing. *CFA Institute Research Foundation*, 2018.

A. Belloni, V. Chernozhukov, and C. Hansen, "Inference on treatment effects after selection among high-dimensional controls," *The Review of Economic Studies*, vol. 81, no. 2, pp. 608-650, 2014.

Bender, J., Briand, R., Melas, D., & Subramanian, R. A. (2013). Foundations of factor investing. Available at SSRN 2543990. D. Bianchi, M. Büchner, and A. Tamoni, "Bond risk premia with machine learning," Available at SSRN 3232721, 2018.

Birant, D., & Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering*, *60*(1), 208-221.

Bodie, Z., Kane, A., & Marcus, A. J. (2011). Investments. New York: McGraw-Hill/Irwin.

Boiy, E., & Moens, M. F. (2009). A machine learning approach to sentiment analysis in multilingual Web texts. Information retrieval, 12(5), 526-558.

H. Buehler, L. Gonon, J. Teichmann, and B. Wood, "Deep hedging," *Quantitative Finance*, pp. 1-21, 2019.

Burkov, A. (2019). *The Hundred-Page Machine Learning Book*. Andriy Burkov.

Carhart, M. M. (1997). "On Persistence in Mutual Fund Performance." *Journal of Finance* **52**(1): 57-82.

Chen, I.-F. and C.-J. Lu (2017). "Sales Forecasting by Combining Clustering and Machine-Learning Techniques for Computer Retailing." *Neural Computing and Applications* **28**(9): 2633-2647.

J. C. Chow, "Analysis of Financial Credit Risk Using Machine Learning," arXiv preprint arXiv:1802.05326, 2018.

Chowdhury, G. G. (2003). "Natural language processing." *Annual Review of Information Science and Technology* **37**(1): 51-89.

G. Choy et al., "Current applications and future impact of machine learning in radiology," *Radiology*, vol. 288, no. 2, pp. 318-328, 2018.

Chui, M., J. Manyika, M. Miremadi, N. Henke, R. Chung, P. Nel and S. Malhotra (2018). Notes from the AI Frontier: Applications and Value of Deep Learning. *Notes from the AI frontier*, McKinsey & Company.

Cochrane, J. H. (2011). "Presidential Address: Discount Rates." *Journal of Finance* **66**(4): 1047-1108.

Colombo, M. (2016). "Why Build a Virtual Brain? Large-Scale Neural Simulations as Jump Start for Cognitive Computing." *Journal of Experimental & Theoretical Artificial Intelligence* **29**(2): 361-370.

Danielsson, P. E. (1980). Euclidean distance mapping. *Computer Graphics and image processing*, *14*(3), 227-248.

L. Deng and D. Yu, "Deep learning: methods and applications," *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197-387, 2014

M. Dixon, D. Klabjan, and J. H. Bang, "Classification-based financial markets prediction using deep neural networks," *Algorithmic Finance*, no. Preprint, pp. 1-11, 2017.

P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78-87, 2012.

Fabozzi, F. J. (2005). *Bond Markets, Analysis and Strategies"(Int'l Edition)–5th Edition.* Prentice Hall.

Fama, E. F. and K. R. French (1993). "Common Risk Factors in the Returns on Stocks and Bonds." *Journal of Financial Economics* **33**(1): 3-56.

Fama, E. F. and K. R. French (2016). "Dissecting Anomalies with a Five-Factor Model." *The Review of Financial Studies* **29**(1): 69-103.

Fama, E. F. and J. D. MacBeth (1973). "Risk, Return, and Equilibrium: Empirical Tests." *Journal of Political Economy* **81**(3): 607-636.

S. Fecamp, J. Mikael, and X. Warin, "Risk management with machine-learning-based algorithms," arXiv preprint arXiv:1902.05287, 2019.

Feng, G., Giglio, S., & Xiu, D. (2017). Taming the factor zoo. *Fama-Miller Working Paper*, *24070*.

Ke, Q., Ferrara, E., Radicchi, F., & Flammini, A. (2015). Defining and identifying sleeping beauties in science. *Proceedings of the National Academy of Sciences*, *112*(24), 7426-7431.

Ferrucci, D., E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg and J. Prager (2010). "Building Watson: An Overview of the DeepQA Project." *AI magazine* **31**(3): 59-79.

E. Fons, P. Dawson, J. Yau, X.-j. Zeng, and J. Keane, "A novel dynamic asset allocation system using Feature Saliency Hidden Markov models for smart beta investing," arXiv preprint arXiv:1902.10849, 2019.

French, C. W. (2003). "The Treynor Capital Asset Pricing Model." *Journal of Investment Management* **1**(2): 60-72.

Fung, W., & Hsieh, D. A. (2004). Hedge fund benchmarks: A risk-based approach. Financial Analysts Journal, 60(5), 65-80.

Brian Garvey, Hsin-Chieh Chen (2004) *Mapping investment regimes* [White paper]

Geyer, C. J. (1992). Practical markov chain monte carlo. *Statistical science*, 473-483.

I. Goodfellow et al., "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672-2680.

J. Gonzalvez, E. Lezmi, T. Roncalli, and J. Xu, "Financial Applications of Gaussian Processes and Bayesian Optimization," arXiv preprint arXiv:1903.04841, 2019.

L. Goudenège, A. Molent, and A. Zanette, "Gaussian Process Regression for Pricing Variable Annuities with Stochastic Volatility and Interest Rate," arXiv preprint arXiv:1903.00369, 2019

Graham, B. and D. Dodd (2008). *Security Analysis: Foreword by Warren Buffett, McGraw-Hill Professional.*

Grinold, R. C. and R. N. Kahn (2000). "Active Portfolio Management."

Hamilton, J. D. (2016). Regime switching models, *The new palgrave dictionary of economics*, 1-7.

Harvey, C. R., & Liu, Y. (2018). Lucky factors. Available at SSRN 2528780.

Harrington, P. (2012). "Machine Learning in Action." *Shelter Island, NY: Manning Publications Co.*

R. Harvey, Y. Liu, and H. Zhu, "… and the Cross-Section of Expected Returns," *The Review of Financial Studies*, vol. 29, no. 1, pp. 5-68, 2016.

Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR.

J. Heaton, N. Polson, and J. H. Witte, "Deep learning for finance: deep portfolios," *Applied Stochastic Models in Business and Industry*, vol. 33, no. 1, pp. 3-12, 2017.

R. Hisano, D. Sornette, and T. Mizuno, "Predicting Adverse Media Risk using a Heterogeneous Information Network," arXiv preprint arXiv:1811.12166, 2018.

Homescu, C. (2015). Better investing through factors, regimes and sensitivity analysis. *Regimes and Sensitivity Analysis (January 25, 2015)*.

Hitt, M. A., Tihanyi, L., Miller, T., & Connelly, B. (2006). International diversification: Antecedents, outcomes, and moderators. *Journal of Management*, *32*(6), 831-867.

Huang, J., M. Zhou and D. Yang (2007). *Extracting Chatbot Knowledge from Online Discussion Forums*. IJCAI.

Huij, J., Lansdorp, S., Blitz, D., & van Vliet, P. (2014). Factor investing: Long-only versus long-short. Available at SSRN 2417221.

Jamshidian, F. (1997). LIBOR and swap market models and measures. *Finance and Stochastics*, *1*(4), 293-330.

Z. Jiang, D. Xu, and J. Liang, "A deep reinforcement learning framework for the financial portfolio management problem," arXiv preprint arXiv:1706.10059, 2017.

Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, *32*(3), 241-254.

Kahn, R. N. (2018). The Future of Investment Management, *CFA Institute Research Foundation.*

Kahn, R. N. and M. Lemmon (2016). "The Asset Manager's Dilemma: How Smart Beta is Disrupting the Investment Management Industry." *Financial Analysts Journal* **72**(1): 15-20.

Karypis, G., Han, E. H. S., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, (8), 68-75.

Kaski, S. (1997). Data exploration using self-organizing maps. In *Acta polytechnica scandinavica: Mathematics, computing and management in engineering series no. 82*.

D. Kinn, "Reducing Estimation Risk in Mean-Variance Portfolios with Machine Learning," arXiv preprint arXiv:1804.01764, 2018.

Z. Kakushadze and W. Yu, "Statistical industry classification," *Journal of Risk & Control*, vol. 3, no. 1, pp. 17-65, 2016.

Z. Kakushadze and W. Yu, "Statistical risk models," *The Journal of Investment Strategies*, vol. 6, no. 2, pp. 1-40, 2017.

Z. Kakushadze and W. Yu, "Machine Learning Risk Models," *Journal of Risk & Control*, vol. 6, no. 1, pp. 37-64, 2019.

L. Khaidem, S. Saha, and S. R. Dey, "Predicting the direction of stock market prices using random forest," arXiv preprint arXiv:1605.00003, 2016.

Kohonen, T. (1997, June). Exploration of very large databases by self-organizing maps. *In Proceedings of International Conference on Neural Networks* (ICNN'97) (Vol. 1, pp. PL1-PL6). IEEE.

R. K. Lindsay, B. G. Buchanan, E. A. Feigenbaum, and J. Lederberg, "DENDRAL: a case study of the first expert system for scientific hypothesis formation," *Artificial intelligence*, vol. 61, no. 2, pp. 209-261, 1993.

Landsberg, R. D. (2013). "A Macro-View of the New Definition of Fiduciary Under ERISA Proposed Sec.3 (21)." *Journal of Personal Finance* **12**(1).

L. Le, E. Ferrara, and A. Flammini, "On predictability of rare events leveraging social media: a machine learning perspective," in Proceedings of the 2015 *ACM on Conference on Online Social Networks*, 2015, pp. 3-13: ACM.

Levy, H., & Sarnat, M. (1970). International diversification of investment portfolios. *The American Economic Review*, *60*(4), 668-675.

J. Lintner, "The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets," in *Stochastic Optimization Models in Finance: Elsevier*, 1975, pp. 131-155.

Z. Liang, K. Jiang, H. Chen, J. Zhu, and Y. Li, "Deep Reinforcement Learning in Portfolio Management," arXiv preprint arXiv:1808.09940, 2018.

Liao, T. W. (2005). Clustering of time series data—a survey. *Pattern recognition*, *38*(11), 1857-1874.

Likas, A., Vlassis, N., & Verbeek, J. J. (2003). The global k-means clustering algorithm. *Pattern recognition*, *36*(2), 451-461.

Lintner, J. (1975). The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolios and Capital Budgets. Stochastic Optimization Models in Finance, *Elsevier:* 131-155.

W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11-26, 2017/04/19/ 2017.

Lopez de Prado, M. (2016). Mathematics and economics: a reality check. *Journal of Portfolio Management*, 43(1).

M. Lopez de Prado, "Building diversified portfolios that outperform out-of-sample," *Journal of Portfolio Management*, 2016

M. Lopez de Prado and M. J. Lewis, "Detection of False Investment Strategies Using Unsupervised Learning Methods," Available at SSRN: https://ssrn.com/abstract=3167017 or http://dx.doi.org/10.2139/ssrn.3167017, 2018.

H. Markowitz, "Portfolio selection," *The journal of finance*, vol. 7, no. 1, pp. 77-91, 1952.

McCulloch, W. S. and W. Pitts (1943). "A Logical Calculus of the Ideas Immanent in Nervous Activity." *The Bulletin of Mathematical Biophysics* **5**(4): 115-133.

D. Modha and C. Witchalls, "A computer that thinks," vol. 224, ed: *New Scientist* Ltd., 2014, pp. 28-29.

B. G. Malkiel and E. F. Fama, "Efficient capital markets: A review of theory and empirical work," *The journal of Finance*, vol. 25, no. 2, pp. 383-417, 1970.

Malkiel, B. G. and E. F. Fama (1970). "Efficient Capital Markets: A Review of Theory and Empirical Work." *Journal of Finance* **25**(2): 383-417

Modha, D. and C. Witchalls (2014). A Computer that Thinks, *New Scientist* Ltd. **224:** 28-29.

Modha, D. S., R. Ananthanarayanan, S. K. Esser, A. Ndirango, A. J. Sherbondy and R. Singh (2011). "Cognitive Computing." *Communications of the ACM* **54**(8): 62-71.

J. Mossin, "Equilibrium in a capital asset market," Econometrica: *Journal of the econometric society*, pp. 768-783, 1966.

K. Nakagawa, T. Uchida, and T. Aoshima, "Deep Factor Model," arXiv preprint arXiv:1810.01278, 2018.

K. Nakagawa, T. Uchida, and T. Aoshima, "Deep factor model," in ECML PKDD 2018 Workshops, 2018, pp. 37-50: Springer.

K. Nakagawa, T. Ito, M. Abe, and K. Izumi, "Deep Recurrent Factor Model: Interpretable Non-Linear and Time-Varying Multi-Factor Model," in AAAI-19 *Workshop on Network Interpretability for Deep Learning*, Honolulu, Hawaii, USA, 2019: arXiv preprint arXiv:1901.11493.

