STATISTICAL TIME SERIES ANALYSIS METHODS WITH APPLICATIONS TO PORTFOLIO MANAGEMENT

by

Yaman Kındap

B.S., Chemical Engineering, Boğaziçi University, 2017

B.S., Physics, Boğaziçi University, 2017

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2020

ACKNOWLEDGEMENTS

I would like to express my gratitude to Ali Taylan Cemgil for enabling me to change my academic field and pursue a career in machine learning. His expertise and intuitive explanation of abstract mathematical concepts has laid a solid foundation in probability and statistics for me, and has given me the ambition to achieve a deeper understanding in these fields for which I am grateful. In his absence, I would like to thank Lale Akarun for accepting me as a graduate student, inviting me to her reading group and her valuable advice on this work. Likewise, I would like to thank my examiner Sinan Yıldırım for his helpful feedback on this work.

I would also like to thank my friends and colleagues at Algosis where I had the pleasure of working on real problems. Firstly, I would like to thank Şafak Uluğ and Furkan Enes Yalçın for helping me become a better software programmer, and Yener Ülker for sharing his extensive knowledge in signal processing and financial systems. I would especially like to thank Arman Boyacı, who has become a great mentor and taught me how to critically think about challenges in both academic research and business.

This work has been completed during the COVID-19 pandemic which has disrupted the works of many researchers. I have been incredibly fortunate to have the support of my parents Çiğdem and Ayhan Kındap which enabled me to work on my thesis without experiencing the negative impacts of the pandemic. Lastly, I would like to thank Seren Ocak for her continued support throughout years, for listening to my ramblings about abstract ideas and helping me understand them, and motivating me to follow my dreams.

ABSTRACT

STATISTICAL TIME SERIES ANALYSIS METHODS WITH APPLICATIONS TO PORTFOLIO MANAGEMENT

In many domains of science and engineering, such as signal processing, bioinformatics and computational finance, sequential data modelling and analysis is essential for various tasks including clustering, anomaly detection and forecasting. In this work, we present the fundamentals of time series analysis methods with a focus on modelling dynamical systems. Our goal is to make statistical inferences that are able to account for our uncertainty about the system while also being able to incorporate domain specific knowledge into these system representations.

Hidden Markov models have been extensively studied in the literature because of their relative simplicity and flexibility. We propose an extension to this model called the Gaussian-Gamma hidden Markov model which introduces an additional latent scale parameter, along with its state inference and parameter estimation algorithms. The model is inspired by our prior knowledge of financial markets in terms of displaying persistent regimes and having heavy-tailed distributions. The intuition behind the need for such a model in computational finance is discussed with respect to the shortcomings of the standard mathematical framework of constructing an optimal portfolio called the Mean-Variance Analysis. We illustrate the performance of our model in both synthetically generated and real financial data sets with regime identification and portfolio management problems. Results show that our model is able to discover meaningful insights about the dynamics of financial markets.

ÖZET

İSTATİSTİKSEL ZAMAN SERİSİ ANALİZ YÖNTEMLERİ İLE PORTFÖY YÖNETİMİ UYGULAMALARI

Sıralı veri modelleme ve analizi, sinyal işleme, biyoenformatik ve hesaplamalı finans gibi birçok bilim ve mühendislik alanında kümeleme, olağandışılık tespiti ve öngörme görevlerinde önem taşır. Bu çalışmada, dinamik sistemlerin modellenmesine odaklanarak zaman serisi analiz yöntemlerinin temellerini sunuyoruz. Amacımız, sistem hakkındaki belirsizliğimizi açıklayabilen ve aynı zamanda alana özgü bilgileri bu sistem temsillerine dahil edebilen istatistiksel çıkarımlar yapabilmektir.

Saklı Markov modelleri, göreceli sadelikleri ve esneklikleri nedeniyle literatürde kapsamlı bir şekilde incelenmiştir. Bu modele ilave bir saklı ölçek değişkeni sunarak, Gauss-Gamma dağılımlı saklı Markov modeli adında yeni bir model ve bu modelin durum çıkarımı ve parametre kestirme algoritmaları ileri sürüyoruz. Model, kalıcı rejimler sergilemek ve ağır kuyruklu dağılımlara sahip olmak açısından finansal piyasaların dinamiklerinden ilham alıyor. Hesaplamalı finans alanında böyle bir modele duyulan ihtiyacın arkasındaki nedenler, optimum portföy oluşturma alanında bir standart olan Ortalama-Varyans Analizinin matematiksel eksiklikleri ile ilişkilendirilerek tartışılmaktadır. Modelin performansı yapay üretilmiş ve gerçek finansal veri setleri üzerinde rejim belirleme ve portföy yönetimi deneyleri yürütülerek incelenmiştir. Sonuçlar modelin finansal piyasa dinamikleri hakkında anlamlı çıkarımlar yapabildiğini göstermektedir.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . iii
ABSTRACT . . . iv
ÖZET . . . v
LIST OF FIGURES . . . viii
LIST OF TABLES . . . x
LIST OF SYMBOLS . . . xi
LIST OF ACRONYMS/ABBREVIATIONS . . . xiii
1. INTRODUCTION . . . 1
1.1. Challenges of Dynamics in Financial Systems . . . 3
1.2. Intuition Behind Diversification . . . 7
1.3. Related Work . . . 10
1.4. Scope of Work . . . 11
1.5. Organization of Thesis . . . 12
2. THEORETICAL BACKGROUND . . . 13
2.1. Time Series Modeling . . . 13
2.2. Markov Chains . . . 15
2.2.1. Stochastic Matrices . . . 17
2.2.2. Autoregressive Model . . . 20
2.3. Latent Variable Models . . . 24
2.4. Clustering . . . 27
2.5. Hidden Markov Model . . . 28
2.5.1. Inference . . . 30
2.5.1.1. Forward Algorithm . . . 31
2.5.1.2. Backward Algorithm . . . 34
2.5.1.3. Forward-Backward Algorithm . . . 35
2.5.2. Learning . . . 35
2.5.2.1. The Baum-Welch Algorithm . . . 41
2.5.3. Handling Imbalanced Data Sets . . . 45

2.6. Exponential Families 46
3. GAUSSIAN-GAMMA HIDDEN MARKOV MODEL 50
3.1. Inference 54
3.1.1. Filtering 55
3.1.2. Moments of the Latent Scaling Variable 57
3.1.3. Smoothing 60
3.2. Learning 62
4. APPLICATION OF GAUSSIAN-GAMMA HIDDEN MARKOV MODEL . . 66
4.1. Modern Portfolio Theory 66
4.1.1. Financial Interpretations 66
4.1.2. Mathematical Framework 68
4.1.3. Portfolio Optimization 72
4.2. Asset Allocation with GHMM 76
4.3. Asset Allocation with GGHMM 77
5. EXPERIMENTS AND RESULTS 79
5.1. Synthetic Data 80
5.1.1. Evaluation Criteria 80
5.1.2. Visualization of Experiments 81
5.1.3. Performance Statistics of Experiments 85
5.2. Financial Data and Asset Allocation 88
5.2.1. Regime Identification 88
5.2.2. Asset Allocation 97
6. CONCLUSION 102
REFERENCES 105
APPENDIX A: PROBABILITY DENSITY REFRESHER 112
A.1. Multivariate Gaussian Distribution 112
A.2. Gamma Distribution 112
APPENDIX B: MULTIVARIATE T-DISTRIBUTION DERIVATION 113

LIST OF FIGURES

Figure 1.1. Visualization of dynamical systems. 2

Figure 1.2. Visualization of autocorrelation in stock returns. 5

Figure 1.3. Visualization of heavy-tails in stock returns. 7

Figure 1.4. Visualization of synthetic assets and portfolios. 9

Figure 2.1. First order Markov chain as a graphical model. 16

Figure 2.2. AR(1) model on stock prices. 22

Figure 2.3. AR(1) model on stock returns. 23

Figure 2.4. Graphical model of a state-space representation. 24

Figure 3.1. Gaussian-gamma hidden Markov model. 52

Figure 4.1. Visualization of real assets and portfolios. 67

Figure 5.1. Visualization of synthetically generated data with corresponding states. 82

Figure 5.2. Visualization of state estimates by GGHMM and GHMM. 83

Figure 5.3. Cumulative returns of each asset. 89

Figure 5.4. State identification with smoothed estimates of GHMM. 91

Figure 5.5. State identification with smoothed estimates of GGHMM. 94

Figure 5.6. Difference in densities of equal variance distributions. 97

Figure 5.7. Asset allocation based on GHMM. 99

Figure 5.8. Asset allocation based on GGHMM. 100

LIST OF TABLES

Table 5.1. Performance statistics of equally trained models 85

Table 5.2. Performance statistics of 20 to 50 epoch realizations 86

Table 5.3. Performance statistics of equally trained models in 5-dimensional observation space 87

Table 5.4. Annualized Mean Return Results for GHMM 92

Table 5.5. Covariance characteristics learned by GHMM for the turbulent state 93

Table 5.6. Covariance characteristics learned by GHMM for the non-turbulent state 93

Table 5.7. Annualized Mean Return Results for GGHMM 95

Table 5.8. Covariance characteristics learned by GGHMM for the turbulent state 96

Table 5.9. Covariance characteristics learned by GGHMM for the non-turbulent state 96

LIST OF SYMBOLS

a_{ij}	Element of transition matrix A
$\mathbf{A}$	State transition matrix of a hidden Markov model
Categorical(.)	Categorical distribution
C	Constant terms
Gamma(.)	Gamma distribution
Gaussian(.)	Gaussian distribution
L(.)	Likelihood function
$l(.)$	Loglikelihood function
LB(.)	Lower-bound functional
N	Dimension of hidden states
p(.)	Probability distribution function
p_w	Portfolio return vector with weights ω
q(.)	Variational distribution function
r_i	Return series of asset i
$\mathbf{r}_t$	Return vector of at time t
t	Current time
T	Total number of time slices
w_t	Latent scale variable at time t
x_t	Latent variable vector at time t
$x_{1:T}$	Set of latent variables
y_t	Observation vector at time t
$y_{1:T}$	Set of observations
α	Expected return vector of assets
$\alpha_t(x_t)$	Forward recursion
α_ω	Expected portfolio return with weights ω
$\beta_t(x_t)$	Backward recursion
$\Gamma(.)$	Gamma function

$\gamma_t(x_t)$	Forward-Backward recursion
ϵ_t	System Gaussian noise at time t
θ	Parameter set
μ	Mean
ν	Degrees of freedom parameter
ν_t	Measurement Gaussian noise at time t
$\xi(.)$	Joint latent variable distribution
π	Initial state probability vector
Σ	Covariance matrix
σ	Standard deviation parameter
ω	Vector of weights

LIST OF ACRONYMS/ABBREVIATIONS

AR	Autoregressive
ARCH	Autoregressive Conditional Heteroscedasticity
ARMA	Autoregressive Moving-Average
ARIMA	Autoregressive Integrated Moving-Average
DNA	Deoxyribonucleic acid
EM	Expectation Maximization
GARCH	Generalized Autoregressive Conditional Heteroscedasticity
GHMM	Gaussian Hidden Markov Model
GGHMM	Gaussian-Gamma Hidden Markov Model
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
KKT	Karush–Kuhn–Tucker
MA	Moving-Average
MVA	Mean-Variance Analysis
NBER	National Bureau of Economic Research
SHMM	Student's t-distribution Hidden Markov Model
VAR	Vector Autoregression

1. INTRODUCTION

Modelling sequential data and making inference on these systems is an essential area of research in various fields of science and engineering. Sequential data may arise from dynamical systems in the form of a time series or a single dimensional spatial process where the position of an observation is critical such as in biological sequences. While the evolution of a dynamical system is often described by a deterministic function, some systems are inherently stochastic or the complexity of some systems may require statistical arguments in which case the model of the system evolution is stochastic.