P. Nousi et al., "Machine learning for forecasting mid-price movement using limit order book data," arXiv preprint arXiv:1809.07861, 2018.

Oates, T., Firoiu, L., & Cohen, P. R. (1999, August). Clustering time series with hidden markov models and dynamic time warping. In *Proceedings of the IJCAI-99 workshop on neural, symbolic and reinforcement learning methods for sequence learning* (pp. 17-21). Sweden Stockholm.

N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Temporal Logistic Neural Bag-of-Features for Financial Time series Forecasting leveraging Limit Order Book Data," arXiv preprint arXiv:1901.08280, 2019.

Pástor, Ľ., & Stambaugh, R. F. (2003). Liquidity risk and expected stock returns. *Journal of Political economy*, 111(3), 642-685.

P. F. Procacci and T. Aste, "Forecasting market states," Available at SSRN: https://ssrn.com/abstract=3215945 or http://dx.doi.org/10.2139/ssrn.3215945, 2018.

T. Raffinot, "Hierarchical clustering-based asset allocation," Available at SSRN 2840729, 2017.

R. Rana and F. S. Oliveira, "Dynamic pricing policies for interdependent perishable products or services using reinforcement learning," *Expert Systems with Applications*, vol. 42, no. 1, pp. 426-436, 2015.

Rasmussen, C. E. (2000). The infinite Gaussian mixture model. In *Advances in neural information processing systems* (pp. 554-560).

Recht, B. and A. Rahimi (2017). Reflections on Random Kitchen Sinks, 2017.

Ren, R., D. D. Wu and T. Liu (2018). "Forecasting Stock Market Movement Direction Using Sentiment Analysis and Support Vector Machine." *IEEE Systems Journal*.

H. Reynolds, "AI? Or cognitive computing?," *KM World*, Article vol. 26, no. 9, pp. 4-5, 2017.

J.-C. Richard and T. Roncalli, "Constrained Risk Budgeting Portfolios: Theory, Algorithms, Applications & Puzzles," arXiv preprint arXiv:1902.05710, 2019.

G. Ritter, "Machine learning for trading," 2017.

B. Rosenberg, "Extra-market components of covariance in security returns," *Journal of Financial and Quantitative Analysis*, vol. 9, no. 2, pp. 263-274, 1974.

Ross, S. A. (2013). The Arbitrage Theory of Capital Asset Pricing. H*andbook of the Fundamentals of Financial Decision Making: Part I, World Scientific***:** 11-30.

S. Russell and P. Norvig, Artificial Intelligence A Modern Approach Third Edition. *Prentice Hall*, 2009.

Y.-L. K. Samo and A. Vervuurt, "Stochastic Portfolio Theory: a machine learning perspective," presented at the *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, Jersey City, New Jersey, USA, 2016.

M. Sardelich and S. Manandhar, "Multimodal deep learning for short-term stock volatility prediction," arXiv preprint arXiv:1812.10479, 2018.

D. Sculley, J. Snoek, A. Wiltschko, and A. Rahimi, "Winner's Curse? On Pace, Progress, and Empirical Rigor," 2018.

Sebastiani, F. (2002). "Machine Learning in Automated Text Categorization." *ACM Computing Surveys (CSUR)* **34**(1): 1-47.

D. Shah, H. Isah, and F. Zulkernine, "Predicting the Effects of News Sentiments on the Stock Market," in *2018 IEEE International Conference on Big Data* (Big Data), 2018, pp. 4705-4708: IEEE.

Sharma, A., & Dey, S. (2012, October). A comparative study of feature selection and machine learning techniques for sentiment analysis. In *Proceedings of the 2012 ACM research in applied computation symposium* (pp. 1-7). ACM.

W. F. Sharpe, "Capital asset prices: A theory of market equilibrium under conditions of risk," The journal of finance, vol. 19, no. 3, pp. 425-442, 1964.

Sharpe, W. F. (1991). "The Arithmetic of Active Management." Financial Analysts Journal: 7-9.

Sherwood, C. R., Jay, D. A., Harvey, R. B., Hamilton, P., & Simenstad, C. A. (1990). Historical changes in the Columbia River estuary. *Progress in Oceanography*, 25(1-4), 299-352.

Y.-G. Song, Y.-L. Zhou, and R.-J. Han, "Neural networks for stock price prediction," arXiv preprint arXiv:1805.11317, 2018.

J. De Spiegeleer, D. B. Madan, S. Reyners, and W. Schoutens, "Machine learning for quantitative finance: fast derivative pricing, hedging and fitting," Quantitative Finance, pp. 1-9, 2018.

E. Taghizadeh, "Utilizing artificial neural networks to predict demand for weather-sensitive products at retail stores," arXiv preprint arXiv:1711.08325, 2017.

A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Forecasting Stock Prices from the Limit Order Book Using Convolutional Neural Networks," in 2017 IEEE 19th Conference on Business Informatics (CBI), 2017, vol. 01, pp. 7-12.

W. P. Wagner, "Trends in expert system development: A longitudinal content analysis of over thirty years of expert system case studies," Expert systems with applications, vol. 76, pp. 85-96, 2017.

Y. Wang and X. S. Ni, "A XGBoost risk model via feature selection and Bayesian hyper-parameter optimization," International Journal of Database Management Systems (IJDMS) preprint arXiv:1901.08433, 2019.

T. Wiecki, A. Campbell, J. Lent, and J. Stauth, "All That Glitters Is Not Gold: Comparing Backtest and Out-of-Sample Performance on a Large Cohort of Trading Algorithms," The Journal of Investing, vol. 25, no. 3, pp. 69-80, 2016.

I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, Data Mining: Practical machine learning tools and techniques. *Morgan Kaufmann,* 2016.

M. Zhang, Z. Luo, and H. Lu, "Latent Dirichlet Allocation with Residual Convolutional Neural Network Applied in Evaluating Credibility of Chinese Listed Companies," arXiv preprint arXiv:1811.11017, 2018.

X. Zhou, J. Wang, X. Yang, B. Lev, Y. Tu, and S. Wang, "Portfolio selection under different attitudes in fuzzy environment," *Information Sciences*, vol. 462, pp. 278-289, 2018.

Appendix I Published Literature Review

Trends and Applications of Machine Learning in Quantitative Finance

Sophie Emerson Bsc, Ruairi Kennedy Bsc, and Luke O'Shea Bsc, Dr John O'Brien

***Abstract* — Recent advances in machine learning are finding commercial applications across many industries, not least the finance industry. This paper focuses on applications in on core function of the finance, the**

investment process. This function includes return forecasting, risk modelling and portfolio construction. The study evaluates the current state of the art through an extensive reviewed of recent literature. Themes and technologies are identified and classified, and the key use cases highlighted. Quantitative investing, traditionally a leading in adopting new techniques is found to be the most common source of use cases in the emerging literature.

***Index Terms*—Machine Learning, Quantitative Finance, Portfolio Construction, Return Forecasting**

INTRODUCTION

Machine Learning (ML) is a subfield of Artificial Intelligence (AI) that uses statistical techniques that provide computer models with the ability to learn from a dataset, allowing the models to perform specific tasks without explicit programming [1]. ML is being applied to improve function across the finance industry in a wide range of areas including, for example, fraud detection, payment processing and regulation. This research evaluates current and potential applications of machine learning to the investment process. In particular, this includes the development of ML applications for return forecasting, portfolio construction and risk modelling.

The first widespread commercial use cases of artificial intelligence were "expert systems", originating in Stanford in the 1960s [2] and popularized in the 1980s and 1990s. Expert systems were designed to solve complex problems in a specific field, in a manner similar to a subject matter expert. Original expert systems were rule-based programmes developed in languages such as LISP and Prolog. In recent years, there has been a significant drop in interest in classic expert systems, as they are superseded by systems incorporating artificial intelligence [3]. AI systems are systems that replicate human thought processes. [4]. Many of these systems are advertised today as cognitive computing systems.

Manuscript received April 19, 2019. This work was supported by State Street Corporation and the authors wish to thank State Street for this support.

Luke O'Shea (davidluke.oshea@gmail.com), Sophie Emerson (sophieemerson@gmail.com), and Ruairi Kennedy (r.ocinneide@umail.ucc.ie) are researchers in the State Street Advanced Technology Centre, Cork University Business School, UCC, Ireland.

John O'Brien (j.obrien@ucc.ie) is a lecturer in the Department of Accounting & Finance, Cork University Business School, UCC, Ireland.

Cognitive computing describes a computer system which mimics human conative process in some way, cognitive processes are those that allow individuals to remember, think, learn and adapt [5]. The term has gained recognition in the public domain in recent years, due in large to the introduction of Watson, IBM's cognitive computing system. These systems are constructed by combining computer science with statistical and ML techniques developed over the last century [1]. Watson, in its original form, was a question answering computing system, responding to questions posed in natural language. It was introduced on the television quiz show "Jeopardy!" – where it defeated two of the show's most celebrated contestants in the "IBM Challenge" [6]. Large-scale systems such as Watson combine many techniques [6] to provide "augmented human intelligence" services to users [7]. However, the use of individual techniques, for example deep learning neural networks or reinforcement learning, has found significant success across industry and applications [8-10].

Recently, there has been a proliferation of ML techniques and growing interest in their applications in finance, where they have been applied to sentiment analysis of news, trend analysis, portfolio optimization, risk modelling among many use cases supporting investment management. This paper explores the potential of ML to enhance the investment process. We begin with a broad survey of the area to determine the main programming languages, frameworks and use cases for ML from the perspective of the financial industry. We then focus on cognitive systems and ML, along with their potential applications to quantitative investment. We look at research that has applied ML to the investment process, analyzing the technologies used, the functions of the applications and evidence of potential to improve investment outcomes. Our findings are relevant to both academics and practitioners with interest in investment management, and in particular quantitative investment, by providing a detailed discussion of the latest technologies, their potential uses and probability of successful application.

The paper is organized as follows. In Section II, we provide an overview of the development of the area as a background for the discussion, this includes the emergence of ML, common algorithms and methodologies, and a review of the evolution and theory of quantitative investing We then describe the research methods in Section III. Section IV provides a detailed description of the current state of the art in the application of ML to investment. We conclude with a discussion of the evidence presented in Section

BACKGROUND

Machine Learning

Although variations of ML have long been around, the discipline has developed rapidly in recent years. Many factors have combined to derive this development. Increased computer power has made real time processing feasible for many complex tasks, increase connectivity has driven innovation and automation in the delivery of traditional tasks and services, the potential to extract useful information from the vast amounts of data generated via the internet (Big Data) has led to novel analytic methods. Alongside this, the development of easy to use programming languages, such as Python and R, and ML focused frameworks such as TensorFlow, has contributed to the wide investigation of ML applications in industry. It has already found commercial application across multiple industries from automated trading systems in the finance industry to the health sector where ML algorithms assist decision making in fertility treatments [11]. The success of these applications is driving commercial research into further applications.

Common ML Approaches and Algorithms

Three main approaches to training ML algorithms are recognized; supervised learning, unsupervised learning and reinforcement learning. Supervised learning generates a function that maps inputs to outputs based on a set of training data. The algorithm infers a function linking each set of inputs with the expected, or labeled, output in the training set.
Unsupervised learning finds hidden patterns in and draws inferences from unlabeled data. Unsupervised learning provides on inputs to models, but does not specify an expected set of outcomes, the outcomes are unlabeled. Reinforcement learning enables algorithms to learn by trial and error, based feedback from past experiences. Like unsupervised learning, it does not require labeled data. A hybrid system, semi-supervised learning, combines supervised and unsupervised learning, using both labeled and unlabeled data to train models. This is useful where there is limited data or the process labeling data could introduce biases.

The main research areas in supervised learning are regression and classification (specifying the category or class to which something belongs), this approach is often used in developing predictive models. Regression techniques predict continuous responses using algorithms such as linear regression, decision trees and Artificial Neural Networks (ANNs). Classification techniques predict discrete responses using algorithms such as logistic regression, Support Vector Machines (SVMs) or K-Nearest Neighbors (KNN). The main research area in unsupervised learning is clustering. Clustering refers to grouping objects together, such that objects that are put in the same group are more similar to each other than objects in other groups.

Artificial neural networks have become a key technology in the development of ML. They were first proposed over 75 years ago, inspired by the workings of the human brain [12]. They are a collection of algorithms replicate the process of a biological brain at the neuron level [1].