While randomness caused by measurement noise is involved in most systems of interest, the evolution of the dynamical system may be stationary which simply means that some statistical properties of the system are constant with respect to time. Such systems can be modelled by evolution functions that are independent of time. On the other hand, non-stationary dynamical systems present the additional problem that the evolution mechanism of the system also changes in time. Some observations from dynamical systems are shown in Figure 1.1 which belong to speech, biological and financial signals. The analyses of these signals are necessary in order to tackle problems in their respective domains such as predicting emotions, biological responses and future returns of an asset. It is clearly visible that all of these systems display non-stationary behavior, especially in the speech and financial domains. In the case of a speech signal, non-stationary behavior is caused by the changing frequency components that are used to produce different vowel and consonant sounds. For financial signals, the magnitude of variation in the signal depends on changing government policies, regulations, etc. and sentiment on current events which introduces non-stationary behavior to the system.

With all the success signal processing models and algorithms had in all of the mentioned fields, one particular field has yet to realize the potential benefits. Popular signal processing algorithms used in various fields fail to produce tangible results in

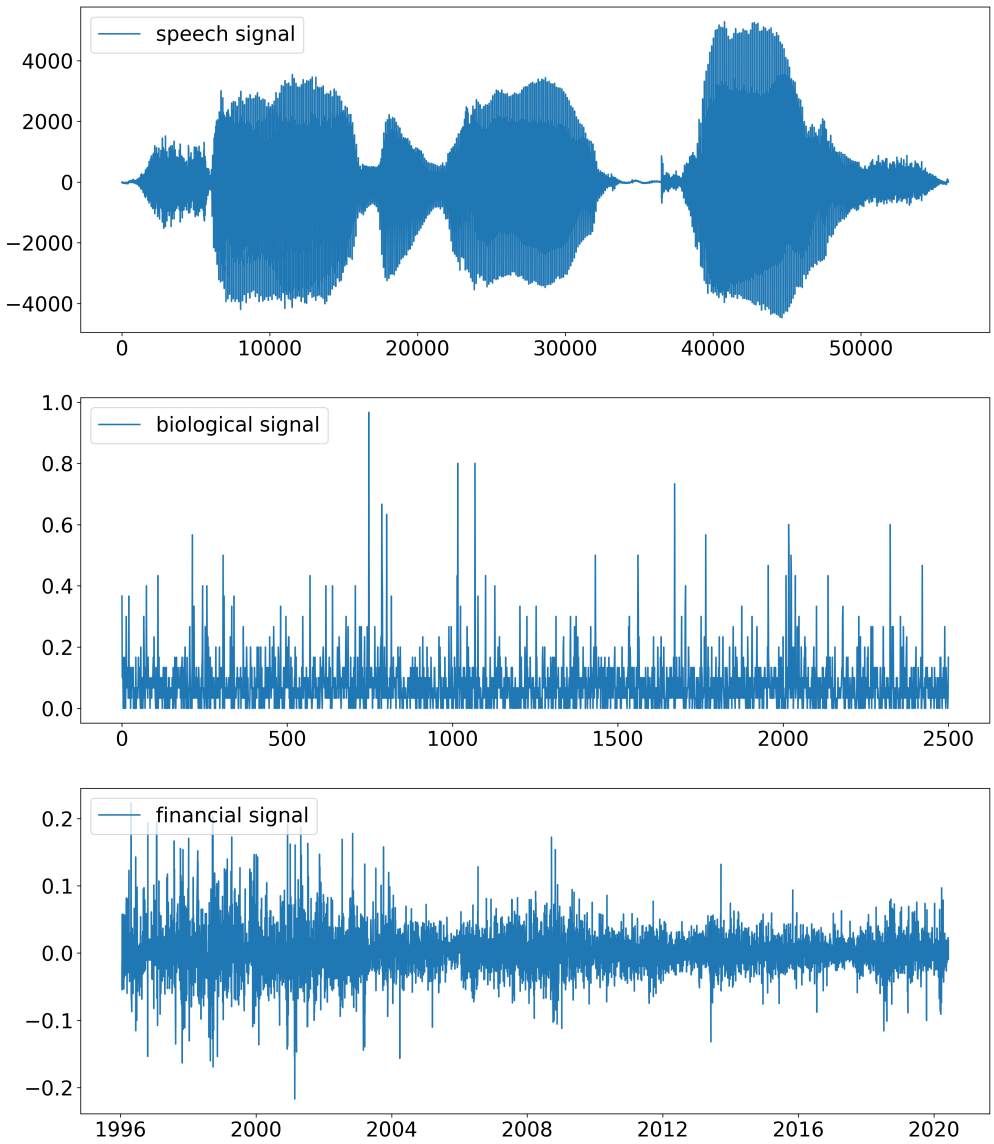

Figure 1.1. Visualization of dynamical systems.

computational finance. The most apparent reason for this is that financial markets have a considerably low signal-to-noise ratio. In other words, the algorithms cannot differentiate between which signals actually contain information and which signals are purely the result of random behavior of millions of people participating in the markets. This creates a unique challenge for researchers in the signal processing and machine learning communities since the successful implementation of machine learning algorithms for financial series requires a deep understanding of financial markets as well as mathematical methods. Data-driven strategies instead of financial considerations are increasingly being used in financial markets where the performance of quantitative experts outperform financial experts in the long run [1].

1.1. Challenges of Dynamics in Financial Systems

The development of probability theory and statistics are historically related to the understanding of uncertainty and being able to have a rational measure of it. In the 17th and 18th centuries, these developments were motivated by the desire to profit from games of chance where the outcome of the game is influenced by some source of randomness. In the recent past, with a similar motivation, numerical methods and mathematical models are adapted for use in financial markets which produced the birth of computational finance. Uncertainty is one of the most important characteristic features of financial markets and the development of methods to distinguish informative signals from noise is crucial. Thus, our main subject in this work is to discuss ways of rational decision making in highly uncertain systems.

Portfolio management for the purpose of meeting long-term financial objectives and managing risk is one of the main problems in computational finance. This task requires creating mathematical models of stochastic price series of assets as well as a rational framework for decision making. Such models should be able to accurately represent the dynamics of individual assets and their interdependence while also taking uncertainty into account.

A notable observation of financial markets is that assets respond differently to macroeconomic factors. While assets individually have an inherent uncertainty, the difference in their response to factors can be exploited by investors in order to reduce the uncertainty in the outcome of their individual investments. By combining assets in an intelligent way the exposure to the inherent uncertainty of any single asset can be limited. The resulting combination of assets is called a portfolio and this strategy is broadly defined as diversification. This strategy is at the core of portfolio management and the intuition behind it is presented in section 1.2.

One of the earliest mathematical frameworks for constructing such a portfolio is the mean-variance analysis by Harry Markowitz which was later awarded a Nobel Prize in Economics [2]. In his work, Markowitz reduces the available asset classes into two dimensional entities with sample mean returns and covariance matrix. This approach lets an investor allocate their wealth by taking the correlations between asset classes into consideration. In this work, we take the framework introduced by mean-variance analysis as a standard in portfolio management and build on it by introducing improved inference methods for mean returns and covariance of assets influenced by empirical evidence and financial intuition.

Let's start by addressing the fundamental limitation of mean-variance analysis which is that sample mean and covariance is an accurate representation of the statistical properties and future behavior of asset returns. In addition to obviously being generated sequentially, some properties of asset returns display autocorrelation which is a linear measure of the dependence between a variable's current value and its past values. The autocorrelation of stock returns and it's monthly non-overlapping sliding window standard deviation are presented in Figure 1.2. The results show that while the return series seem like a random walk process [3], the non-overlapping sliding window standard deviation of return series display strong autocorrelation. This fact can be interpreted as, the magnitude of change in stock prices tends to be similar in close temporal regions.

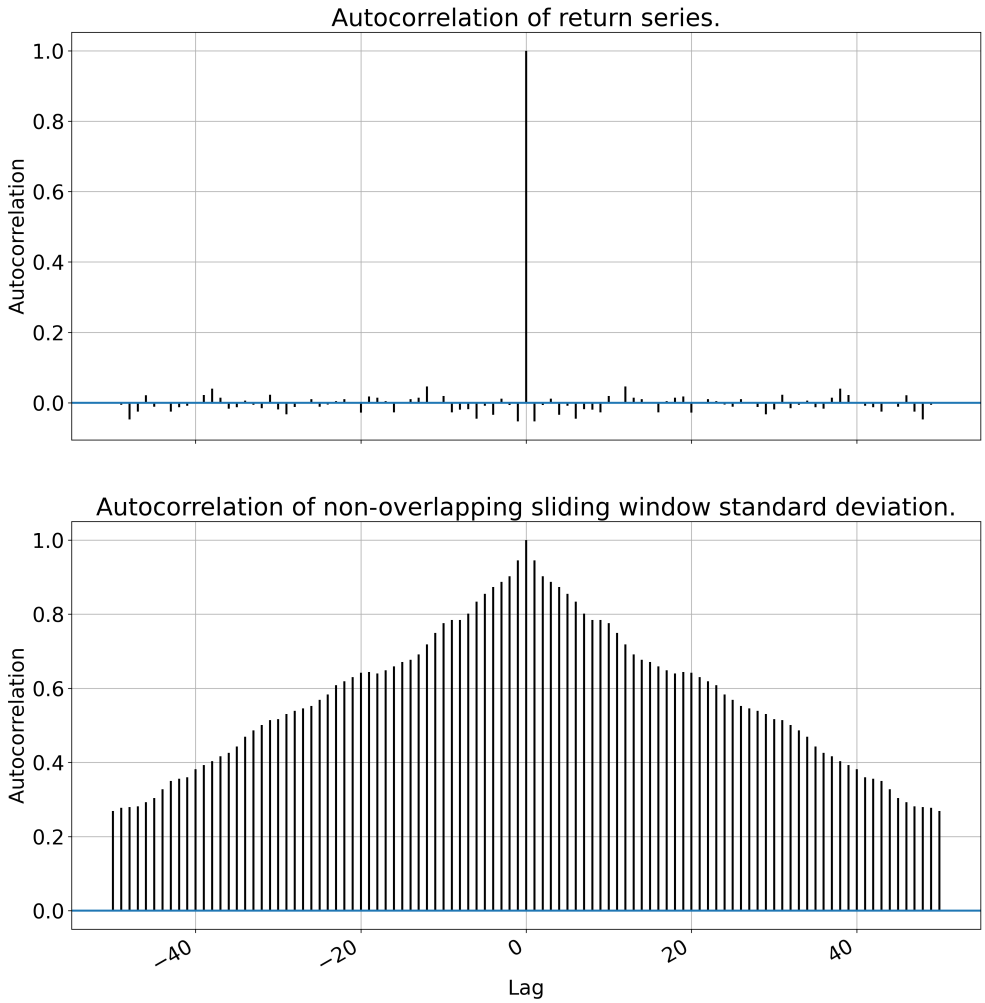

Figure 1.2. Visualization of autocorrelation in stock returns.