There are a number of different classes of artificial neural networks, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and recursive neural networks, among others. CNNs are ideal for things such as image classification and video processing because they're able to identify patterns by focusing on fragments of images. RNNs are better for dealing with things like speech or text analysis because they use time-series information, such as monthly stock price figures to predict next month's figure. SVMs, used for classification and regression analysis, involve finding a hyperplane which minimizes the distance between a set of data points in an n-dimensional space. Bayesian networks are built from probability distributions and use probability laws for prediction and anomaly detection. KNN selects the most similar data points in the training data, this allows the algorithm to classify future data inputs in the same way. GANs have garnered much interest in recent years since they first introduced in 2014 [13]. GANs are comprised of two neural networks that compete against each other. One neural network generates data similar to the training dataset, and the other tries to evaluate whether data is from the training dataset or generated by the generative network. Some techniques are better suited to particular tasks than others. This research partly seeks to contribute to this area of knowledge. It is important to evaluate the effectiveness of

certain algorithms, to assist in choosing appropriate algorithms for specific tasks in future applications and studies.

The Evolution of Quantitative Investing

Graham and Dodd's *Security Analysis*, published in 1934 following the Wall Street Crash of 1929 is the seminal work on fundamental investing and remains in publication [14]. It is one of the first books to distinguish investing from speculation, advocating the use of a systematic framework for analyzing securities for stock selection.

A systematic approach to portfolio construction and risk analysis was presented in *Portfolio Selection* [15], published in 1952. In this, Markowitz provides a mathematical definition of risk as the standard deviation of return. The approach focused on maximizing portfolio performance by optimizing the trade-off between risk and return. This was the foundation of modern portfolio theory, providing an analytical framework for the construction and analysis of investment portfolios [16] [17].

A quantitative approach to market analysis gained popularity as advances in computing technology made the collection and analysis of large amounts of market data possible. This allowed the development and verification of market models on a scale not previously possible, contributing to significant advances in the understanding of financial markets, including the Capital Asset Pricing Model (CAPM) [18-21] and Efficient Market Hypothesis (EMH) [22].

In 1973, Fama and MacBeth used the Center for Research in Security Prices (CRSP) financial dataset (one of the first of its kind) to perform an empirical analysis of the CAPM [23]. They showed that the CAPM provided a good quantitative approximation of the behaviour of security prices while setting a standard for empirical cross-sectional analysis of market data [23].

The empirical support for the EMH, enhanced by the success of market indices, such as the S&P 500, led to the dominant view, particularly in academia, that active investing was futile, as it was impossible to beat a passive investment. In comprehensive literature reviews, [16] and [17] provide evidence that research and empirical evidence that challenged the CAPM and EMH was strongly discouraged. At the same time many examples of research that argued that although difficult, it is possible for active management to beat passive management, by exploiting market inefficiencies not covered by the CAPM and EMH. Strategies based on risk factor models, first explored by Rosenberg [24] and Ross [25] in the 1970s, surged in popularity [26] after the publication of the Fama-French three-factor model [27].

From Markowitz portfolio optimization to CAPM, EMH and factor models more recently, quantitative investors have shown that they are willing to embrace new techniques and strategies. A key argument for applying ML techniques to financial problems is that ML methods capture non-linear relationships [28] in the data. Non-linear methods required to model data where outputs are not directly proportional to the inputs [29] and many tradition analysis methods assume linear relations or non-linear models that can be simplified to linear models. Typical examples of well-established non-linear ML methods include SVM, KNN, and ANN [20].

ML has been applied with positive results across many areas of quantitative investing, including portfolio optimization [30, 31], factor investing [32], bond risk predictability [29], derivative pricing, hedging and fitting [33], back-testing [34]. The results section contains a comprehensive summary of papers where ML techniques are applied to areas of quantitative finance.

METHODOLOGY

Initially, a broad search was conducted to identify the major themes related to ML. This search yielded information on the popular use cases and technologies. This information informed a second, more focused investigation of relevant material. Here, the aim was to draw connections between popular use cases in finance and current ML techniques.

As quality and scope of published research can vary widely, measures were taken to reduce the possibility of including unreliable information in the final dataset. Before inclusion in the concept matrix, each paper was assessed on quality. This was achieved by using a variety of quality indicators including; the citation count, the quality of an institute's research

activities associated with the paper, bias created from funding sources, and the impact factor of the journal.

An appropriate search strategy was devised and carried out based on the main topics that were identified during the first investigation of the literature. The arXiv and SSRN databases were searched to ensure that the most up to date research papers is included. However, as these are not peer-reviewed papers, extra care was taken to ensure that the papers were from reputable authors, focusing on the quality of authors' previous publications. The topic phrases used in search were "portfolio management", "stock market forecasting", and "risk management". All of these topic phrases were used in conjunction with the key phrase "machine learning" in an attempt to return only relevant research papers. The purpose of searching by topic was to identify which technologies are widely and effectively used within each area. As we are evaluating the current state of the art, we wanted to ensure that only recent papers were included. Thus, we only included papers that were submitted in 2015 or later. From the initial search we collected a total of 118 papers. After an initial review of abstracts, papers that were not relevant to machine learning in finance (specifically investing) were removed. Any papers that were duplicates under more than one search topic were kept under the topic that appeared most relevant. Papers were then assessed in relation to their quality using the quality indicators mentioned above. This reduced the number of papers to 55.

RESULTS

Popular Machine Learning Use Cases and Algorithms

A concept-centric matrix was utilized initially to identify which areas commonly use machine learning techniques. Recurring concepts and themes were noted and counted across a sample of 67 papers identified. An initial list of recurring themes was identified and analyzed. Some themes, such as 'Geopolitics' were removed as they were deemed irrelevant due to the lack of research on the topic. A list of the most recurring themes with relevance to ML is presented in Table I.

TABLE I: RECURRING THEMES FROM THE LITERATURE REVIEW.

Theme	References
Return Forecasting	21
Portfolio Construction	12
Ethics	8
Fraud Detection	8
Decision Making	8
Language Processing	7
Sentiment analysis	7

The most common use-cases of identified were return forecasting and portfolio construction. Quantitative methods were introduced to finance through the equity market and it is unsurprising that it should lead the way in incorporating the latest advances in its processes. A large number of the papers above also discussed risk modelling. This led us to take return forecasting, portfolio construction, and risk modelling as our three core topics. The most popular ML techniques presented in the papers researched are presented in Table II as well as a breakdown of the different acronyms used in the table.

TABLE II: POPULAR TECHNIQUES FEATURED IN MACHINE LEARNING AND FINANCE PAPERS

	MLP	**SVM**	**LSTM**	**GRU**	**RNN**	**CNN**	**RF**	**GPR**	**LR**
Return Forecasting	7	5	4	2	-	1	2	-	-
Portfolio Construction	7	2	3	1	1	1	4	2	1
Risk Modelling	6	2	2	1	1	1	4	3	4

MLP	Multilayer Perceptron
SVM	Support Vector Machine
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
RNN	Recurrent Neural Network (basic)
CNN	Convolutional Neural Network
RF	Random Forests/Decision Trees
GPR	Gaussian Process Regression
LR	Logistic Regression

Many techniques used in the papers only appear once, some twice. Since the purpose of this paper is to identify the most popular machine learning techniques used in finance, specifically in the topics above, only techniques which appeared in at least three papers were included in the table. We also decided to include RNN, although it is only mentioned explicitly in two papers, it appears implicitly more frequently as both LSTM and GRU are subsets of the technology. The results of the analysis are presented in Table II.

Artificial neural networks are used in all three areas of finance studied, with a standard feedforward network (multilayer perceptron) being the most common. Useful results are found from networks that range from small to very large networks (deep neural networks). There is also evidence of preferences for some techniques in particular areas. Gaussian process regression is used in both portfolio construction and risk modelling but has not been applied to return forecasting.

Summary of Key Insights from Recent Papers

The paper selection included ML papers published in recent years as well as papers yet to be published by established authors from reputable institutions. These papers have been submitted for publication and are awaiting acceptance. Including the most recent studies in this field was done to help evaluate the cutting edge and state of the art of the use of ML for financial applications.

Portfolio Construction

Portfolio construction is the process of combining return forecasts and risk models to create an optimum portfolio given an investor's constraints. A variety of ANN methodologies are applied to the portfolio optimization problem, often outperforming traditional optimization techniques. Deep learning reappeared a number of times during this search in the context of portfolio construction. Deep learning refers to models that consist of multiple layers or stages of nonlinear information processing (for example, a neural network with many hidden layers) [35]. Both hierarchical clustering and reinforcement learning were used to improve portfolio diversification. Multiple papers discuss the method of applying Markov models to predict the performance of stocks. Markov models are a type of ML method that model variables that change randomly through time. The complicated nature of the global market makes using this type of model a viable option.

- The authors present a deep learning framework for portfolio design, applying their framework to the stocks in the IBB index, demonstrating that their portfolio weighted using deep learning outperformed the index [31].
- The author outlines a reinforcement learning solution for a rational risk-averse investor seeking to maximize expected utility of final wealth, giving an example of a Q-learning agent exploiting an approximate arbitrage in a simulation [36].
- The authors of both papers make use of hierarchical clustering algorithms for constructing diversified portfolios. The portfolios are constructed using variations of risk parity [30] and equal risk contribution methods [37] which take the hierarchical correlation structure of the assets into account. The portfolios constructed are shown to have superior diversification and out-of-sample risk adjusted performance.
- The authors make use of convex analysis techniques to devise an optimal portfolio coupled with a Hidden Markov Model (HMM) used to estimate growth rates in the market model, which achieves improved results over a simple model using geometric Brownian motions [38].
- The authors provide an overview of the financial applications of Gaussian processes and Bayesian optimization, providing examples for forecasting the yield curve with Gaussian processes, and using Bayesian optimization to build an online trend-following portfolio optimization strategy [39].
- The authors compare the use of Feature Salient Hidden Markov Models (FSHMM) and HMM for constructing factor investing portfolios. The FSHMM selects relevant factors for use from a pool of available factors, while the HMM uses the whole pool of factors. Both models outperformed benchmark

portfolios, with the FSHMM portfolio showing better performance [40]

- The authors use factors as inputs to deep neural network, SVM and random forest models for predicting stock returns. While their research again show the effectiveness of a deep learning model, more significantly they used Layer-wise Relevance Propagation (LRP) to determine individual factor contributions to the neural network's prediction [41].
- The authors create a non-linear multi-factor model using LSTM to estimate the non-linear function. As in the previous paper the authors make use of LRP to identify which factors contribute to the model. The performance of the LSTM model is compared to the neural network model used in [32] and gives superior returns [42].
- The authors examine the use of three deep reinforcement learning algorithms, Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO) and Policy Gradient (PG), in managing a portfolio of assets in the Chinese stock market. They determine that training conditions used in game playing and robot control are unsuitable for use with portfolio management, finding that DDPG and PPO gave unsatisfying performance in the training process. They propose the use of adversarial training methods and employ a revised PG algorithm which outperforms a Uniform Constant Rebalanced Portfolio (UCRP) benchmark [43].
- The authors employ models constructed using Gaussian processes and Monte Carlo Markov Chains which learn optimal strategies from historical data, based on user-specified performance metrics (e.g. excess return to the market index, Sharpe ratio, etc.) This approach addresses the inverse problem of Stochastic Portfolio Theory – devising suitable investment strategies that meet the desired investment objective, when initially given a user-defined portfolio selection. The models outperform the benchmark in-sample and out-of-sample for absolute terms (returns) and also after adjusting for risk (Sharpe Ratio) [44].
- The author provides an ML framework for estimating optimal portfolio weights. They apply this framework using three ML methods – Ridge and Lasso regression, and two newly introduced methods; Principal Component regression, Spike and Slab regression. All methods outperform the mean-variance, minimum-variance, and equal weight portfolios. [45].
- The authors propose a framework for applying machine learning algorithms to distinguish "good stocks" from "bad stocks". The strategy was validated by testing its performance on the Chinese stock market [46].
- The authors propose a way to find the risk budgeting portfolio by using optimization algorithms to find a solution to the logarithmic barrier problem. They use algorithms such as cyclical coordinate descent, alternative direction method of multipliers [47].
- The authors present a financial-model-free reinforcement learning framework to as a solution to the portfolio management problem. The study tests the proposed framework with the following neural networks: CNN, a basic RNN and LSTM [48].