On top of individual assets displaying autocorrelation in standard deviations, evidence shows that financial markets sometimes experience abrupt changes in their behavior which then tends to persist for some while [4]. These varying characteristic behaviors of financial markets are defined as market regimes and are influenced by macroeconomic variables. For example, research shows that there is an increase in the correlations between asset classes during financial crises [5]. While the market regime cannot be directly observed in real time, the class of statistical models called the latent variable models are especially well-suited for carrying out such an analysis.

Combining these two observations that asset returns display temporal dependence and there are market regime switching behavior a more realistic method of infering mean returns and covariance of assets compared to using sample statistics can be constructed. One extensively studied model that integrates both observations is the Gaussian hidden Markov model (GHMM). Using discrete latent state variables, asset returns can be clustered in order to identify different characteristic behaviors of financial markets as a whole. Furthermore, the mean and covariance of the distribution of assets in each regime can be learned while also taking the temporal structure of the data set into account.

The limitation of using a GHMM is that such a model assumes that asset returns follow a normal distribution which is invalidated by empirical data. Financial data series display extreme deviations from its median behavior which are represented by heavy-tailed distributions in statistics. A comparison of using a Gaussian density and a t-distribution density which is heavy-tailed in stock returns is shown in Figure 1.3. The fact that financial data series display heavy-tailed behavior needs to be taken into account for a more realistic representation.

In this work, we propose a new mathematical model of asset prices by introducing an additional latent scale parameter into the GHMM which enables the dynamical modeling of the covariance of assets as well as take the heavy-tailed nature of returns into account. The proposed model is defined as the Gaussian-Gamma hidden Markov

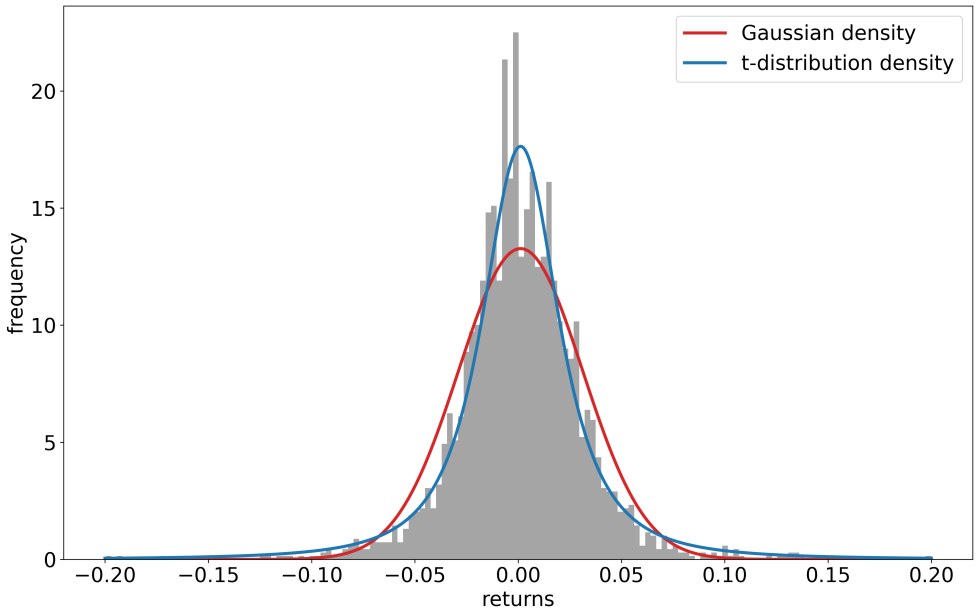

Figure 1.3. Visualization of heavy-tails in stock returns.

model (GGHMM). Algorithms for inference of latent variables and learning the parameters of the proposed model are presented along with their theoretical formulations.

1.2. Intuition Behind Diversification

There is an inherent duality between consumption, which can be defined as spending money in order to receive some type of satisfaction from consuming a good or service; and saving, defined as the amount of income that is not spent or consumed for immediate satisfaction. There are lots of reasons to save money, including emergency situations in the future that might require spending, private education and retirement. An investment is a method of saving that can simply be described as an allocation of money into assets with the objective of increasing wealth.

Assets are properties that are considered to have value and can be categorized as financial and non-financial. Most individuals and institutions own various non-

financial assets which have a physical value. Examples include real estate, vehicle or a computer. In the context of finance, assets usually derive their value from contractual claim rather than a physical claim. Some of the major financial asset classes are equities, commodities which include gold and crude oil, and cash equivalents which include foreign exchange currencies. The goal of portfolio management is to create an allocation strategy in such a way that suits the individual needs of the investor.

The difficulty in allocation of wealth is that assets usually have an uncertainty associated with their outcomes. Thus, in order to make an informed decision, we need to create mathematical models that represent the assets and analyze their statistics in order to find useful patterns in the data. Let's start by considering the historical sample means and standard deviations of some hypothetical assets which are shown in Figure 1.4.

The black dots in the figure represent the yearly expected return and standard deviation of 3 different assets. The first thing to notice in this plot is that as the expected return of an asset increase, its standard deviation also increases. If that wasn't the case, an asset with a high standard deviation and low expected return would never find an investor. As the standard deviation increase, the uncertainty associated with the asset also increase. Depending on how much uncertainty the investor finds acceptable, the choice of asset will be different. However, let's say that the investor aims to gain an annual return of 0.05 and does not want to risk losing her wealth for a higher expected return.

One of the most important observations in investment management is that, we can actually combine the other two assets in Figure 1.4 in order to create a portfolio that has an annual return of 0.05 as requested by the investor but has a lower standard deviation compared to 0.10. How much better statistical properties this portfolio will have depends on the covariance of the two assets. The curved lines in Figure 1.4 shows all possible weighted combinations of the two assets. As shown in the figure, as long as the assets are not completely linear which is rarely the case in financial markets, we

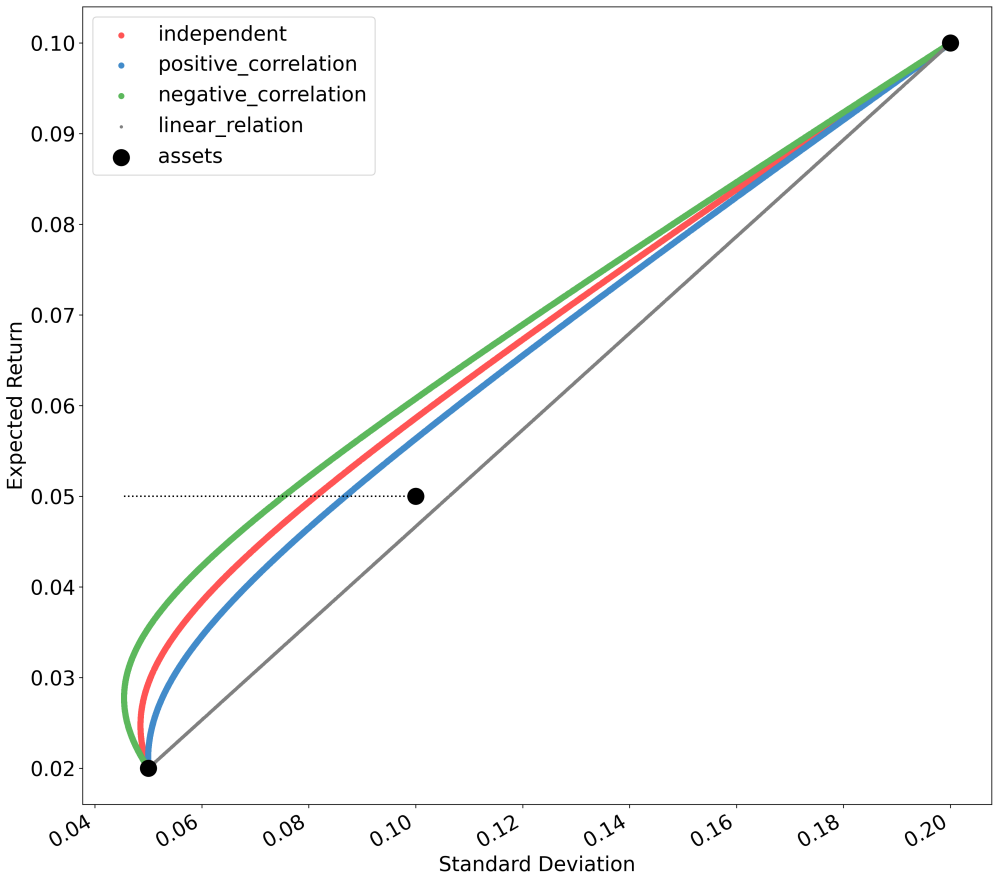

Figure 1.4. Visualization of synthetic assets and portfolios.

can construct a portfolio with a lower standard deviation. The act of allocating wealth into multiple assets is called diversification and the intuition behind it is demonstrated in this toy example. Notice that by diversifying we are always decreasing our expected return because expectation is a linear operator. On the other hand, covariance is not a linear operator, this makes the improvement in standard deviation possible.

1.3. Related Work

The hidden Markov model has been studied in the context of various applications including speech recognition [6] and bioinformatics [7] starting from the 1960s. The problem of parameter initialization and determination of the number of latent states in a system is studied in [8]. An introductory review on latent variable models and their relation to the GHMM are presented in [9] and a more generalized approach to modelling time series with latent variables is presented in [10] and [11].

While one of the aims of this study is to learn the clustering structure of standard deviations in financial data series without explicitly deriving an algorithm to cluster on standard deviations, we should mention stochastic volatility models where the variance is explicitly modelled using continuous latent variables. Such models start with the assumption that variance has a clustering structure and the observed variance is a noisy measurement of the actual system variance. A review on such models is presented in [12].

The application of GHMMs in financial modelling has been extensively studied in the literature, particularly for the task of identifying market regimes [13]. In the context of asset allocation [14] presents an introductory discussion on how to interpret the learned parameters of the a GHMM. The relation between mean-variance analysis and GHMMs are established in [15] and [16].

The use of scale mixture of Gaussian distributions for the purpose of modelling heavy-tailed observations is presented in [17]. Recently, the extension of hidden Markov

models to include heavy-tailed observations are studied, especially in the case of Student's t-distribution [18]. It has been shown that the Student's t-distribution HMM (SHMM) is able to identify more persistent states in time series data [19]. The use of SHMMs in classification tasks is presented in [20]. Similar to our motivation, [21] presents the use of SHMMs in financial heavy-tailed data sets and their relevance.

1.4. Scope of Work

Our proposed model GGHMM is mathematically similar to SHMM in terms of its formulation. However, instead of integrating over all possible scale mixtures of Gaussian distributions in order to form a t-distribution, the GGHMM explicitly identifies which scale mixture component generates the observations. This adjustment allows the variance of a particular state in the system to dynamically change depending on the observations which allows us to model heavy-tailed distributions while also being able to utilize our intuitive understanding of the Gaussian distribution for observations.

Beside the formulation of the proposed model, our contribution includes the use of the fact that the moments of exponential families can be calculated using the derivatives of the log-normalizer in the parameter estimation of both GGHMM and SHMMs. The derivation of an expectation-maximization algorithm for parameter estimation in GGHMMs is presented along with forward-backward inference algorithms for state estimation.

Since it is not possible to evaluate the performance of our model on its accuracy in inferring true latent states for real financial data sets, we introduce an asset allocation framework which builds on the mean-variance analysis framework for both GHMMs and GGHMMs. This framework allows us to discuss how well the identified states represent our prior financial knowledge.

1.5. Organization of Thesis

In chapter 2, we start by presenting considerations for sequential data modelling and latent variable models. Inference and parameter learning in hidden Markov model with Gaussian emissions is covered in detail. We end the chapter by presenting an introduction to exponential families and their properties related to our derivations.

In chapter 3, we introduce the Gaussian-Gamma hidden Markov model and present its similarities and differences with the GHMM. State inference and parameter learning algorithms along with their derivations are shown.

In chapter 4, we introduce the mathematical framework of the mean-variance analysis for portfolio optimization in detail. We discuss its theoretical drawbacks and present asset allocation frameworks using the GHMM and GGHMM in order to mitigate these drawbacks.

In chapter 5, we present the results of some experiments that highlight the strengths and weaknesses of our proposed model. These properties are discussed in comparison to GHMMs for a thorough assessment of the model. The experiments include synthetically generated data as well as real financial data sets. The performance of GGHMM in regime identification and asset allocation are presented.

In chapter 6, we present a summary of our discussions and draw conclusions from the results of our experiments. Finally we end our work by discussing possible future applications and research concerning the GGHMMs and provide some auxiliary derivations in the appendix.

2. THEORETICAL BACKGROUND

2.1. Time Series Modeling

In most machine learning applications the training data set is assumed to be independent and identically distributed (i.i.d.) which greatly simplifies the underlying mathematical models. This implies that all observations $(y_1, \ldots, y_T)$, where $T \in \mathbb{Z}^+$, independently reflect some property of the same underlying generative process and the order in which we observe the data set is inconsequential because there is no interdependence between them. However, the i.i.d. assumption breaks down when the process in question is a system that is changing with a dependence on time such as financial data or the data has an ordered sequential structure such as nucleotide bases along a DNA strand. While time series models which are discussed in this work are applicable to both types of data sets, we will be focusing on temporal relationships. A more detailed review on time series modelling is presented in [22].

Systems that are changing in time are generally referred to as dynamical systems and described using differential equations in natural sciences and engineering disciplines where the evolution of the system is considered to be continuous. Nevertheless, for many applications discretization of the time domain and interpolating the continuity of the evolution is sufficient and computationally more efficient. In fact, most financial analyses use discrete data points collected at periodic intervals such as the daily closing price of stocks, company earnings for a business quarter or the annual gross domestic product of a country.

An important structural distinction between time series data is having a stationary or non-stationary distribution. While stationarity is a strong mathematical assumption, a more relaxed assumption, called the weak-sense stationary, assumes that the mean and autocovariance are constant in time and the variance is defined. In other words, the statistical distribution from which data are generated is constant.

For example, the statistical interdependence between y_{t-l} and y_t should be the same as y_t and y_{t+l} where l is any positive integer number. Any time series data that does not satisfy this definition is considered to be non-stationary. In this more general case, the properties of the statistical distribution is assumed to change with time. However, the weak-sense stationary assumption is frequently made in order to simplify the mathematical models and such models mostly yield adequate results.

One of the defining properties of a time series analysis is the choice of the system's mathematical representation. We have mentioned that a dynamical system may be represented as a set of differential equations when we are dealing with continuous time problems. While for many physical application domains such as fluid dynamics and heat transfer, these equations have a theoretical basis for being chosen to represent the system, other domains such as signal processing or econometrics may not have a straightforward choice of representation. Thus in general, the choice of representation has an inherent uncertainty associated with it since the data generating process is unknown or the system is too complex to represent efficiently. The representation of a system is also referred to as its model throughout the literature.

In general, learning the representation of a system can be described as finding a function $f(x, \theta)$, where x is a variable related to the system which we can observe and measure, and θ are the parameters of the function. In dynamical systems, the goal of learning this function is to understand how the system changes in time and make predictions. In the frequentist approach to statistical inference this problem can be separated into two stages: (1) Finding a suitable functional structure and (2) Finding the best set of parameters θ for the function. On the other hand, a Bayesian approach to learning a representation focuses on the functional structure of the process and considers a weighted average over all possible sets of parameters that the function can have.

Naively, an approach to model a dynamical system may be to replace x with t which represents time explicitly. However, this approach is misleading since dynamical

systems generally do not depend on the absolute magnitude of time. Instead, the system depends on its own properties at a previous point in time. This is caused by the fact that change in a system occurs continuously. For example, if you are tracking the location of your smartphone, the previous location of the phone a second ago gives a pretty good approximation for its location now. Thus, in general we can represent the change in a system with a function $y_t = f(y_{1:t-1}, \theta)$ where $y_{1:t-1}$ is a shorthand notation for the set $(y_1, \ldots, y_{t-1})$ of all previous observations of the system.

Notice that the complexity of representing a dynamical system as a function $y_t = f(y_{1:t-1}, \theta)$ grows with t indefinitely which is an undesirable property. One intuition we have already discussed in the smartphone example is that recent observations may be much more informative. This intuition can be encoded into a model by using a discrete-time Markov chain which mathematically describes randomly evolving phenomena with limited memory of its past [23].

2.2. Markov Chains

In general, the complete description of a sequence $y_{1:T}$ requires the joint probability distribution $p(y_1, \ldots, y_T)$ to be defined. Without loss of generality, the chain rule of probability can be used to factorize the distribution using conditional probability distributions as shown in equation 2.1.

$$p(y_1, \ldots, y_T) = \prod_{t=1}^{T} p(y_t | y_1, \ldots, y_{t-1}) \tag{2.1}$$

In such a factorization, the conditional probability of each variable in the sequence depends on the value of all previous variables. Thus, as t increases the number of parameters of the conditional distribution grows with t. Since the number of variables in the sequence, T, may be very large, the amount of memory space required for the

computation of the joint probability distribution will become intractable. Thus, we need a method of limiting the memory of such a computation.

A Markov chain is a mathematical model that describes the evolution of a stochastic process where the time domain is considered to be discrete. The generalization of this model to continuous-time is called a Markov process. Both of these models make the characteristic assumption that the conditional probability distribution of y_t depends only on the most recent variable in the sequence y_{t-1} and independent of all previous variables. This assumption is called the Markov property and assuming that our sequence has the Markov property, we can factorize the joint probability distribution as shown in equation 2.2.

$$p(y_1, \ldots, y_T) = \prod_{t=1}^{T} p(y_t | y_{t-1}) \tag{2.2}$$

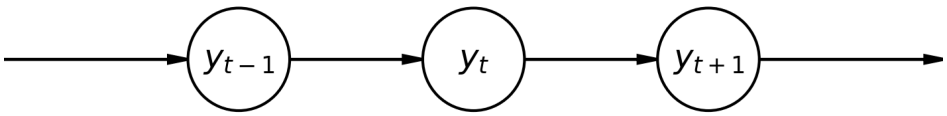

Figure 2.1. First order Markov chain as a graphical model.

Equation 2.2 is said to define a first-order Markov chain which is shown as a graphical model in figure 2.1. The Markov property can be generalized to include more than one recent observation which are called higher-order Markov chains. While allowing the model to incorporate information about more variables is desirable since there can be longer term trends and seasonality in the data set, this information comes with a significant computational cost.

An important distinction to be made is whether the conditional distributions

$p(y_t|y_{t-1})$ depends on the value of t. This argument is related to the time series having a stationary or non-stationary distribution. If the parameters of the conditional distribution $p(y_t|y_{t-1})$ is assumed to be constant in time, this is called a time-homogeneous Markov chain and it corresponds to a stationary time series. A time-inhomogeneous Markov chain will depend on the values of y_t, y_{t-1} and also t which corresponds to a non-stationary time series.

2.2.1. Stochastic Matrices

Let's first study the properties of time-homogeneous Markov chains for discrete-valued variables with a sample space of N distinct values. In such a context, each value is generally referred to as a state and consequently the sample space is called the state space. Each factor $p(y_t = j|y_{t-1} = i)$ in equation 2.2 can be defined as a transition probability since it is the probability of transitioning to state j at time t given that you are at state i at time $t-1$. Since the factors are conditional probability distributions on states, each factor must necessarily add up to 1.

$$\sum_{j=1}^{N} p(y_t = j|y_{t-1} = i) = 1 \tag{2.3}$$

The structure that the Markov assumption imposes on the transition probabilities can be efficiently described as a matrix. Since each transition probability is a function of two state variables and both state variables share a common set of N values, all information related to the conditional probability distribution can be encoded as a $N \times N$ square matrix. Each element a_{ij} of the transition probability matrix $\mathbf{A}$, represents the probability of moving from state i to state j such that the sum of all rows equal to 1 and all elements a_{ij} are non-negative. Matrices that exert this property are called stochastic matrices.

$$\mathbf{A} = \begin{bmatrix} a_{11} & \dots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \dots & a_{NN} \end{bmatrix} \tag{2.4}$$

such that

$$\sum_{j=1}^{N} a_{ij} = 1 \qquad a_{ij} \geq 0 \tag{2.5}$$

The matrix notation of conditional probabilities enable us to make some interesting calculations with simple matrix operations. For example, the nth power of a transition probability matrix yields the probability of moving from state i to state j in n-steps. Let's show this property for $n = 2$. The product $\mathbf{AA}$ can be expressed by defining each of it's elements $(\mathbf{AA})_{ij}$, which can be computed as:

$$(\mathbf{AA})_{ij} = \sum_{l} a_{il} a_{lj} \tag{2.6}$$

Let's write each element of the matrix in it's probability form:

$$\begin{aligned}\sum_l a_{il}a_{lj} &= \sum_l p(X_l|X_i)p(X_j|X_l) \\ &= \sum_l p(X_j|X_l)p(X_l|X_i)\end{aligned} \tag{2.7}$$

Notice that this operation corresponds to summing the probabilities of moving from a chosen state i to some variable state l and then moving from state l to state j. Summing over l means that we are considering any path that may lead to state j from state i in 2-steps.

While defining the transition matrix $\mathbf{A}$ describes the evolution of the process, the marginal distribution of the initial state of the process, $p(y_1)$, has to also be defined for a complete description. For notational convenience, the initial probability distribution on states is generally defined as a set π where π_i corresponds to the probability that the Markov chain starts on state i. Furthermore, since we are describing the transition probabilities as a matrix, the most natural way to represent π is as a vector. This way, we can calculate the probability of any state i at any point in the sequence. For example, the product $\pi\mathbf{A}$ is a vector such that each element corresponds to the probability of a state at the second variable in the sequence.

We can generalize the marginal distribution notation of the initial variable to include the marginal distribution of each variable in the sequence by adding a superscript 1 to the initial state distribution. Thus, $\pi^{(1)}$ is the initial probability distribution. Therefore, the following holds for any Markov chain process:

$$\pi^{(t+1)} = \pi^{(t)}\mathbf{A} \tag{2.8}$$

An important property that some Markov chains possess is that there exists a distribution π^* such that:

$$\pi^* = \pi^* \mathbf{A} \tag{2.9}$$

where π^* is called the stationary distribution. The process is said to converge to a distribution which means that once the process enters this distribution, the chain will remain in the same distribution indefinitely. This property plays a central role in popular approximate inference methods based on Markov chains [24].