I. Return Forecasting

Return forecasting, predicting the investment return from an asset or asset class, is central to investment management and features highly in the literature. Many types of ANN are tested on their ability to forecast return. Deep neural networks, CNNs, LSTMs are all applied to the problem of stock forecasting. In one theme, the new ML technology is applied to improve forecasts made from traditional inputs, such as fundamental accounting data or technical indicators. A second approach uses ML to extract new inputs from, such as sentiment from new sources of data. Finally, authors predicting movement at the market level rather than individual security, for example using ML to identify states,

- The authors use a CNN strategy to analyze and detect price movement patterns in high-frequency limit order book data. Multilayer neural network methods and SVMs were also considered however they conclude the CNNs provide better performance for this task [49].
- The authors train a deep neural network on reported fundamental data from publicly traded companies (revenue, operating income, debt etc.). The model forecasts future fundamental data based on a trailing 5-years window. A value investing factor strategy based on forecasted fundamental data outperforms a traditional value factor investing approach with compounded annual return of 17.1% vs 14.4% for a standard factor model [50].
- The authors implement several ML algorithms to predict future price movements using limit order book data. They employ two feature learning methods: Autoencoders, and Bag of Features. They compare three different classifiers: SVM, a Single Hidden Layer Feedforward Neural Network (SLFNN), and an MLP. They test the performance of the classifiers with an anchored walk forward analysis, to determine if the models can capture temporal information, as well as a hold-out per stock method, to determine if the models can learn features that can be applied to previously unseen stocks. The results from the MLP are better than the other classifiers. However, the use of the Autoencoder and Bag of Features in combination with the MLP lead to fewer correct predictions [51].
- The authors introduce a novel Temporal Logistic Neural Bag-of-Features approach, that can be used to tackle the challenges that come with working with data of a high dimensionality, in this case high-frequency limit order book data [52].
- The authors create a simple buy-hold-sell strategy to predict direction of movement for 43 CME listed commodities and FX futures based on an ANN trained on a multitude of features for each instrument designed to capture co-movements and historical memory in the data. An average prediction accuracy of 42% is achieved across all instruments, with higher accuracies achieved for certain instruments [53].
- The authors use a random forest model to predict the direction of stock prices based on price information and a number of momentum indicators (Relative Strength Index, Moving Average Convergence Divergence, Stochastic Oscillator, Williams %R, On Balance Volume, and Price Rate of Change). The algorithm is shown to outperform existing algorithms found in the literature [54].
- The authors provide a sentiment analysis dictionary which they use to predict stock movements in the pharmaceutical market sector. With this model they achieve an accuracy of 70.59%. [55]
- The authors present a methodology to define, identify, classify and forecast market states. They use a Triangulated Maximally Filtered Graph network to filter information, and simple logistic regression for predicting market states. They compare five models, with a Gaussian Mixture Model as their baseline. All five models outperform the baseline in terms of risk/return significance [56].
- The authors compare five ANN models for forecasting stock prices: a standard neural network using back propagation, a Radial Basis Function (RBF), a General Regression Neural Network (GRNN), SVM Regression (SVMR), and Least Squares SVM Regression (LS-SVMR). However, they compare the models on just three stocks: Bank of China, Vanke A, and Kweichou Moutai. The standard neural network using back propagation outperforms all of the other models across all three stocks, in terms of both Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE). [57]
- The authors use 25 risk factors as inputs to ML stock returns prediction models. Results show that deep neural networks generally outperform shallow neural networks, and the best networks also outperform representative machine learning models [58].

- The author employs ANNs to predict product demand for weather sensitive products in Walmart stores around the time of major weather events [59].
- The authors implement a Gaussian Naïve Bayes Classifier for prediction based on sentiment analysis of Twitter data. The data used was obtained from Twitter and pertained to the 2014 FIFA world cup. Their framework obtained an accuracy and Area Under the curve of the Receiver Operating Characteristic (AUROC) of around 80% and an 8% marginal profit when tested [60].

II. Risk

Three different themes are identified under the broad heading of risk. The first attempts to employ ML to improve tradition measure of risk used in the mean variance framework. The second theme looks for companies at risk of default or bankruptcy, natural language processing, identifying words that indicate higher risk, is a key technology here. The final theme uses ML to develop hedging strategies. Some authors look at identifying what selection of ML methods is best for risk modelling problems.

- The authors use k-means clustering to construct risk models by clustering stock returns normalized and by standard deviation squared and adjusted by mean absolute deviation using a method proposed in [61]. They demonstrate that this ML approach outperforms statistical risk models [62] in quantitative trading applications [63].
- The authors present a framework for hedging a portfolio of derivatives in the presence of market frictions such as transaction costs, market impact, liquidity constraints or risk limits [64].
- The authors show how Gaussian Process Regression can assist in pricing and hedging a Guaranteed Minimum Withdrawal Benefit (GMWB) Variable Annuity with stochastic volatility and stochastic interest rate [65].
- The authors show that machine learning can be as effective as other existing algorithms at solving difficult hedging problems in moderate dimension. They use techniques such as a modified LSTM neural network to calculate their hedging strategies [66].
- The authors aim to explore the optimal model for business risk prediction. They attempt to do this using XGBoost, and by simultaneously examining feature selection methods and hyper-parameter optimization in the modeling procedure [67].
- The authors try to predict daily stock volatility using news and price data. Their model, which utilizes a Bidirectional Long Short-Term Memory (BiLSTM) neural network and stacked LSTM's, outperforms the well-known Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model in all sectors analyzed (financial, health care, etc) [68].
- The authors exploit a heterogeneous information network of 35,657 global firms to improve the predictive performance of predicting firms likely to be added to a blacklist. Blacklists are used to keep track of entities that have unacceptable problems, such as financial or environmental issues. Blacklists help keep portfolios profitable and "green". Their model consists of a simple multilayer perceptron with thirty hidden units [69].
- The authors estimate corporate credibility of Chinese companies using a CNN and natural language processing. They use Latent Dirichlet Allocation to summarize the text of news articles and use a CNN to extract the most important words from each topic. The CNN learns how news articles may reflect the credibility of a company though the wording of articles and word occurrence. They verify their model works by building a negative rating system and showing a correlation between their model's results and the negative rating [70].
- The authors compare different strategies for solving a variation of the multi-armed bandit problem. In their version of the problem, the learner can pull several arms simultaneously, or none at all. This could easily be applied to assist in investment

decisions. Out of the strategies compared, Bayes-UCB-4P and TS-4P perform the best [71].

- The authors compare several ML algorithms: Logistic Regression, K-Dimensional Tree (K-D Tree), SVM, Decision Trees, AdaBoost, ANN, and Gaussian Processes (GP) for forecasting business failures (corporate bankruptcy). Models are compared on datasets of manufacturing companies in Korea and Poland. All of the models are compared on their performance when combined with different dimensionality reduction techniques. The techniques used are: Principal Component Analysis (PCA), Linear Discriminate Analysis (LDA), Isometric Feature Mapping (ISOMAP), and Kernel PCA. On the Korean dataset, all models perform similarly. K-D Tree, SVM, and GP perform best over all of the dimensionality reduction methods used. On the Polish dataset, the linear regression model performs the best. Although having a lower accuracy than some of the other models, it is the best performing method when compared over other results such as precision, recall, F1 score, and AUC (Area Under Curve) [72].

DISCUSSION

Strategy Development & Analysis

The results of the literature search demonstrate that there is a wide range of ML techniques being successfully applied to many areas in the development of quantitative investing strategies, outperforming traditional benchmarks, previously used techniques and algorithms in many cases. Algorithms that assume a linear relationship between data can result in reduced accuracy. [28] highlights this issue in terms of many of the econometric models employed by finance academics and investment managers. The author argues for the use of more advanced mathematical models and ML techniques such as unsupervised learning that are capable of modelling complex non-linear relationships in financial systems.

Taking factor investing as an example of this, [73] and [74] make use of statistical algorithms to show that many factors discovered over the last number of years (particularly those found using empirical evidence) can be considered inaccurate or invalid. In the aptly named paper, *Taming the Factor Zoo*, a double selection LASSO ML method was used to analyze the contribution and usefulness of individual factors amongst the large number available today [75]. LASSO (Least Absolute Shrinkage and Selection Operator) is a regression analysis method capable of reducing the dimensionality of a large sample while selecting variables significant to the final result [76]. In [58] the author uses twenty-five factors as model inputs, comparing the use of shallow and deep neural networks, as well as SVMs and random forests for predicting stock returns, finding the deep neural networks (more layers) superior to the other methods. Using a similar approach [41] uses factors as inputs to deep neural network, SVM and random forest models for predicting stock returns. While their research again showed the effectiveness of a deep learning model, more significantly they used layer-wise relevance propagation to determine individual factors contributions to the neural network's prediction.

In these cases, not only has ML been used to develop investment strategies, but also to detect which input features were significant and which were not.

The use of Alternative Data

The use of ML for the analysis and application of alternative data for example, sentiment analysis, supply chain data etc. has opened up opportunities for new investment strategies. As seen in Table I, sentiment analysis was identified as a popular use case for ML. [17] provide a thorough overview of the growth of big data and sentiment analysis research over the last 30 years, highlighting the use of techniques such as NLP, SVMs and ANNs for the analysis of news, conference calls, reports, and social media activity. They concluded that, to date, sentiment information provided short-term, easy to exploit insights but long-term persistent insights as hard to achieve (falling in line with EMH). [16] acknowledges the effectiveness of big data for the modern fundamental investor, as it can provide insights and improve decision making by widening their research capabilities. This sentiment is echoed in [28] where the author makes reference to the recently emerged term "quantamental" – describing a fundamentally leaning investor who manages their portfolio

based on data-driven insights provided by ML algorithms. Examples of ML and alternative data being applied together in the results section mainly fall under return forecasting or risk modelling, where decisions may be made based on good or bad news [55], weather [59], or social media sentiment [60].

Choosing Machine Learning Algorithms

It is important to understand the relevant factors that contribute to the choice of ML algorithm, given the wide range available. These factors include accuracy, training time, linearity, number of parameters, the number of features and the structure of the data [77]. Some systems do not need to be a high level of accuracy. Estimates may be sufficient, for example, when calculating different route times for a journey. Model training times can also vary hugely between algorithms, making some algorithms more appealing than others when under time constraints. Many algorithms assume a linear relationship between input and output (linear regression, logistic regression, SVMs). This can result in reduced accuracy when dealing with non-linear problems. The number of parameters an algorithm has can indicate its flexibility, but also indicates that more time and effort may be required to find optimal values for training the model. The number of features can also be overwhelming for some algorithms. This is particularly a problem with textual data, where the number of words in the dictionary vastly outweighs the number of words in say, a paragraph being used for sentiment analysis. It's important to consider the structure of the data and the specific problem, as some algorithms are better suited for certain problems and data structures [78].

Backtesting & Strategy Verification

While ML techniques can provide superior performance financial data is notorious for having a low signal-to-noise ratio, which can lead to the detection of false patterns and results. Backtesting protocols have been proposed to tackle this [79]. ML solutions have also been applied to this problem. In [34] the authors present an unsupervised learning strategy which makes use of a modified k-means clustering algorithm to extract the number of uncorrelated trials from a series of backtests, which can be used in estimating the probability of false positives and estimating the expected value of the maximum Sharpe ratio. While in [80] the authors use a machine learning strategy for backtesting and the evaluation of automated trading strategies which is trained on a number of performance and risk metrics, demonstrating that this strategy outperforms standard metrics such as Sharpe ratio out-of-sample.

The development of new backtesting strategies and protocols is welcome and necessary, especially taking into account recent "black box" criticisms by leading deep learning researchers regarding a lack of testing and reproducibility in the field of ML. In their acceptance speech after winning the "test-of-time" award at NIPS, the leading AI conference, the authors of [81] compared much of recent ML research to "alchemy", highlighting a situation where algorithms were being created and trained using trial and error methods, with the researchers unable to explain the fundamental operation. They later published a paper highlighting instances of this [82].

CONCLUSION

As the previous section discusses, ML offers an opportunity for more complex financial analysis than was previously possible. The literature shows that quantitative investors have embraced new tools and techniques as they have emerged [16, 17].

There is a growing body of literature applying ML techniques to investment problems. Varieties of ML methods have been applied to areas of quantitative finance– the most popular methods are MLPs followed SVMs and LSTM. ML has been applied to problems in areas such as return forecasting, portfolio construction, and risk modelling.
These ML methods utilize traditional financial data, as well as making use of new types of alternative data. Big data is providing new datasets that need to be analyzed and ML techniques are capable of modeling complex (non-linear) relationships and analyzing new data.

[28] notes the recent trend of traditional hedge funds hiring an increasing proportion of STEM graduates for portfolio construction positions, as they possess the required mathematical

skillset for performing complex analysis and computer modelling. An understanding of machine learning, as well as the languages (Python, R, etc.) and frameworks (e.g. TensorFlow) needed to construct complex models could certainly be considered advantageous for any quantitative investor looking for an edge.

REFERENCES

[1] P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78-87, 2012.

[2] R. K. Lindsay, B. G. Buchanan, E. A. Feigenbaum, and J. Lederberg, "DENDRAL: a case study of the first expert system for scientific hypothesis formation," *Artificial intelligence*, vol. 61, no. 2, pp. 209-261, 1993.

[3] W. P. Wagner, "Trends in expert system development: A longitudinal content analysis of over thirty years of expert system case studies," *Expert systems with applications*, vol. 76, pp. 85-96, 2017.

[4] S. Russell and P. Norvig, *Artificial Intelligence A Modern Approach Third Edition*. Prentice Hall, 2009.

[5] D. Modha and C. Witchalls, "A computer that thinks," vol. 224, ed: New Scientist Ltd., 2014, pp. 28-29.

[6] D. Ferrucci *et al.*, "Building Watson: An overview of the DeepQA project," *AI magazine*, vol. 31, no. 3, pp. 59-79, 2010.

[7] H. Reynolds, "AI? Or cognitive computing?," *KM World*, Article vol. 26, no. 9, pp. 4-5, 2017.

[8] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11-26, 2017/04/19/ 2017.

[9] R. Rana and F. S. Oliveira, "Dynamic pricing policies for interdependent perishable products or services using reinforcement learning," *Expert Systems with Applications*, vol. 42, no. 1, pp. 426-436, 2015.

[10] G. Choy *et al.*, "Current applications and future impact of machine learning in radiology," *Radiology*, vol. 288, no. 2, pp. 318-328, 2018.

[11] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[12] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115-133, 1943.

[13] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672-2680.

[14] B. Graham and D. Dodd, *Security Analysis: Foreword by Warren Buffett*. McGraw-Hill Professional, 2008.

[15] H. Markowitz, "Portfolio selection," *The journal of finance*, vol. 7, no. 1, pp. 77-91, 1952.

[16] R. N. Kahn, *The Future of Investment Management*. CFA Institute Research Foundation, 2018.

[17] Y. L. Becker and M. R. Reinganum, *The Current State of Quantitative Equity Investing*. CFA Institute Research Foundation, 2018.

[18] W. F. Sharpe, "Capital asset prices: A theory of market equilibrium under conditions of risk," *The journal of finance*, vol. 19, no. 3, pp. 425-442, 1964.

[19] J. Mossin, "Equilibrium in a capital asset market," *Econometrica: Journal of the econometric society*, pp. 768-783, 1966.

[20] J. Lintner, "The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets," in *Stochastic Optimization Models in Finance*: Elsevier, 1975, pp. 131-155.

[21] C. W. French, "The Treynor capital asset pricing model," *Journal of Investment Management*, vol. 1, no. 2, pp. 60-72, 2003.

[22] B. G. Malkiel and E. F. Fama, "Efficient capital markets: A review of theory and empirical work," *The journal of Finance*, vol. 25, no. 2, pp. 383-417, 1970.

[23] E. F. Fama and J. D. MacBeth, "Risk, return, and equilibrium: Empirical tests," *Journal of political economy*, vol. 81, no. 3, pp. 607-636, 1973.

[24] B. Rosenberg, "Extra-market components of covariance in security returns," *Journal of Financial and Quantitative Analysis*, vol. 9, no. 2, pp. 263-274, 1974.

[25] S. A. Ross, "The arbitrage theory of capital asset pricing," in *HANDBOOK OF THE FUNDAMENTALS OF FINANCIAL DECISION MAKING: Part I*: World Scientific, 2013, pp. 11-30.

[26] J. H. Cochrane, "Presidential Address: Discount Rates," *The Journal of Finance*, vol. 66, no. 4, pp. 1047-1108, 2011.

[27] E. F. Fama and K. R. French, "Common risk factors in the returns on stocks and bonds," *Journal of financial economics*, vol. 33, no. 1, pp. 3-56, 1993.

[28] M. Lopez de Prado, "Mathematics and economics: a reality check," 2016.

[29] D. Bianchi, M. Büchner, and A. Tamoni, "Bond risk premia with machine learning," *Available at SSRN 3232721*, 2018.

[30] M. Lopez de Prado, "Building diversified portfolios that outperform out-of-sample," *Journal of Portfolio Management*, 2016.

[31] J. Heaton, N. Polson, and J. H. Witte, "Deep learning for finance: deep portfolios," *Applied Stochastic Models in Business and Industry*, vol. 33, no. 1, pp. 3-12, 2017.

[32] K. Nakagawa, T. Uchida, and T. Aoshima, "Deep Factor Model," *arXiv preprint arXiv:1810.01278*, 2018.

[33] J. De Spiegeleer, D. B. Madan, S. Reyners, and W. Schoutens, "Machine learning for quantitative finance: fast derivative pricing, hedging and fitting," *Quantitative Finance*, pp. 1-9, 2018.

[34] M. Lopez de Prado and M. J. Lewis, "Detection of False Investment Strategies Using Unsupervised Learning Methods," *Available at SSRN: https://ssrn.com/abstract=3167017 or http://dx.doi.org/10.2139/ssrn.3167017*, 2018.

[35] L. Deng and D. Yu, "Deep learning: methods and applications," *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197-387, 2014.

[36] G. Ritter, "Machine learning for trading," 2017.

[37] T. Raffinot, "Hierarchical clustering based asset allocation," *Available at SSRN 2840729*, 2017.

[38] A. Al-Aradi and S. Jaimungal, "Active and Passive Portfolio Management with Latent

Factors," *arXiv preprint arXiv:1903.06928*, 2019.
[39] J. Gonzalvez, E. Lezmi, T. Roncalli, and J. Xu, "Financial Applications of Gaussian Processes and Bayesian Optimization," *arXiv preprint arXiv:1903.04841*, 2019.
[40] E. Fons, P. Dawson, J. Yau, X.-j. Zeng, and J. Keane, "A novel dynamic asset allocation system using Feature Saliency Hidden Markov models for smart beta investing," *arXiv preprint arXiv:1902.10849*, 2019.
[41] K. Nakagawa, T. Uchida, and T. Aoshima, "Deep factor model," in *ECML PKDD 2018 Workshops*, 2018, pp. 37-50: Springer.
[42] K. Nakagawa, T. Ito, M. Abe, and K. Izumi, "Deep Recurrent Factor Model: Interpretable Non-Linear and Time-Varying Multi-Factor Model," in *AAAI-19 Workshop on Network Interpretability for Deep Learning*, Honolulu, Hawaii, USA, 2019: arXiv preprint arXiv:1901.11493.
[43] Z. Liang, K. Jiang, H. Chen, J. Zhu, and Y. Li, "Deep Reinforcement Learning in Portfolio Management," *arXiv preprint arXiv:1808.09940*, 2018.
[44] Y.-L. K. Samo and A. Vervuurt, "Stochastic Portfolio Theory: a machine learning perspective," presented at the Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence, Jersey City, New Jersey, USA, 2016.
[45] D. Kinn, "Reducing Estimation Risk in Mean-Variance Portfolios with Machine Learning," *arXiv preprint arXiv:1804.01764*, 2018.
[46] X. Zhou, J. Wang, X. Yang, B. Lev, Y. Tu, and S. Wang, "Portfolio selection under different attitudes in fuzzy environment," *Information Sciences*, vol. 462, pp. 278-289, 2018.
[47] J.-C. Richard and T. Roncalli, "Constrained Risk Budgeting Portfolios: Theory, Algorithms, Applications & Puzzles," *arXiv preprint arXiv:1902.05710*, 2019.
[48] Z. Jiang, D. Xu, and J. Liang, "A deep reinforcement learning framework for the financial portfolio management problem," *arXiv preprint arXiv:1706.10059*, 2017.
[49] A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Forecasting Stock Prices from the Limit Order Book Using Convolutional Neural Networks," in *2017 IEEE 19th Conference on Business Informatics (CBI)*, 2017, vol. 01, pp. 7-12.
[50] J. Alberg and Z. C. Lipton, "Improving Factor-Based Quantitative Investing by Forecasting Company Fundamentals," *arXiv preprint arXiv:1711.04837*, 2017.
[51] P. Nousi *et al.*, "Machine learning for forecasting mid price movement using limit order book data," *arXiv preprint arXiv:1809.07861*, 2018.
[52] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Temporal Logistic Neural Bag-of-Features for Financial Time series Forecasting leveraging Limit Order Book Data," *arXiv preprint arXiv:1901.08280*, 2019.
[53] M. Dixon, D. Klabjan, and J. H. Bang, "Classification-based financial markets prediction using deep neural networks," *Algorithmic Finance*, no. Preprint, pp. 1-11, 2017.
[54] L. Khaidem, S. Saha, and S. R. Dey, "Predicting the direction of stock market prices using random forest," *arXiv preprint arXiv:1605.00003*, 2016.
[55] D. Shah, H. Isah, and F. Zulkernine, "Predicting the Effects of News Sentiments on the Stock Market," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018, pp. 4705-4708: IEEE.
[56] P. F. Procacci and T. Aste, "Forecasting market states," *Available at SSRN: https://ssrn.com/abstract=3215945 or http://dx.doi.org/10.2139/ssrn.3215945*, 2018.
[57] Y.-G. Song, Y.-L. Zhou, and R.-J. Han, "Neural networks for stock price prediction," *arXiv preprint arXiv:1805.11317*, 2018.
[58] M. Abe and H. Nakayama, "Deep Learning for Forecasting Stock Returns in the Cross-Section," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2018, pp. 273-284: Springer.
[59] E. Taghizadeh, "Utilizing artificial neural networks to predict demand for weather-sensitive products at retail stores," *arXiv preprint arXiv:1711.08325*, 2017.
[60] L. Le, E. Ferrara, and A. Flammini, "On predictability of rare events leveraging social media: a machine learning perspective," in *Proceedings of the 2015 ACM on Conference on Online Social Networks*, 2015, pp. 3-13: ACM.
[61] Z. Kakushadze and W. Yu, "Statistical industry classification," *Journal of Risk & Control*, vol. 3, no. 1, pp. 17-65, 2016.
[62] Z. Kakushadze and W. Yu, "Statistical risk models," *The Journal of Investment Strategies*, vol. 6, no. 2, pp. 1-40, 2017.
[63] Z. Kakushadze and W. Yu, "Machine Learning Risk Models," *Journal of Risk & Control*, vol. 6, no. 1, pp. 37-64, 2019.
[64] H. Buehler, L. Gonon, J. Teichmann, and B. Wood, "Deep hedging," *Quantitative Finance*, pp. 1-21, 2019.
[65] L. Goudenège, A. Molent, and A. Zanette, "Gaussian Process Regression for Pricing Variable Annuities with Stochastic Volatility and Interest Rate," *arXiv preprint arXiv:1903.00369*, 2019.
[66] S. Fecamp, J. Mikael, and X. Warin, "Risk management with machine-learning-based algorithms," *arXiv preprint arXiv:1902.05287*, 2019.
[67] Y. Wang and X. S. Ni, "A XGBoost risk model via feature selection and Bayesian hyper-parameter optimization," *International Journal of Database Management Systems (IJDMS) preprint arXiv:1901.08433*, 2019.
[68] M. Sardelich and S. Manandhar, "Multimodal deep learning for short-term stock volatility prediction," *arXiv preprint arXiv:1812.10479*, 2018.
[69] R. Hisano, D. Sornette, and T. Mizuno, "Predicting Adverse Media Risk using a Heterogeneous Information Network," *arXiv preprint arXiv:1811.12166*, 2018.
[70] M. Zhang, Z. Luo, and H. Lu, "Latent Dirichlet Allocation with Residual Convolutional Neural Network Applied in Evaluating Credibility of Chinese Listed Companies," *arXiv preprint arXiv:1811.11017*, 2018.
[71] M. Achab, S. Clémençon, and A. Garivier, "Profitable Bandits," presented at the Proceedings of The 10th Asian Conference on Machine Learning, Proceedings of Machine Learning Research, 2018. Available: http://proceedings.mlr.press

[72] J. C. Chow, "Analysis of Financial Credit Risk Using Machine Learning," *arXiv preprint arXiv:1802.05326*, 2018.
[73] C. Harvey and Y. Liu, "Lucky factors," 2017.
[74] C. R. Harvey, Y. Liu, and H. Zhu, "… and the Cross-Section of Expected Returns," *The Review of Financial Studies*, vol. 29, no. 1, pp. 5-68, 2016.
[75] G. Feng, S. Giglio, and D. Xiu, "Taming the factor zoo," 2017.
[76] A. Belloni, V. Chernozhukov, and C. Hansen, "Inference on treatment effects after selection among high-dimensional controls," *The Review of Economic Studies*, vol. 81, no. 2, pp. 608-650, 2014.
[77] R. Barga, V. Fontama, and W. H. Tok, "Introducing Microsoft Azure Machine Learning," in *Predictive Analytics with Microsoft Azure Machine Learning*: Springer, 2015, pp. 21-43.
[78] P. Harrington, "Machine learning in action," *Shelter Island, NY: Manning Publications Co*, 2012.
[79] R. Arnott, C. R. Harvey, and H. Markowitz, "A Backtesting Protocol in the Era of Machine Learning," *The Journal of Financial Data Science*, vol. 1, no. 1, pp. 64-74, 2019.
[80] T. Wiecki, A. Campbell, J. Lent, and J. Stauth, "All That Glitters Is Not Gold: Comparing Backtest and Out-of-Sample Performance on a Large Cohort of Trading Algorithms," *The Journal of Investing*, vol. 25, no. 3, pp. 69-80, 2016.
[81] B. Recht and A. Rahimi, "Reflections on random kitchen sinks, 2017," ed, 2017.
[82] D. Sculley, J. Snoek, A. Wiltschko, and A. Rahimi, "Winner's Curse? On Pace, Progress, and Empirical Rigor," 2018.