2.2.2. Autoregressive Model

For discrete time systems, one of the simplest choice of representations is the autoregressive (AR) model [25], which relates an observation y_t at time t, to a linear combination of the past l observations $(y_{t-l}, \ldots, y_{t-1})$ where l is used to denote the number of time lags. This model is simply referred as $AR(l)$ and can be shown as:

$$y_t = \mathcal{C} + \sum_{i=1}^{l} w_i y_{t-i} + \epsilon_t \tag{2.10}$$

where $\mathcal{C}$ is a constant term, w_i are weights associated with each past observation and ϵ_t is a noise term which expresses the dynamics of the system that our chosen model cannot express. These noise terms are called residuals in the statistics literature, and can be utilized to measure how well our model fits the associated dynamical system. Naturally, we would like to produce a model which has noise terms that are close to zero. However, we encounter the problem of overfitting if we do not account for the

intrinsic noise the system may have and our model will not be able to generalize well. Furthermore, having small residuals does not translate into having a good predictive performance.

Notice that this is a Markov model with order l and each observation y_t has a Gaussian distribution with a mean as a linear function of $y_{t-l:t-1}$. Let's give an example from the task of predicting stock prices. If we use an $AR(1)$ model with the observation variables as the prices of a stock, we obtain Figure 2.2. Initially, the algorithm seems to perform well and using an error metric such as mean squared error gives a small predictive error. However, upon closer inspection the algorithm is predicting that today's closing price will be yesterday's closing price and this does not have any significant predictive power. Increasing the lag parameter will result in obtaining a smoothed average of past prices. While technical financial analyst frequently use these smoothed average prices to detect long term patterns in price movement, the problem is that we cannot meaningfully measure the predictive performance of our models when the observed variable is stock prices.

Instead of using prices, we can instead transform the problem into using the returns of stocks where returns, r_t are defined as:

$$r_t = \frac{y_t - y_{t-1}}{y_{t-1}} \tag{2.11}$$

where y_t is the stock price at time t. The random walk hypothesis famously suggests that stock returns form a random walk or a pure stochastic process which means that they are completely random and thus cannot be predicted [3]. However, more recent work presents evidence that there are trends in stock returns and prediction is in fact possible [26]. Let's demonstrate how prediction becomes harder in the return space by using a $AR(1)$ model on stock returns. Figure 2.3 shows that stock returns

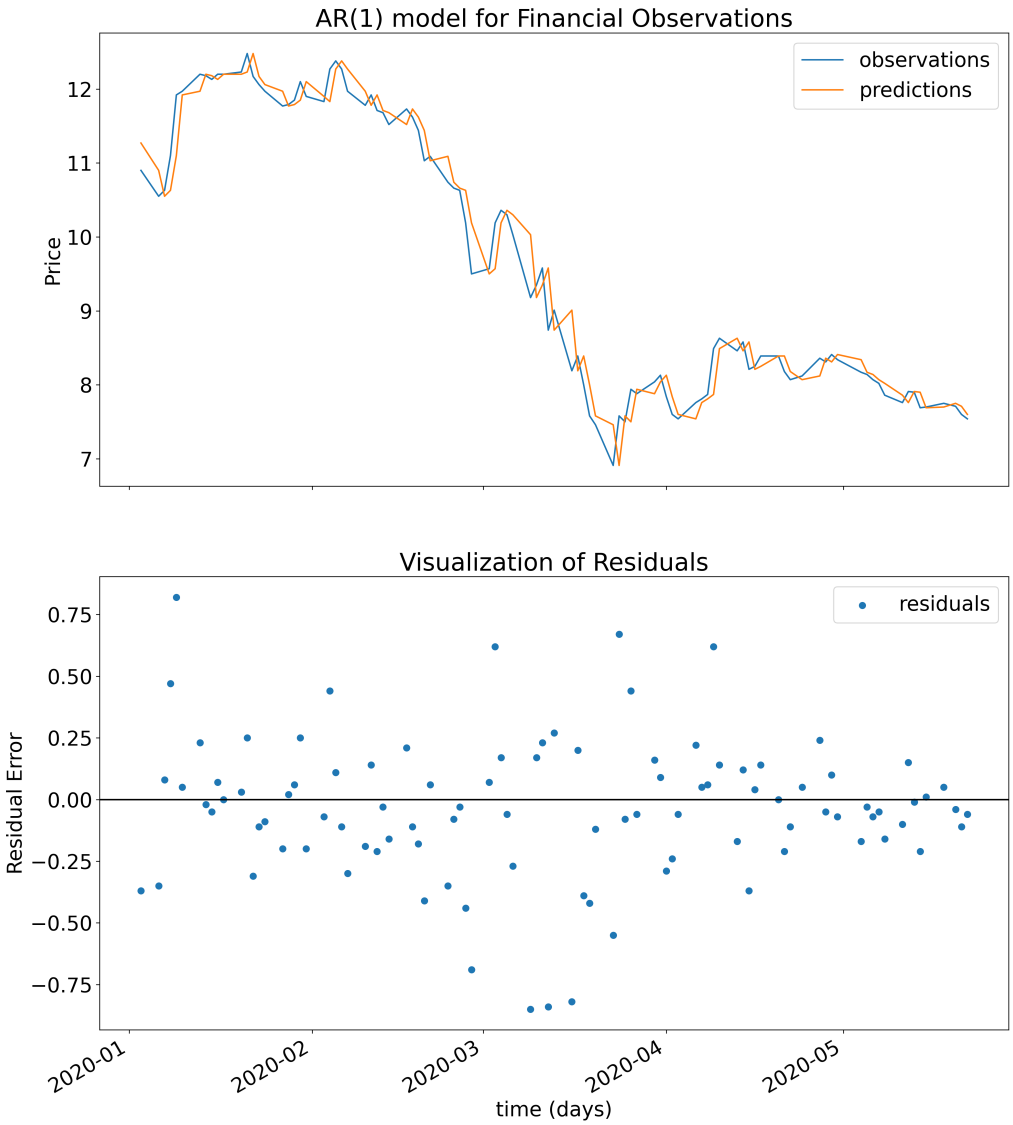

Figure 2.2. AR(1) model on stock prices.

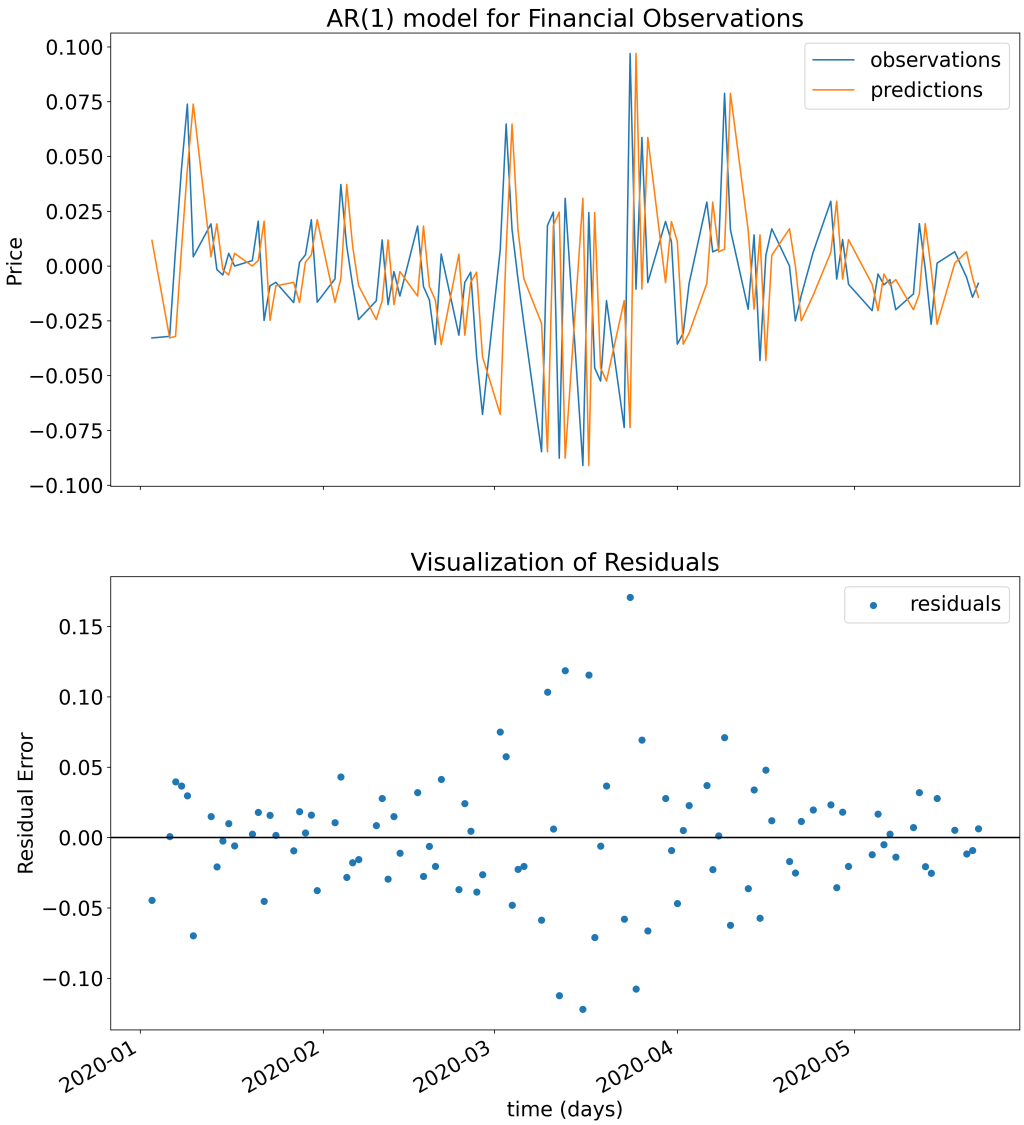

Figure 2.3. AR(1) model on stock returns.

seem like random noise at first glance and the predictive power of our model has decreased significantly. Thus, we have established that while Markov models define a simple and explainable framework to analyze sequences, they are somewhat limited in application.

There is an extensive literature built upon the AR model and a similar variant moving-average (MA) model which share the Markov property. These models include the autoregressive-moving-average (ARMA) model and the autoregressive integrated-moving-average (ARIMA) model which lay the foundation of time series forecasting in statistics and econometrics and are discussed in detail in [25] and [27].

Up to this point, we have discussed the properties of the AR model when there is only a single stream of measurements which is formally called a univariate time series. The only information available to us in such a model is the values of past measurements. Introducing multiple related temporal variables to our model can be a useful addition since it will also inform us about the correlations between these variables. These are called multivariate models and the extension of the univariate AR model to multivariate series is called the vector autoregressive (VAR) model. The interested reader can study the technical details of VAR models in [28] and their application in econometrics in [29].

2.3. Latent Variable Models

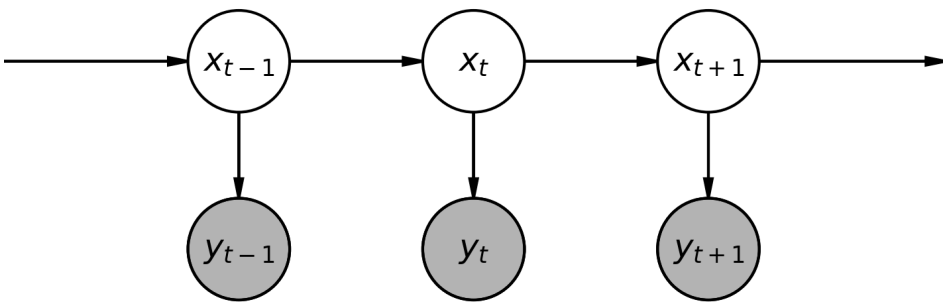

Figure 2.4. Graphical model of a state-space representation.