Appendix II - Quant Award Essay Competition submission

Investor Regime Analysis using Self-Organising Maps and Hierarchical Clustering

---

Quant Awards 2019

Abstract

This study investigates how modern machine learning techniques can be used to cluster equity flows to define investor regimes and inform the creation of a hedge fund style investment model. Equity flow data is analysed using the artificial neural network technology and visualisation tool known as self-organising maps. This analysis informs the number of investor regimes to look for, the optimal number was found to be four. The Hierarchical clustering algorithm and dynamic-time warping distance measure are implemented to determine four investor regimes. These regimes are characterised by stability and average weekly returns. The results of which informed the creation of a portfolio model. The performance of the investment model is evaluated by comparing it to a risk-free rate. The portfolio is compared quantitatively to one created by repeating the methodology, relying solely on return data to inform the regimes in this iteration. The model created using equity flows outperformed the one made by clustering returns. performance was measured using the Sharpe ratio. The model created using equity flows was calculated to have a Sharpe ratio of 0.24 while the model created using the analysis of return data is shown to have a Sharpe ratio of -0.21. By examining the probability matrices of both models, we see that the regimes created by clustering equity flow data are shown to be more stable than the regimes created using return data.

Introduction

In recent decades, advanced statistical techniques and machine learning technologies have revolutionized how we process and analyse data (Domingos 2012). The term Machine Learning (ML) was coined by IBM in the 1970's, it refers to the field of study involving machines and computer programmes capable of performing useful tasks or gaining insight from data without being explicitly programmed to do so (Burkov 2019). The development of easy to use programming languages such as Python and R has contributed to the wide usage of ML algorithms. R is the primary tool used in this study to carry our data cleaning, analysis and the implementation of the machine learning algorithms. This study looks at how modern ML clustering and visualisation techniques can be used to cluster equity flows to define investor regimes. Equity flow data is clustered using hierarchical clustering and dynamic time warping. Intuitive explanations and definitions of these terms will be provided.

The word 'learning' can be deceiving as machines cannot learn the same ways in which humans do. The purpose of this catchy name was to encourage further research into this area, push the best talent to work for IBM and impress clients (Domingos 2012). These modern techniques have the potential to revolutionise the quantitative investing industry. This study discusses and tests

how ML assists in portfolio management in the modern day and the potential for financial applications in the future.

Portfolio managers have many tools and techniques at their disposal (Becker and Reinganum 2018, Kahn 2018). One of many effective strategies is the consideration of what "investor regime" the market resides in to assist in making investment decisions. An "investor regime" refers to a state the market is in, where equity is moving and what regions are exhibiting high, low or neutral returns (Ang, 2004).

This study relies on the established principle used by investors that if the market resides in a certain regime in each week then it is most likely to be in the same regime in the following week. These regimes inform the creation of a portfolio informed by investor regimes, the details of which are included. The same strategy is carried out again but this time clustering the regional return data to define regimes. We examine how profitable the resulting investment model is by comparing it to a risk-free rate. One can compare the use of equity flow data and return data when defining regimes and investigate the validity of using this modern technology to inform international portfolio management.

Self-organising maps and hierarchical clustering are used in tandem with the dynamic time warping distance measure are used on return and equity flow data to determine investor regimes and inform the creation of a profitable investment model. The results and methodology of this experiment are given in detail. We begin with a brief and intuitive explanation to the terms; Machine learning, supervised learning, unsupervised learning, self-organising maps, hierarchical clustering and dynamic time warping. Following on from that this study will describe how these techniques can be used to inform international portfolio diversification decisions.

Literature Review

Portfolio Management and Investor Regimes

Modern portfolio theory states that diversification of security returns with lower correlation should yield more favourable results for an investor (Levy & Sarnat, 1970). There exists a variety of different approaches that an investor can take to diversify a portfolio, including diversification by sector and by country or region. In a 1970 paper by Levy and Sarnat discusses the high degree of correlation between security returns in a single economy and presents the benefits of diversifying assets internationally in comparison to holding assets across different industries domestically (Levy & Sarnat, 1970). For many years international diversification has been an established portfolio management strategy and has grown more popular in recent decades (Hitt, 2006).

Equity Flows

Equity flows is the change in asset allocation across sector, countries or regions. Change of flows occur when investor move, buy or sell assets. It is established that they can be used to assist in forecasting future equity returns (Froot 2001). A 2001 study by Froot, O Connell and Seasholes showed their persistent nature, meaning that equity flows in general are less volatile and more reliable than returns (Froot 2001). This study uses machine learning techniques to analyse equity flow data to determine investor regimes and creating a portfolio that tracks investor behaviour in those regimes. We examine the hypothesis that equity flow data is more persistent/stable than that of equity return data by forming market regimes with each and creating transition/probability matrices of how the regimes change or stay in each regime from week to week.

Types of Learning

There exists a variety of different learning methods in which ML can be achieved. The three main types of learning will be outlined very briefly here. First, we consider the area of **Supervised learning**, where the dataset used in the algorithm must be labelled data (Burkov 2019). Each element of the data set can contain various facets of information. A single datapoint might refer to a person and each person might have several features such as gender, height, weight ect (Burkov 2019). In the context of this study, each data point is a week and the features of that data point are the equity flow values in that week. When implementing **unsupervised** learning, the dataset is a collection of unlabelled examples (Burkov 2019). **Reinforcement learning/ competitive learning** lies between supervised learning and unsupervised learning. It operates through continuing interactions between a learning system and the environment (Haykin, 2009), competitive learning is the learning type of main relevance in this study.

Self-Organising Maps

The first ML analysis tool used in this study is that of the **self-organising map**. A self-organising map (SOM), is a type of artificial neural network (Kohonen, 1990). They can reduce the dimensionality of data thus making them useful for visualization (Kohonen, 1990). Prof Teuvo Kohonen developed this data analysis technique in the 1980's (Kohonen, 1990). An established use for this technology is in the area of exploratory analysis, in examining the structure and finding patterns in large datasets (Kaski, 1997). An effective exploratory analysis tool is essential for analysts working on large and complicated data sets. Analysis of these data sets can sometimes be difficult and time-consuming (Kaski, 1997).

Like most ANN's, SOM's operate in two modes; training and mapping (Kohonen, 1990). The training part of the operation involves building the map using input examples. The mapping part of the operation automatically classifies a new input vector (Kohonen, 1990. This form of training is known as **competitive learning**. When a training example is fed to the network, its Euclidean distance to all weight vectors is computed. The Euclidean distance between two time series, V and W;

$$V = v_1, v_2, \ldots \ldots v_n \quad (1)$$
$$W = w_1, w_2, \ldots \ldots w_n \quad (2)$$

by the following formula

$$E(v, w) = \sum_{i=1}^{n} (v_i - w_i)^2 \quad (3)$$

The map space is pre-defined before the training process. The space consists of nodes arranged in a rectangular or hexagonal grid; the dimensions of this grid are pre-set. SOM's can be helpful in the area of data visualisation. Data science is more than just building machine learning models; it's also about explaining the models and using them to drive data-driven decisions. Displaying data in an informative and visually appealing way can play a very important role of presenting data in a powerful and credible way.

Data

Equity flow data

The primary data set used in this study is that of the equity flow data. This was provided by State Street Global markets. The data are derived from data held by State Street Bank & Trust (SSB). SSB the largest mutual fund custodian in the US and hold roughly 40% of the industry's funds under custody (Froot, 2001). An approximate estimate to the quantity of

assets under custody by SSB is $6 trillion. The period selected for analysis was from 7th of January 2012 to 18th of August 2018. The dataset was originally in daily format but was transformed to weekly in the statistical software; R. A breakdown of the regions can be found in Appendix I.

MSCI and LIBOR

The **MSCI Index** is a measurement of stock market performance in a particular area, it is the industry's accepted gauge of global stock market activity. The weekly MSCI data was downloaded from Bloomberg.com. The MSCI indices were used to calculate the total regional weekly returns.

The risk-free rate used in this study is the LIBOR dollar rate. It is the average interest rate at which leading banks borrow funds from other banks in the London market. It is a widely used global "benchmark" or reference rate for short term investments.

Methodology

The first task carried out during this study was to choose an appropriate number of investor regimes to look for. The exploratory analysis tool used was the SOM. Once the optimal number of regimes were determined, the weekly equity flow data was clustered using the hierarchical clustering algorithm and the dynamic time warping distance measure. Four investor regimes were determined and were characterised by average weekly returns and their stability. A portfolio model was constructed based on the analysis of these four regimes.

Number of regime selection by SOM

This helpful technique allows one to get an overall sense of a large dataset and quickly observe what the results look like when different numbers of regimes are chosen. By examining the resulting maps.

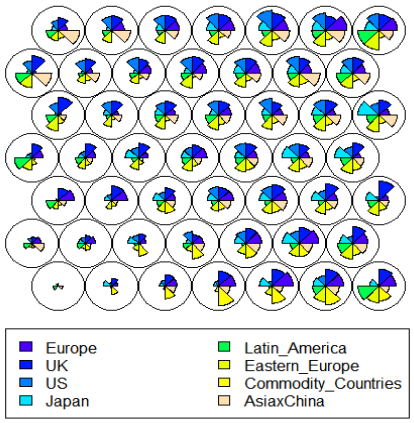


Figure 1: Example of 7x7 SOM, produced in R using equity flow data

In figure 1, This technique allows an analyst to get an overview of the dataset in question. Shown below is an example of a self-organising map produced using the total flow dataset, its smaller size makes it easier to read. Inside each node/circle there is a coloured wedge representing the magnitude of equity flow in a certain region. This magnitude can be compared by their relative size. For example, the top right corner of the SOM is populated by nodes representing weeks of high equity flow into all regions whereas in the bottom left corner, the size of the wedges are small which represent weeks of low or negative equity flow across the regions.

Determining Investor Regimes using Hierarchical Clustering and Dynamic-Time Warping

There exists a multitude of different **ML algorithms**, all possess individual purposes and advantages. The ML algorithm in focus here is that of **hierarchical clustering** (Steinbach, 2000). Hierarchical clustering either falls into the top-down or bottom-up category. The method used for these clusters was a 'bottom-up' method. Bottom-up algorithms treat each data point as a single cluster in the beginning and then merge (or agglomerate) pairs of clusters until all clusters have been merged into a single cluster that contains all data points

(Steinbach, 2000). Bottom-up hierarchical clustering is therefore called hierarchical agglomerative clustering or HAC.

Dynamic time warping (DTW) was the selected distance measure for this study. Once a clustering algorithm is chosen the next task is to select an appropriate distance measure. A **distance measure** defines what is the measure of similarity or dissimilarity between two data points. An example which highlights the importance of a distance measure is if you wished to write a program which calculated the time taken to get to a destination in a car. The most standard measure of similarity is that of the Euclidean distance. This measure, in terms of the car example, would calculate the 'birds' eye' view distance between two points on a map. In real life this is not a good measure of the distance a car must travel, as it must stay on roads and in some cases roads way have one-way systems.

Each distance measure has different specifications of what defines a cluster, so a certain clustering distance measures might be preferred depending on what types of clusters one wishes to obtain. Time-Series data can pose some challenges, partly due to factors like large size and dimensionality. A first important issue is to decide whether clustering must be governed by a "shape-based" or "structure-based" dissimilarity concept.

DTW is a 'shape-based' clustering algorithm (Berndt, Clifford 1994). It clusters together time series that have similar shapes. Consider time series S and T;

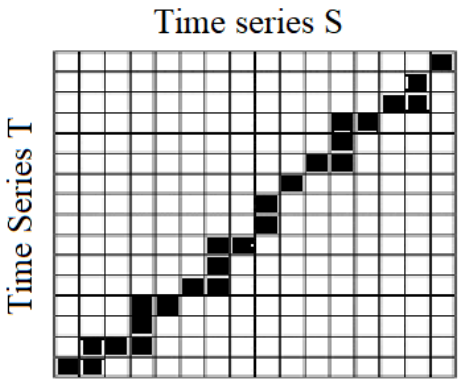
$$S = s_1, s_2, \ldots \ldots s_n \quad (4)$$
$$T = t_1, t_2, \ldots \ldots t_n \quad (5)$$

The DTW measure computes the difference between a point on S to every other point on T. It then iteratively does this to every point on S and creates a matrix out of these values. The **warping path** is the path taken to get from the lower left matrix entry up to the upper right matrix entry (Berndt, Clifford 1994). The DTW algorithms clusters together time series that are computed to have the smallest warping path, w, between them. This can be expressed as;

$$DTW(S,T) = min_W\left[\sum_{k=1}^{p} \partial(w_k)\right] \quad (6)$$

An example of a warping path between two time series is given below in figure 2.

Figure 2: Small example SOM, produced in R using equity flow data.



An advantage of dynamic time warping is that having dates out of sync across time series will not affect the results of clusters obtained. This can prove particularly useful when analysing international time-series data where time zones can cause differences in date/times of close of market (Berndt, Clifford 1994). Dynamic time warping simply considers the overall shape of the time series when measuring similarity. This can save time in data cleaning, getting date and times to match up exactly across many time-series can be time consuming and thus costly for companies (Berndt, Clifford 1994).

Exploratory Analysis using SOMs

The first step of the analysis process involved the equity flow dataset. Exploratory analysis was primarily carried out using SOMs, as discussed previously, this has been established as a useful tool for this stage of analysis. This technique allows an analyst to get an overview of the

dataset in question. Shown below is an example of a self-organising map produced using the total flow dataset, it's smaller size makes it easier to read.

This type of investment model relies on the principle that if the market is in a certain regime one week then it will most likely be in the same regime in the next week. This theory is examined with the use of probability matrices for each set of clusters found, we can see what the probability is for the market to change regime or to stay in the same regime.

The Sharpe ratio allows investors to compare the return of an investment to its risk. In general, the higher the Sharpe ratio, the more attractive the portfolio is to an investor. The recognized Sharpe ratio is

$$S = \frac{R_p - R_f}{\sigma_p} \qquad (7)$$

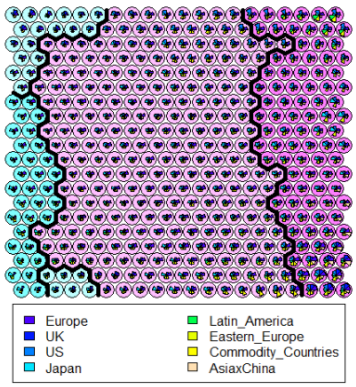
R_p is the return of the portfolio, R_f is the risk-free rate and σ_p is the standard deviation of the portfolio's excess annualised return.

Results and Discussion

Exploratory Analysis using SOMs

Shown overhead in figure 3, is a SOMs produced in Rstudio, using the kohonen package using the equity flow data. This analysis techniques allows one to get a general sense of the overall structure of the dataset. The SOM grid consists of many circular nodes, one can set the desired number of nodes depending on the size and nature of the dataset. The below grids consist of 400 nodes (20x20). Inside each node, there are eight wedges of varying size, each wedge representing the magnitude of equity flow. There is approximately 350 data points of average weekly equity flow from 2012 to the present day. The above SOM was trained with a dataset of similar magnitude to the number of nodes it possesses. For example, in the bottom right of both figures one can observe that each of these nodes correspond to weeks during this period where the equity flows across all regions are of a large positive magnitude. Each node roughly corresponds to a week during this period. A disadvantage to a SOM of this size is that is can cause the nodes to be difficult to read.

Figure 3: Full size SOM produced in R using equity flow data from 2012-2018.



Four distinct regions can be observed in figure 3. This was not the case in SOM produced where the number of clusters were set to be larger than four. This result informed the selection of four investor regimes moving forward in this research.

Characterising Regimes by Returns and Stability

Stability of Market Regimes

Probability matrices are also known as transition matrices, they display the probability of transitioning from one state to another. The following probability matrices illustrate the differences between using equity flow data to and return data to define market regimes. The two probability matrices shown in this section demonstrate the stability of the market regimes found created by first the equity flow data and secondly by utilising the return data.

Table 1: Probability matrix of market regimes created by equity data

	1	2	3	4
1	0.779	0.110	0.000	0.110
2	0.190	0.660	0.050	0.100
3	0.059	0.235	0.706	0.000
4	0.321	0.196	0.000	0.482

Table 2: Probability matrix of market regimes created by regional returns.

	1	2	3	4
1	0.152	0.261	0.326	0.261
2	0.126	0.336	0.263	0.274
3	0.139	0.278	0.306	0.278
4	0.115	0.229	0.364	0.292

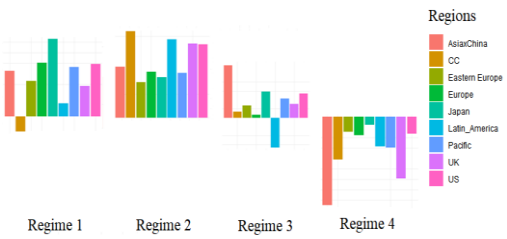
Each cell of the matrices represents the probability of the market changing to a certain regime or staying in the regime it is already in. The largest numbers in table 3 are those along the diagonal from the top left corner to the bottom right. This indicates that the regimes are relatively stable, if the market is in a certain regime then it is most likely to stay in that regime in the following week. This may be a contributing factor to the investment model created by equity flows outperforming the model created by return data.

This is not the case in table 4, where the numbers along the diagonal are not significantly larger than those in all other positions in the matrix. This shows the unstable nature of the market regimes created by the analysis of return data. This supports the hypothesis that equity data is more persistent and stable than equity return data.

Characterising Regimes by Average Weekly Returns

We examine the regimes obtained by clustering the total weekly equity flows from the period of January 2012 to July 2018. The average returns in each of the four regimes are found and inform the three investment models.

Figure 4: Return analysis of market regimes created by clustering regional equity flow



In the above plots, one can see that regime 1 and 2 are characterized by positive returns, regime 3 is generally low positive returns and returns are on average negative during regime 4 across all regions. The portfolio takes a long position in all regions except the commodity countries (neutral) and Latin America (short) during regime 1. In regime 2 the model takes a long position in all regions. In regime 3, the model takes a long position in Asia, Japan, Pacific, UK and US. It takes a neutral position in all other regions except in Latin America where it takes a short position. For regime 4, the model takes a short position in all regions. Larger versions of this plot and the return analysis comparison can be found in appendix IV.

Figure 5: Cumulative returns of hedge fund model informed by analysis of regional returns



Figure 5 above shows how the portfolio outperforming the risk-free rate. This proves the concept of the potential of this technology to create inform the creation of profitable portfolios. The details of this model can be found below in table 1.

Table 1: Hedge fund model created from equity flow data performance figures compared to the risk-free rate.

	Hedge Fund Model	Risk Free Rate
Total Returns	7.21%	4.46%
Annual Cumulative Returns	1.054%	0.660%
Volitility	1.64%	-
Sharpe ratio	0.24	0

Conclusions

The self-organising map technology was a useful and effective tool in the exploratory analysis of the equity flow data. By repeatedly using this technology and changing the number of clusters one can quickly and easily determine what is appropriate number of clusters to use when implementing the more sophisticated clustering algorithm of hierarchical clustering with dynamic time warping distance measure. Many SOMs were produced in order to determine the optimal number of regimes to look for. An example of one of these maps produced using the equity flow data is given in figure 3 where four distinct regions can be observed. Thus for the next portion of the study, four regimes were determined and analysed to inform the portfolio.

The results of this study showed that the portfolio model results in higher returns when the equity flow data is used to create the market regimes. The model created by clustering equity flow data was calculated to have a Sharpe ratio of of 0.24 while the model created using the analysis of return data is shown to have a Sharpe ratio of -0.21. The full results of the analysis of creating a portfolio by relying only on return data t create the regimes can be found in appendix IV.

Furthermore, the probability matrix of market regimes produced from equity flow data, shows that these clusters are more stable than clusters produced from clustering return data. These results show some of the benefits of using equity flow data to inform portfolio management decisions and strengthens the hypothesis that equity flows are more persistent and stable to returns.

The potential to use modern machine learning techniques to create profitably investment models has been shown during this study. Furthermore, this study shows the advantages of determining investor regimes using equity flow data in comparison to using return data. This is shown in the fact that the resulting portfolio is more profitable and less risky in addition to the regimes being more reliable. Self-organising maps were helpful in the exploratory analysis of large financial datasets and assisted in the selection of an appropriate number of market regimes to define going forward in the research. Hierarchical clustering used in conjunction with dynamic time warping were successfully implemented to inform a regime portfolio.

References

Levy, H., & Sarnat, M. (1970). Diversification, portfolio analysis and the uneasy case for conglomerate mergers. *The journal of finance*, *25*(4), 795-802.

Hitt, M. A., Tihanyi, L., Miller, T., & Connelly, B. (2006). International diversification: Antecedents, outcomes, and moderators. Journal of Management, 32(6), 831-867.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, *78*(9), 1464-1480.

Kaski, S. (1997). Data exploration using self-organizing maps. In *Acta polytechnica scandinavica: Mathematics, computing and management in engineering series no. 82*.

Ang, A., & Bekaert, G. (2004). How regimes affect asset allocation. *Financial Analysts Journal*, *60*(2), 86-99.

Burkov, A. (2019). *The Hundred-Page Machine Learning Book*. Andriy Burkov.

Berndt, D. J., & Clifford, J. (1994, July). Using dynamic time warping to find patterns in time series. In *KDD workshop* (Vol. 10, No. 16, pp. 359-370).

Becker, Y. L. and M. R. Reinganum (2018). "The Current State of Quantitative Equity Investing", CFA Institute Research Foundation.

Kahn, R. N. (2018). "The Future of Investment Management", CFA Institute Research Foundation.

Steinbach, M., Karypis, G., & Kumar, V. (2000, August). A comparison of document clustering techniques. In *KDD workshop on text mining* (Vol. 400, No. 1, pp. 525-526).

Froot, K. A., O'connell, P. G., & Seasholes, M. S. (2001). The portfolio flows of international investors. *Journal of financial Economics*, *59*(2), 151-193.

Appendix III – Code Extracts

This appendix presents some extracts from code used to carry out the ML experiment portion of this research.

This section of code reads in the relevant libraries into Rstudio. It is followed by code that reads in the daily equity flow data and the return data, assigning the data frames to variable names.

```
library(data.table)
library(datetime)
library(anytime)
library(kohonen)
library(lubridate)
library(dplyr)
library(dtwclust)

#getting weekly data
proper_regions$Date <- as.Date(x=proper_regions$Date, format='%d/%m/%Y')
proper_regions$Week_Day <- as.numeric(format(proper_regions$Date, format='%w'))
proper_regions$End_of_Week <- proper_regions$Date + (6 - proper_regions$Week_Day)
proper_regions$day <- proper_regions$Week_Day <- as.numeric(format(proper_regions$Date, format='%d'))
proper_regions$month <- proper_regions$Week_Day <- as.numeric(format(proper_regions$Date, format='%m'))
proper_regions$year <- proper_regions$Week_Day <- as.numeric(format(proper_regions$Date, format='%Y'))
proper_regions$year_month_SE <- paste(proper_regions$year, "-", proper_regions$month,
                                      "-", proper_regions$bi_month)
```

This code cleans the daily data to get it ready for analysis and transformation.

```
proper_regions$bi_month <- proper_regions$Week_Day
proper_regions
warnings()

nrow(weekly_data)
count(proper_regions$Week_Day)
proper_regions[1,11]

typeof(proper_regions$Week_Day)
proper_regions$End_of_Week

for (i in 1:5415){
  if(proper_regions[i,15] <=15){
    proper_regions[i,13] <- 0
    } else{
      proper_regions[i,13] <- 1
  }
}


proper_regions <- proper_regions[1:5411,]
```

This section transforms the daily data into weekly data.

```
#create new aggregated dataframe of weekly data
proper_regions$Date <- as.Date(x=proper_regions$Date, format='%d/%m/%Y')
proper_regions$Week_Day <- as.numeric(format(proper_regions$Date, format='%w'))
proper_regions$End_of_Week <- proper_regions$Date + (6 - proper_regions$Week_Day)

Europe_average <- aggregate(Europe~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
UK_average <- aggregate(UK~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
US_average <- aggregate(US~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
Japan_average <- aggregate(Japan~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
Latin_America_average <- aggregate(Latin_America~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
Eastern_Europe_average <- aggregate(Eastern_Europe~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
Commodity_Countries_average <- aggregate(Commodity_Countries~End_of_Week, FUN=mean, data=proper_regions, na.
AsiaxChina_average <- aggregate(AsiaxChina~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)
China_average <- aggregate(China~End_of_Week, FUN=mean, data=proper_regions, na.rm=TRUE)

weekly_data$Eastern_Europe[1084] <- -0.2
Eastern_Europe_average[nrow(Eastern_Europe_average)+1,] <- NA
nrow(Commodity_Countries_average)
#create new dataframe of weekly data
weekly_data <- data.frame("Date" = Europe_average$End_of_Week, "Europe" = Europe_average$Europe,
        "UK" = UK_average$UK, "US" = US_average$US, "Japan" = Japan_average$Japan,
        "Latin_America" = Latin_America_average$Latin_America, "Eastern_Europe" =
          Eastern_Europe_average$Eastern_Europe, "Commodity_Countries" =
          Commodity_Countries_average$Commodity_Countries, "AsiaxChina" =
          AsiaxChina_average$AsiaxChina, "China" = China_average$China )
```