An extension to the simple models that have been discussed is to incorporate unobservable or latent (hidden) variables into our representation of the dynamical system. These latent variables may correspond to meaningful features of the system that cannot be directly measured or they can be used to find patterns in the data set and used for the purpose of dimensionality reduction. These type of representations are called latent variable models. In the case of i.i.d. data sets, mixture models that are used to identify subpopulations in the data set are well-known examples of latent variable models. We will briefly introduce mixture models in section 2.4.

A special latent variable model for time series data is called the state-space model where the representation associated with the system process is differentiated from the model associated with our measurements. The latent variables in such a model correspond to the unobservable values the process actually takes and the observed variables are the noisy measurements of the process.

For continuous time systems, the state-space model consists of a set of first order ordinary differential equations. For discrete time systems, these differential equations can be simplified into recursive relations. A discrete state-space model consists of two separate models: (1) state transition model and (2) observation likelihood model. Let's denote a measurement and state at time t as y_t and x_t. The general form of a state-space model can be shown as:

$$x_t = f(x_{1:t-1}, \epsilon_t, \theta) \tag{2.12}$$

$$y_t = g(x_{1:t}, \nu_t, \theta) \tag{2.13}$$

where f is the state transition function, g is the observation likelihood function and θ denotes all possible parameters of both models. The variable ϵ_t and ν_t are random variables that represents the dynamics of the state transition that our model

is not able to express and the measurement error, respectively. Note that both x_t and y_t may be multivariate variables, which means that multiple processes and related observations may be represented in this model. The state-space representation will be the foundation for our modelling considerations in this work.

Notice that by introducing a latent first-order Markov state transition process that emits the observations, we are able to encode the dependence of all observations $y_{1:T}$ without explicitly using the observations. The simplest way to see this dependence is to use the d-separation concept on figure 2.4 where observed variables are shaded for emphasis. A review on how to represent time series models as graphical models is presented in [30]. In such a model, we assume that observations $y_{1:T}$ are conditionally independent given states $x_{1:T}$. However, since the states $x_{1:T}$ are unobserved, this relationship defines a dependence between all observed values. The state-space model in this case is:

$$x_t = f(x_{t-1}, \epsilon_t, \theta) \tag{2.14}$$

$$y_t = g(x_t, \nu_t, \theta) \tag{2.15}$$

Using a state-space model for stock returns explicitly assumes that there is a noise associated with our observations of the system. The efficient market hypothesis famously states that stock prices reflect all information related to the valuation of a stock [31]. This suggests that the system does not have any noise associated with the pricing of assets. However, parallel to the view that the value of an asset is determined by how much investors seek to buy or sell the asset, we assume that investors are not completely rational and have personal biases. Thus, stock prices also reflect these behavioral signals which can be interpreted as noise. The idea that markets are not efficient in reality is discussed in detail in [32].

2.4. Clustering

For complex systems, it is often useful to identify the current behavior of the system in terms of regimes which are distinct characteristic behaviors of the system. By identifying different regimes in the process, we may be able to model non-linear dynamics with multiple stationary linear models. This task can be treated as a clustering analysis for time series models.

Clustering is simply the process of grouping unlabeled observations and is one of the main sub-fields of machine learning. Some of the widely used models that are being used for this task are the k-means clustering [33], mixture models [34] and t-distributed stochastic neighbor embedding [35]. Mixture models are structurally similar to state-space models introduced in section 2.3 except that there is no dependence between the latent variables.

The simplest mixture model is the Gaussian mixture model (GMM) where each data point is distributed as a Gaussian that is defined by parameters of the mixture component it belongs. Using such a model, it is possible to identify multiple Gaussian components in the data set. As the number of mixture components increase, we are able to approximate many non-Gaussian distributions with such a model. However, the GMM and most clustering models assume that the observations are conditionally independent given the parameters and the order of observations are irrelevant. Such an assumption is not valid for sequential data sets.

For financial data sets, an empirical remark made in [36] states that the magnitude of the variation in subsequent observations tend to cluster. This is called volatility clustering and it has led to the research of mathematical models that are able to handle variation of dispersion in the data set. Classical regression models assume that the modelling errors have the same magnitude of variance and data that displays such property is said to be homoscedastic. However, since empirically we see that financial data is heteroscedastic, which is the absence of homoscedasticity, different modelling

considerations have to be made. Robert Engle was awarded a Nobel Prize in Economics for his autoregressive conditional heteroscedasticity (ARCH) model [37] in 2003 which led to the development of the generalized autoregressive conditional heteroscedasticity (GARCH) [38] and other stochastic volatility models that are prevalent in mathematical finance.

While the models mentioned above model the variance of the random process explicitly, our aim is to find different regimes in the time series that display differences in their magnitude of variation. The simplest way to classify such data sets is to use the state-space model shown in figure 2.4 where the latent states are discrete-valued variables that define the current regime.

2.5. Hidden Markov Model

The hidden Markov model (HMM) is a particular class of the state-space model shown in figure 2.4 where the latent variables are discrete-valued. The model assumes that an unobserved first-order Markov process generates observations from a mixture distribution $p(y_t|x_t)$. Hence, the HMM can be interpreted as an extension of mixture models where sequential information is also encoded [9]. HMMs are frequently applied in portfolio optimization [39] and credit risk modelling [40] in computational finance, statistical speech recognition [41], protein folding [42] and many more fields.

In order to model the returns of financial assets, the simplest distribution we can use is the Gaussian distribution which yields the Gaussian HMM. Let's assume that we have a N-state system. The state at time t, defined as x_t, has an evolution process which is assumed to be a discrete state, discrete time first order Markov process. Thus, we can represent this process as a categorical distribution where the parameters of the distribution depend on the value assumed by the previous state x_{t-1}.

$$f(x_t|x_{t-1}, A) = \prod_{i=1}^{N}\prod_{j=1}^{N} a_{ij}^{[x_{t-1}=i][x_t=j]} \tag{2.16}$$

where a_{ij} is the ith row and j^{th} column of a transition matrix A. The notation $[P]$ is the Iverson bracket where $[P]$ equals 1 if P is true and 0 otherwise. The initial probabilities of states at time 1 is parameterized as a vector π, where the density is defined as $p(x_1|\pi)$.

An observation at time t, defined as y_t, is assumed to be conditionally independent of previous observations $y_{1:t-1} \triangleq (y_1, \ldots, y_{t-1})$ given the current state x_t and has a multivariate Gaussian distribution with mean μ and variance Σ. The corresponding model can be shown as:

$$\begin{aligned} y_t|x_t, \mu, \sigma &\sim \text{Gaussian}\big(\mu, \Sigma\big) \\ &= \frac{1}{\sqrt{(2\pi)^k|\Sigma|}}\exp\Big(-\frac{1}{2}(y_t-\mu)^T\Sigma^{-1}(y_t-\mu)\Big) \end{aligned} \tag{2.17}$$

When $x_t = i$ the state dependent parameters are shown with the subscript i as μ_i and Σ_i. Using the model assumptions we have defined, the joint probability distribution of the observed variables $y_{1:T}$ and latent variables $x_{1:T}$ is defined as:

$$p(y_{1:T}, x_{1:T}|\theta) = p(x_1|\pi)\Big[\prod_{t=2}^{T} p(x_t|x_{t-1}|A)\Big]\prod_{t=1}^{T} p(y_t|x_t, \mu, \Sigma) \tag{2.18}$$

2.5.1. Inference

Our main goal is to infer the posterior probability distribution of the state of the system at each time t. As a consequence of the sequential structure of time series data, there is a distinction to be made between offline and online analysis. When analyzing a system, if the measurements $y_{1:T}$ related to the evolution of the system have been completely collected, an offline analysis is carried out. This means that all information related to the system is available at the time of the analysis. While this approach generally results in better estimates of system behavior, it is computationally expensive which can become a problem for time-sensitive decision making. On the other hand, analyzing a system in real-time is called an online analysis, where only a subset of the measurements, $(y_1, \ldots, y_t)$ where $t < T$, are collected. Contrary to offline analysis, online analysis generally requires less computation while trading-off estimation accuracy. The choice of carrying out an online or offline analysis depends on the application, and both types are frequently used in conjunction.

We formulate two common inference problems for the estimation of latent variables in our model using a recursive Bayesian framework. The first problem, referred to as filtering, is the online estimation of the latent variables which corresponds to calculating the probability density $p(x_t|y_{1:t}, \theta)$. The second problem, referred to as smoothing, is the estimation of the latent variables given all evidence up to time T, which corresponds to computing $p(x_t|y_{1:T}, \theta)$.

In order to demonstrate a recursive framework, let's assume that the posterior distribution of the state x_{t-1} at time $t-1$ is known. Using the state evolution equation $f(x_t|x_{t-1}, A)$ we can obtain a prior distribution on the state x_t. This procedure is referred to as the prediction step in the filtering literature and is shown in equation 2.19.

$$p(x_t|y_{1:t-1},\theta) = \sum_{x_{t-1}} f(x_t|x_{t-1},A)p(x_{t-1}|y_{1:t-1},\theta) \tag{2.19}$$

When an observation y_t is measured, we can apply the Bayes' theorem in order to update our prior distribution on the state x_t, which is referred to as the correction step in the filtering literature. The joint posterior distribution on the latent variable x_t is shown in equation 2.20.

$$p(x_t|y_{1:t},\theta) = \frac{p(y_t|x_t,\mu,\Sigma)p(x_t|y_{1:t-1},\theta)}{p(y_t|y_{1:t-1},\theta)} \tag{2.20}$$

Notice that the denominator $p(y_t|y_{1:t-1},\theta)$ is obtained by marginalizing x_t out from the nominator in equation 2.20. A final remark that will lead to an efficient filtering algorithm is that since $y_{1:T}$ are observed variables, instead of calculating the posterior distribution of the state x_t, we can calculate the joint probability distribution $p(x_t, y_{1:t})$. This joint probability distribution can be easily normalized at any point by marginalizing x_t and calculating the normalization constant. The resulting algorithm is called the forward algorithm.

2.5.1.1. Forward Algorithm. Assuming that the parameters π which is the initial probability distribution of the state x_1, A which is the transition matrix, μ which contain the mean vectors μ_i of each state i and Σ which contain the covariance matrix Σ_i of each state i of the HMM is known, we can find an updated distribution over x_1 when we observe y_1 using equation 2.21. We will refer to the parameters of the model as a set $\theta = (\pi, A, \mu, \Sigma)$ for notational convenience.

$$p(x_1, y_1|\theta) = p(y_1|x_1, \theta)p(x_1|\theta) \tag{2.21}$$

The joint distribution $p(x_1, y_1|\theta)$ is referred to as $\alpha_1(x_1)$ in the HMM literature. Using the same logic as equation 2.19, we can obtain a prior distribution over state x_2.

$$p(x_2, y_1|\theta) = \sum_{x_1} f(x_2|x_1, A)\alpha_1(x_1) \tag{2.22}$$

The updated joint distribution $\alpha_2(x_2)$ is obtained using:

$$p(x_2, y_{1:2}|\theta) = p(y_2|x_2, \theta)p(x_2, y_1|\theta) \tag{2.23}$$

It is trivial to generalize this algorithm to any time t and the prediction and update equations of the forward algorithm are shown in equations 2.24 and 2.25, respectively.

$$p(x_t, y_{1:t-1}|\theta) = \sum_{x_{t-1}} f(x_t|x_{t-1}, A)\alpha_{t-1}(x_{t-1}) \tag{2.24}$$

$$\begin{aligned} p(x_t, y_{1:t}|\theta) &= p(y_t|x_t, \theta)p(x_t, y_{1:t-1}|\theta) \\ &= \alpha_t(x_t) \end{aligned} \tag{2.25}$$

Notice that we can calculate the posterior distribution of any state x_t by marginalizing $\alpha_t(x_t)$ over x_t, and dividing $\alpha_t(x_t)$ by this normalization constant.

$$p(y_{1:t}|\theta) = \sum_{x_t} p(x_t, y_{1:t}|\theta) \tag{2.26}$$

$$p(x_t|y_{1:t}, \theta) = \frac{p(x_t, y_{1:t}|\theta)}{p(y_{1:t}|\theta)} \tag{2.27}$$

Now, let's discuss how we can have better estimates of a state x_t in the case where all observations $y_{1:T}$ are measured. Thus, our goal is to obtain a smoothed estimate of the latent variables $x_{1:T}$ which utilizes the full information available at time T. While the estimation framework shown above was suitable for an online procedure, the purpose of calculating a smoothed estimate defined as $p(x_t|y_{1:T})$, is generally for parameter estimation.