Here code is presented that clusters the regions and tidies up the data frame for use in plots later. We end up with a data frame of average weekly returns organised by cluster (regime).

```
Europe_weekly_averages_Return <- aggregate(EUREXUK~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
UK_weekly_averages_Return  <- aggregate(UK~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
US_weekly_averages_Return  <- aggregate(US~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
Japan_weekly_averages_Return  <- aggregate(JP~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
Latin_America_weekly_averages_Return  <- aggregate(EMLATINAMERICA~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
Emasia_weekly_averages_Return  <- aggregate(EMASIA~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
Pacific_weekly_averages_Return  <- aggregate(PACIFICEXJAPAN~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
EMEA_weekly_Return  <- aggregate(EMEA~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)
CC_weekly_Return  <- aggregate(COMM~cluster, FUN=mean, data=CBF_weekly_WC, na.rm=TRUE)

average_returns_all_data <-data.frame("UK" = UK_weekly_averages_Return$UK, "Europe"
      = Europe_weekly_averages_Return$EUREXUK, "Japan" = Japan_weekly_averages_Return$JP,
    "US" = US_weekly_averages_Return $US, "Latin_America" = Latin_America_weekly_averages_Return
$EMLATINAMERICA, "EM_Asia"= Emasia_weekly_averages_Return $EMASIA, "Pacific"
          = Pacific_weekly_averages_Return$PACIFICEXJAPAN,
"EMEA" =  EMEA_weekly_Return$EMEA, "CC" =  CC_weekly_Return$COMM)
```

This R function transforms the data frame by transposing it, this is for graphing/plotting purposes.

```
t_average_returns_all_data <- t(average_returns_all_data)
t_average_returns_all_data <- as.data.frame(t_average_returns_all_data)
View(t_average_returns_all_data)

regions <- c("UK",  "Europe",        "Japan",         "US",
             "Latin_America",  "EM_Asia",       "Pacific" , "EMEA" ,         "CC"  )


t_average_returns_all_data$Regions <- regions
```

The following command create plots of our average weekly returns for each regime.

```
simple_bar_chart2 <- ggplot(data = t_average_returns_all_data, aes(x=Regions,
          y = t_average_returns_all_data$`V1`, fill=Regions))
+geom_bar(stat = "identity") + xlab("Regime 1")  + ylab("Average Returns")+ theme_minimal()
simple_bar_chart2


##############################
#Regime 2 flow comparision bar chart
simple_bar_chart3 <- ggplot(data = t_average_returns_all_data, aes(x=Regions,
      y = t_average_returns_all_data$`V2`, fill=Regions)) +geom_bar(stat = "identity") + xlab("Regime 2")
      + ylab("Average Returns")+ theme_minimal()
simple_bar_chart3
##############################
#Regime 3 flow comparision bar chart
simple_bar_chart4 <- ggplot(data = t_average_returns_all_data, aes(x=Regions,
      y = t_average_returns_all_data$`V3`, fill=Regions)) +geom_bar(stat = "identity") + xlab("Regime 3")
      + ylab("Average Returns")+ theme_minimal()
simple_bar_chart4

##############################
#Regime 4 flow comparision bar chart
simple_bar_chart5 <- ggplot(data = t_average_returns_all_data, aes(x=Regions,
      y = t_average_returns_all_data$`V4`, fill=Regions)) +geom_bar(stat = "identity") + xlab("Regime 4")
      + ylab("Average Returns")+ theme_minimal()
simple_bar_chart5
```

This code cleans he data frame for graph creation. Next long and short models are created.

```
CBF_weekly_WC$`weekly_with_clusters$cluster[145:1080]` <-NULL

names(CBF_weekly_WC) <- c("Europe","Latin_America","EMEA","Pacific",
                          "EM_Asia","US", "UK", "Japan", "CC","Date", "cluster" )

colnames(CBF_weekly_WC)

#short everything
CBF_weekly_WC$simple_regime_return_down <- (-1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe
      + CBF_weekly_WC$US + CBF_weekly_WC$Japan +CBF_weekly_WC$Latin_America
      + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC ) + 1

#long everyting
CBF_weekly_WC$simple_regime_return_up <- (1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe
      + CBF_weekly_WC$US + CBF_weekly_WC$Japan +CBF_weekly_WC$Latin_America
      + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC ) + 1


CBF_weekly_WC$simple_model_return <- 1
```

Hedge fund and simple model column are created here.

```
for (i in 1:936){
  CBF_weekly_WC$model_regime[i+1] <-  CBF_weekly_WC$cluster[i]

}

nrow(CBF_weekly_WC)
CBF_weekly_WC$simple_model_return <- 1


for (i in 1:936) {
  if (CBF_weekly_WC$model_regime[i] == 1){
    CBF_weekly_WC$simple_model_return[i] = CBF_weekly_WC$simple_regime_return_up[i]
  } else if (CBF_weekly_WC$model_regime[i] == 2){
    CBF_weekly_WC$simple_model_return[i] = CBF_weekly_WC$simple_regime_return_down[i]
  } else if (CBF_weekly_WC$model_regime[i] == 3){
    CBF_weekly_WC$simple_model_return[i] =  CBF_weekly_WC$simple_regime_return_up [i]
  } else {
    CBF_weekly_WC$simple_model_return[i] = CBF_weekly_WC$simple_regime_return_up [i]
  }
}


CBF_weekly_WC$simple_model_return[1] <- 1
CBF_weekly_WC$simple_model_return <- abs(CBF_weekly_WC$simple_model_return)


CBF_weekly_WC$simple_model_return[1] <- 1
CBF_weekly_WC$simple_model_return <- abs(CBF_weekly_WC$simple_model_return)

#############################################################################



CBF_weekly_WC$cumulative_return_simple <- CBF_weekly_WC$simple_model_return


for (i in 1:936){

  y = CBF_weekly_WC$cumulative_return_simple[i]
  x = CBF_weekly_WC$simple_model_return[i+1]
  z = x*y

  CBF_weekly_WC$cumulative_return_simple[i+1] <- z
}
#####################################################################

#create buy and hld column

CBF_weekly_WC$Buy_n_hold <- 1

for (i in 1:936){

  y = CBF_weekly_WC$Buy_n_hold[i]
  x = CBF_weekly_WC$simple_regime_return_up[i+1]
  z = x*y

  CBF_weekly_WC$Buy_n_hold[i+1] <- z
}


#libor

CBF_weekly_WC$libor <- 1


for (i in 1:936){

  CBF_weekly_WC$libor[i+1] <- (1.000126484)**i

}
```

The model columns have been created so this next portion of code creates visualisations of the model performances.

```
#proper_regions$Date <- as.Date(x=proper_regions$Date, format='%d/%m/%Y')
CBF_weekly_WC$date <- as.character(CBF_weekly_WC$Date)
CBF_weekly_WC$date <- as.date(x = CBF_weekly_WC$date, format = '%d/%m/%Y')
libor$Date <- as.date(x = libor$date, format = '%d/%m/%Y')


simple_model_plot_all <- ggplot(data = CBF_weekly_WC, aes(x=date,
            y = cumulative_return_simple)) + geom_line()
simple_model_plot_all +geom_line(data = CBF_weekly_WC, aes(x=date,
            y = Buy_n_hold), color = "blue") +theme_minimal() +geom_line(data = libor,
            aes(x=Date, y = libor), color = "red")
```

Here we create portfolio models again but this time using a different set of regimes created by clustering returns.

```
CBF_weekly_WC$regime_1_returns_LR <- (1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe
+ CBF_weekly_WC$US + CBF_weekly_WC$Japan +CBF_weekly_WC$Latin_America + CBF_weekly_WC$EM_Asia
+CBF_weekly_WC$Pacific + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC ) + 1

CBF_weekly_WC$regime_2_returns_LR <- 1

CBF_weekly_WC$regime_3_returns_LR <- (1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$US
+CBF_weekly_WC$Latin_America + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific +
 CBF_weekly_WC$EMEA + CBF_weekly_WC$CC ) + 1

CBF_weekly_WC$regime_4_returns_LR <- (1/9)*( CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific
        + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC + CBF_weekly_WC$Latin_America ) + 1


for (i in 1:936) {
  if (CBF_weekly_WC$model_regime[i] == 1){
    CBF_weekly_WC$low_risk_model_return[i] = CBF_weekly_WC$regime_1_returns_LR[i]
  } else if (CBF_weekly_WC$model_regime[i] == 2){
    CBF_weekly_WC$low_risk_model_return[i] = CBF_weekly_WC$regime_2_returns_LR[i]
  } else if (CBF_weekly_WC$model_regime[i] == 3){
    CBF_weekly_WC$low_risk_model_return[i] = CBF_weekly_WC$regime_3_returns_LR[i]
  } else {
    CBF_weekly_WC$low_risk_model_return[i] = CBF_weekly_WC$regime_4_returns_LR[i]
  }
}


CBF_weekly_WC$low_risk_model_return[1] <- 1
CBF_weekly_WC$cumulative_return_low_risk <- CBF_weekly_WC$low_risk_model_return

for (i in 1:936){

  y = CBF_weekly_WC$cumulative_return_low_risk[i]
  x = CBF_weekly_WC$low_risk_model_return[i+1]
  z = x*y

  CBF_weekly_WC$cumulative_return_low_risk[i+1] <- z
}


#short CC and LA
CBF_weekly_WC$regime_1_returns_H <- (1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe
  +CBF_weekly_WC$US + CBF_weekly_WC$Japan +CBF_weekly_WC$Latin_America
  + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC ) + 1

#long everyting
CBF_weekly_WC$regime_2_returns_H <- (1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe
  + CBF_weekly_WC$US + CBF_weekly_WC$Japan +CBF_weekly_WC$Latin_America
  + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC ) + 1

#short
CBF_weekly_WC$regime_3_returns_H <- (1/9)*(CBF_weekly_WC$UK  + CBF_weekly_WC$US
  + CBF_weekly_WC$Japan  + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific
  +CBF_weekly_WC$EMEA + CBF_weekly_WC$Latin_America ) -(1/9)*(CBF_weekly_WC$Japan
  + CBF_weekly_WC$Europe) + 1


#short everything
CBF_weekly_WC$regime_4_returns_H <- (1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe
  + CBF_weekly_WC$US + CBF_weekly_WC$Japan +CBF_weekly_WC$Latin_America
  + CBF_weekly_WC$EM_Asia +CBF_weekly_WC$Pacific + CBF_weekly_WC$EMEA + CBF_weekly_WC$CC )
  +(-1/9)*(CBF_weekly_WC$UK + CBF_weekly_WC$Europe + CBF_weekly_WC$US + CBF_weekly_WC$Japan)  + 1
```

Return hedge fund model is implemented in this portion of code.

```
CBF_weekly_WC$hedgefund_model_return <- 1


for (i in 1:936) {
  if (CBF_weekly_WC$model_regime[i] == 1){
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_1_returns_H[i]
  } else if (CBF_weekly_WC$model_regime[i] == 2){
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_2_returns_H[i]
  } else if (CBF_weekly_WC$model_regime[i] == 3){
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_3_returns_H[i]
  } else {
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_4_returns_H[i]
  }
}




CBF_weekly_WC$hedgefund_model_return[1] <- 1
CBF_weekly_WC$hedgefund_model_return <- abs(CBF_weekly_WC$hedgefund_model_return)


CBF_weekly_WC$hedgefund_model_return <- 1


for (i in 1:936) {
  if (CBF_weekly_WC$model_regime[i] == 1){
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_1_returns_H[i]
  } else if (CBF_weekly_WC$model_regime[i] == 2){
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_2_returns_H[i]
  } else if (CBF_weekly_WC$model_regime[i] == 3){
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_3_returns_H[i]
  } else {
    CBF_weekly_WC$hedgefund_model_return[i] = CBF_weekly_WC$regime_4_returns_H[i]
  }
}




CBF_weekly_WC$hedgefund_model_return[1] <- 1
CBF_weekly_WC$hedgefund_model_return <- abs(CBF_weekly_WC$hedgefund_model_return)
```

These portfolio models are graphed using the package ggplot2.

```
#find cumulative moedel of hedgefund model

CBF_weekly_WC$cumulative_return_hedgefund <- CBF_weekly_WC$hedgefund_model_return

for (i in 1:936){

  y = CBF_weekly_WC$cumulative_return_hedgefund[i]
  x = CBF_weekly_WC$hedgefund_model_return[i+1]
  z = x*y

  CBF_weekly_WC$cumulative_return_hedgefund[i+1] <- z
}


redo_simple <- ggplot(data = CBF_weekly_WC, aes(x=date, y = cumulative_return_simple))
               +geom_line( color = "blue")


redo_simple + geom_line(data = CBF_weekly_WC, aes(x=date, y = Buy_n_hold) , color = "purple")
+ geom_line(data = CBF_weekly_WC, aes(x=date, y = cumulative_return_low_risk), color = "black")
+ geom_line(data = CBF_weekly_WC, aes(x=date, y = cumulative_return_hedgefund), color = "turquoise")
+ geom_line(data = libor, aes(x=Date, y = libor), color = "grey") + ylab("Cumulative Return") +xlab("")
+ theme(legend.position = "none") +theme_minimal()
```