Notice that using the chain rule of probability, we can use the filtered estimates $p(x_t, y_{1:t}|\theta)$ together with a density $p(y_{t+1:T}|x_t)$ in order to calculate the joint probability density of a state x_t that utilizes the complete sequence of information.

$$p(x_t, y_{1:T}|\theta) = p(y_{t+1:T}|x_t, \theta)p(x_t, y_{1:t}|\theta) \tag{2.28}$$

The joint density $p(x_t, y_{1:T}|\theta)$ can be normalized by using the incomplete data likelihood $p(y_{1:T}|\theta)$ in order to obtain the smoothed estimate $p(x_t|y_{1:T}, \theta)$.

2.5.1.2. Backward Algorithm. Similar to the forward algorithm, we define a recursive algorithm for obtaining the densities $p(y_{t+1:T}|x_t, \theta)$ which we will use to calculate the smoothed estimate of the latent variables. However, this time the recursions start from the last data point T. The algorithm is made up of a two steps: update and postdiction.

Assuming we know $p(y_{t+1:T}|x_t, \theta)$, we can find a joint distribution $p(y_{t:T}|x_{t-1}, \theta)$ by initially using the emission density $p(y_t|x_t, \theta)$ in order to find the likelihood of $y_{t:T}$ for each state x_t. This step can be called update since it involves the addition of an observation y_t to a known joint probability distribution.

$$p(y_{t:T}|x_t, \theta) = p(y_t|x_t, \theta)p(y_{t+1:T}|x_t, \theta) \tag{2.29}$$

The next step is to estimate the likelihood of $y_{t:T}$ given that we only know state x_{t-1}. This estimate can be obtained by a marginalization procedure and is called postdiction since it involves the estimation of the likelihood of a state with information from the future.

$$p(y_{t:T}|x_{t-1},\theta) = \sum_{x_t} p(y_{t:T}|x_t,\theta) f(x_t|x_{t-1},A) \tag{2.30}$$

In the HMM literature the conditional joint density $p(y_{t+1:T}|x_t,\theta)$ is denoted as $\beta_t(x_t)$. The backward algorithm is initialized by setting $\beta_T(x_T)$ as a vector of ones.

2.5.1.3. Forward-Backward Algorithm. After we obtain $\alpha_t(x_t)$ and $\beta_t(x_t)$ for all values of t, the calculation of $p(x_t|y_{1:T},\theta)$ is straightforward. The smoothed estimate is denoted as $\gamma_t(x_t)$ in the HMM literature.

$$\gamma_t(x_t) = \frac{\alpha_t(x_t)\beta_t(x_t)}{\sum_{x_t} \alpha_t(x_t)\beta_t(x_t)} \tag{2.31}$$

2.5.2. Learning

In general, the maximum likelihood estimation of parameters is the simplest case of parameter learning when we are able to observe the outcomes of every variable in the model. However, this is not the case for a state-space model where we assume that an unobservable process is generating the data that we observe. Thus, in order to find a maximum likelihood estimate for the parameters of a state-space model, we will have to consider the maximization of the likelihood of only the observed variables. The incomplete data likelihood for a model can be expressed as the marginal of the complete data likelihood which the functional form is equivalent to the joint probability distribution of the observed variables $y_{1:T}$ and the latent variables $x_{1:T}$.

$$p(y_{1:T}|\theta) = \sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta) \tag{2.32}$$

where θ is the set of parameters of the model. Notice that in the case of parameter estimation, $p(y_{1:T}|\theta)$ is a function of θ and $y_{1:T}$ are given. Thus, this function is frequently shown as $L(\theta)$ where L stands for likelihood function. The goal of maximum likelihood estimation is to find the parameters θ that maximize the probability $p(y_{1:T}|\theta)$. The framework we will use to solve this problem is called the expectation maximization (EM) algorithm which is a maximum likelihood estimation method for the parameters of probabilistic models with incomplete data or latent variables.

We can express the problem as:

$$\theta^* = \underset{\theta}{\text{argmax}} \quad p(y_{1:T}|\theta) \tag{2.33}$$

In the case where the data is completely observed, the likelihood function, which is expressed as $p(y_{1:T}, x_{1:T}|\theta)$, has a single global maximum. Therefore, the task of finding a maximum likelihood estimate is relatively easy. However, when we have incomplete data, since we are considering different configurations of $x_{1:T}$ the incomplete data likelihood $p(y_{1:T}|\theta)$ is generally a multimodal function which means that the optimization problem is non-convex. This is due to the fact that the unknown configuration of $x_{1:T}$ introduces new degrees of freedom into the system. This problem gets worse as the number of variables in the sequence, T, increase and it is also increasingly hard to analytically calculate the actual incomplete data likelihood as a function of θ. Thus, in order to find θ^*, we need a different framework than calculating an incomplete data likelihood as a function of θ.

Before we describe the EM algorithm, we need to define an inequality theorem that underlies many of the algorithms in statistics and machine learning.

Theorem 2.1 (Jensen's inequality [43]). *If f is a convex function and X is a random variable, then:*

$$E[f(X)] \geq f(E[X])$$

Moreover, if f is strictly convex, then equality implies that $X = E[X]$ with probability 1, i.e., X is a constant. The inequality sign is flipped when the function f is concave.

The first insight into the solution is to notice that θ^* that maximizes $p(y_{1:T}|\theta)$ and $\log p(y_{1:T}|\theta)$ are equal since logarithm is a strictly monotonically increasing function. This property will allow us to use the Jensen's inequality in the next step in order to find a lower bound to the loglikelihood function denoted as $l(\theta)$.

$$\begin{aligned} l(\theta) &= \log \sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta) \\ &= \log \sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta) \frac{q(x_{1:T})}{q(x_{1:T})} \end{aligned} \tag{2.34}$$

where $q(x_{1:T})$ is a joint probability distribution over the variables $x_{1:T}$. Using the Jensen's inequality and noting that logarithm is a concave function, we can move the logarithm inside the summation.

$$l(\theta) \geq \sum_{x_{1:T}} q(x_{1:T}) \log \frac{p(y_{1:T}, x_{1:T}|\theta)}{q(x_{1:T})} \tag{2.35}$$

The equation above holds for any function $q(x_{1:T})$. However, we want to obtain such a $q(x_{1:T})$ that maximizes the lower bound. Let's define the lower bound as $LB(q)$ which is a functional that takes the function $q(x_{1:T})$ as an argument.

$$LB(q) = \sum_{x_{1:T}} q(x_{1:T}) \log p(y_{1:T}, x_{1:T}|\theta) - \sum_{x_{1:T}} q(x_{1:T}) \log q(x_{1:T}) \tag{2.36}$$

which is subject to the constraint:

$$\sum_{x_{1:T}} q(x_{1:T}) = 1 \tag{2.37}$$

We can use Lagrange multipliers to find a local maxima to the lower bound functional subject to the equality constraint. Let's define the objective function as $\Lambda(q, \lambda)$.

$$\Lambda(q, \lambda) = \sum_{x_{1:T}} q(x_{1:T}) \log p(y_{1:T}, x_{1:T}|\theta) - \sum_{x_{1:T}} q(x_{1:T}) \log q(x_{1:T}) + \lambda\big(1 - \sum_{x_{1:T}} q(x_{1:T})\big) \tag{2.38}$$

Taking the functional derivative of Λ with respect to q and setting it to zero will result in finding the function q that maximizes the lower bound.

$$\begin{aligned}\frac{\delta\Lambda}{\delta q} &= \log p(y_{1:T}, x_{1:T}|\theta) - \big(1 + \log q(x_{1:T})\big) - \lambda \\ &= 0\end{aligned} \tag{2.39}$$

Rearranging the variables will result in:

$$\begin{aligned}\log q(x_{1:T}) &= \log p(y_{1:T}, x_{1:T}|\theta) - (1 + \lambda) \\ q(x_{1:T}) &= p(y_{1:T}, x_{1:T}|\theta) e^{-(1+\lambda)}\end{aligned} \tag{2.40}$$

From the constraint on $q(x_{1:T})$, we can calculate the Lagrange multiplier λ:

$$\begin{aligned}\sum_{x_{1:T}} q(x_{1:T}) &= e^{-(1+\lambda)} \sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta) \\ 1 &= e^{-(1+\lambda)} \sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta)\end{aligned} \tag{2.41}$$

The factor $e^{-(1+\lambda)}$ must equal to the inverse of the summation $\sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta)$ since their product equals 1. Then we can express $q(x_{1:T})$ as:

$$\begin{aligned}q(x_{1:T}) &= \frac{p(y_{1:T}, x_{1:T}|\theta)}{\sum_{x_{1:T}} p(y_{1:T}, x_{1:T}|\theta)} \\ &= \frac{p(y_{1:T}, x_{1:T}|\theta)}{p(y_{1:T}|\theta)} \\ &= p(x_{1:T}|y_{1:T}, \theta)\end{aligned} \tag{2.42}$$

We see that the probability distribution $q(x_{1:T})$ that maximizes the lower bound is $p(x_{1:T}|y_{1:T},\theta)$. Let's return to the expectation maximization algorithm. We have derived that the loglikelihood of the incomplete data is lower bounded by the expectation:

$$\begin{aligned} \log p(y_{1:T}|\theta) &\geq \sum_{x_{1:T}} p(x_{1:T}|y_{1:T},\theta)\log\frac{p(y_{1:T},x_{1:T}|\theta)}{p(x_{1:T}|y_{1:T},\theta)} \\ &= \mathbb{E}_{p(x_{1:T}|y_{1:T},\theta)}\left[\log\frac{p(y_{1:T},x_{1:T}|\theta)}{p(x_{1:T}|y_{1:T},\theta)}\right] \end{aligned} \tag{2.43}$$

The appropriately named expectation maximization algorithm is a method of maximizing the expectation in equation (2.43) in an iterative manner. Notice that the expectation is over the posterior distribution of the latent variables, $p(x_{1:T}|y_{1:T},\theta)$, which is a function of θ. However, in order to calculate the expectation, we need to know the numerical value of this posterior distribution. As a consequence, we use an iterative approach where we first initialize the parameters θ and denote it as θ^{old}. Then we maximize the expectation with respect to θ while keeping the old parameters θ^{old} constant. Let us call the parameters that maximize the expectation $\theta^{'}$. The next step is to calculate $p(x_{1:T}|y_{1:T},\theta^{'})$ and repeat the procedure. This way, we can iteratively find the parameters θ^{*} that maximize the incomplete data likelihood $p(y_{1:T}|\theta^{*})$.

The initial values of the parameters θ is quite important in terms of the speed of convergence to a local maxima for the EM algorithm. Converge properties of the algorithm is discussed in [44]. A practical approach of initializing the parameter set is to assume that the observed data is i.i.d. and use a well known clustering algorithm such as the k-means clustering which will yield an informed starting point.

We can simplify the maximization procedure by noticing that we do not have to

calculate the expectation fully. We will omit the probability measure $p(x_{1:T}|y_{1:T}, \theta^{old})$ in our notation. Let's write out the expectation:

$$\begin{aligned}\mathbb{E}\left[\log\frac{p(y_{1:T}, x_{1:T}|\theta)}{p(x_{1:T}|y_{1:T}, \theta^{old})}\right] &= \sum_{x_{1:T}} p(x_{1:T}|y_{1:T}, \theta^{old})\log p(y_{1:T}, x_{1:T}|\theta) \\ &- \sum_{x_{1:T}} p(x_{1:T}|y_{1:T}, \theta^{old})\log p(x_{1:T}|y_{1:T}, \theta^{old})\end{aligned} \tag{2.44}$$

Notice that the second summation does not depend on θ, therefore we can discard it for the maximization problem. The first summation is generally referred to as $Q(\theta, \theta^{old})$ in the literature. Thus, the problem becomes:

$$\begin{aligned}\theta' &= \underset{\theta}{\text{argmax}} \quad \sum_{x_{1:T}} p(x_{1:T}|y_{1:T}, \theta^{old})\log p(y_{1:T}, x_{1:T}|\theta) \\ &= \underset{\theta}{\text{argmax}} \quad Q(\theta, \theta^{old})\end{aligned} \tag{2.45}$$

Notice that we have not made any assumptions on the functional form of the likelihood $p(y_{1:T}, x_{1:T}|\theta)$ or the distribution $p(x_{1:T}|y_{1:T}, \theta^{old})$. These results carry over to any parameter estimation problem with incomplete data or latent variables. However, analytically calculating θ' is possible in only a handful of cases where taking the derivative of $Q(\theta, \theta^{old})$ is relatively simple. For intractable functions, one method of maximization is called the Monte Carlo expectation-maximization where the function is approximated using a sampling based strategy [45].

2.5.2.1. The Baum-Welch Algorithm. The Baum–Welch algorithm is a special case of the EM algorithm where the model is assumed to be a hidden Markov model. In this case, the Baum-Welch algorithm uses the forward–backward algorithm for the

calculation of $p(x_{1:T}|y_{1:T},\theta)$ which is necessary in order to define $Q(\theta,\theta^{old})$.

For a hidden Markov model, the set of parameters is $\theta=(\pi,A,\mu,\Sigma)$ where π is the initial state probabilities, A is the set of parameters of the state evolution density, μ and Σ are the set of parameters of the observation density. Let's recall the joint probability distribution of $x_{1:T}$ and $y_{1:T}$ defined by the HMM.

$$p(y_{1:T},x_{1:T}|\theta)=p(x_1|\pi)\Big[\prod_{t=2}^{T}p(x_t|x_{t-1}|A)\Big]\prod_{t=1}^{T}p(y_t|x_t,\mu,\Sigma) \tag{2.46}$$

In order to show how the forward-backward algorithm relates to the calculation of $p(x_{1:T}|y_{1:T},\theta)$, let's write out $Q(\theta,\theta^{old})$ with the factorized form of $p(y_{1:T},x_{1:T}|\theta)$.

$$\begin{aligned}
Q(\theta,\theta^{old}) &= \sum_{x_{1:T}} p(x_{1:T}|y_{1:T},\theta^{old})\log\Big(p(x_1|\pi)\Big[\prod_{t=2}^{T}p(x_t|x_{t-1},A)\Big]\prod_{t=1}^{T}p(y_t|x_t,\mu,\Sigma)\Big)\\
&= \sum_{x_{1:T}} p(x_{1:T}|y_{1:T},\theta^{old})\log p(x_1|\pi)\\
&+ \sum_{x_{1:T}} p(x_{1:T}|y_{1:T},\theta^{old})\log\Big[\prod_{t=2}^{T}p(x_t|x_{t-1},A)\Big]\\
&+ \sum_{x_{1:T}} p(x_{1:T}|y_{1:T},\theta^{old})\log\prod_{t=1}^{T}p(y_t|x_t,\mu,\Sigma)
\end{aligned} \tag{2.47}$$

Notice that the parameters π, A, and μ, Σ are separated in the factorized form. This is a convenient result since we can independently find optimal values for these parameters. The last part of the algorithm is to calculate $p(x_{1:T}|y_{1:T},\theta)$ using the forward-backward algorithm.

Let's first show how to calculate the sum $\sum_{x_{1:T}} p(x_{1:T}|y_{1:T}, \theta^{old})\log p(x_1|\pi)$. The general product rule of conditional probabilities lets us write the following factorization for $p(x_{1:T}|y_{1:T}, \theta^{old})$:

$$p(x_{1:T}|y_{1:T}, \theta^{old}) = p(x_{2:T}|x_1, y_{1:T}, \theta^{old})p(x_1|y_{1:T}, \theta^{old}) \tag{2.48}$$

Using this factorization we can write the sum as:

$$\begin{aligned}\sum_{x_{1:T}} p(x_{1:T}|y_{1:T}, \theta^{old})\log p(x_1|\pi) &= \sum_{x_{2:T}} p(x_{2:T}|x_1, y_{1:T}, \theta^{old}) \sum_{x_1} p(x_1|y_{1:T}, \theta^{old})\log p(x_1|\pi) \\ &= \sum_{x_1} p(x_1|y_{1:T}, \theta^{old})\log p(x_1|\pi)\end{aligned} \tag{2.49}$$

where $p(x_1|y_{1:T}, \theta^{old})$ is the smoothed posterior distribution of x_1, referred to as $\gamma(x_1)$ that we obtain when we use the forward-backward algorithm.

Using a similar factorization scheme, we obtain the following factorized form for $Q(\theta, \theta^{old})$.

$$\begin{aligned}Q(\theta, \theta^{old}) = &\sum_{x_1} p(x_1|y_{1:T}, \theta^{old})\log p(x_1|\pi) \\ &+ \sum_{t=2}^{T} \sum_{x_t} \sum_{x_{t-1}} p(x_t, x_{t-1}|y_{1:T}, \theta^{old})\log p(x_t|x_{t-1}, A) \\ &+ \sum_{t=1}^{T} \sum_{x_t} p(x_t|y_{1:T}, \theta^{old})\log p(y_t|x_t, \mu, \Sigma)\end{aligned} \tag{2.50}$$

where the only unknown left is $p(x_t, x_{t-1}|y_{1:T}, \theta^{old})$. This quantity can also be calculated using the $\alpha_t(x_t)$ values calculated during the forward run and $\beta_t(x_t)$ values calculated during the backward run.

$$\begin{aligned} p(x_t, x_{t-1}|y_{1:T}, \theta^{old}) &= \frac{p(y_{t+1:T}|x_t)p(y_t|x_t)p(x_t|x_{t-1})p(x_{t-1}, y_{1:t-1})}{p(y_{1:T})} \\ &= \frac{\beta_t(x_t)p(y_t|x_t)p(x_t|x_{t-1})\alpha_{t-1}(x_{t-1})}{p(y_{1:T})} \end{aligned} \tag{2.51}$$

This quantity is defined is $\xi_{t-1}(x_{t-1}, x_t)$ in the HMM literature. We have shown how the Baum-Welch algorithm uses the forward-backward algorithm to calculate $Q(\theta, \theta^{old})$. Now, we can independently find optimal values for the parameters π, A, μ and Σ. The update equations of each parameter are shown below [34].

$$\pi_i = \frac{\gamma_1(x_1 = i)}{\sum_{j=1}^{N} \gamma_1(x_1 = j)} \tag{2.52}$$

$$A_{ij} = \frac{\sum_{t=2}^{T} \xi_{t-1}(x_{t-1} = i, x_t = j)}{\sum_{k=1}^{N} \sum_{t=1}^{T} \gamma_t(x_t = k)} \tag{2.53}$$

$$\mu_i = \frac{\sum_{t=1}^{T} \gamma_t(x_t = i) y_t}{\sum_{j=1}^{N} \sum_{t=1}^{T} \gamma_t(x_t = j)} \tag{2.54}$$

$$\Sigma_i = \frac{\sum_{t=1}^{T} \gamma_t(x_t = i)(y_t - \mu_i)(y_t - \mu_i)^T}{\sum_{j=1}^{N} \sum_{t=1}^{T} \gamma_t(x_t = j)} \tag{2.55}$$

2.5.3. Handling Imbalanced Data Sets

As we have mentioned, the maximization of $p(y_{1:T}|\theta)$ with respect to θ is a non-convex problem. However, the update equations that the Baum-Welch algorithm yield is a locally optimal strategy. Thus, while the expectation maximization algorithm finds a local maxima, it is not guaranteed to find the global maximum. This results in the need for running the algorithm multiple times for each data set and see if the parameters found by the algorithm are similar with respect to some distance measure.

One heuristic way to improve the maximum likelihood solution for financial data is to notice that the data set in question is imbalanced with respect to different regimes. In clustering or classification problems, a class-imbalanced data set can be defined as a data set where the relative frequency of each class or regime differs significantly. Since we are treating each observation y_t as equally informative in the joint likelihood density, this creates an algorithmic bias towards a certain regime because it has more samples. Fitting the data to the regime that has more samples may result in a global maximum, however, it does not represent the real process. This is a common problem encountered in human-centered machine learning tasks, and is called fairness in machine learning. A recent survey on the issue is presented in [46].

In the case of time series data, to the best of our knowledge, the task of handling imbalanced data sets has not been studied as much. The simplest way to introduce some supervision in the learning process is to weight the observations in the likelihood in terms of some measure of importance. The framework for maximum likelihood estimation with weighted samples is presented in [47] and a methodology for time series data is presented in [48].

For financial forecasting, since we have an approximate idea about when financial crises occurred in the past, we give a larger weight to observations during a crisis in order to understand the dynamics of assets during a period of crisis. The joint probability distribution of $x_{1:T}$ and $y_{1:T}$ can be updated as:

$$p(y_{1:T}, x_{1:T}|\theta) = p(x_1|\pi)\Big[\prod_{t=2}^{T} p(x_t|x_{t-1}|A)\Big]\prod_{t=1}^{T}\omega_t p(y_t|x_t, \mu, \Sigma) \tag{2.56}$$

where we have introduced a weight variable ω_t for each observation y_t. It is straightforward to modify the update equations for the joint density shown in equation 2.56.

2.6. Exponential Families

It is useful to describe how different parametric probability distributions relate to each other in terms of their similarities and differences. The exponential family is one such measure of similarity where any probability distribution which can be expressed in the form shown in equation 2.57 is considered to be an exponential family distribution.

$$p(x|\eta) = h(x)\exp\big(\eta^T t(x) - a(\eta)\big) \tag{2.57}$$

The parameters of an exponential family are shown as a vector η, which is also referred to as the natural parameter. $t(x)$ is called sufficient statistic since the maximum likelihood solution of η can be found using only $t(x)$. $h(x)$ is the Lebesgue measure and $a(\eta)$ is called the log normalizer or log partition function. Some examples of probability distributions that are in the exponential family are Bernoulli, Gaussian, Multinomial and Dirichlet distributions.

Let's show how the exponential family form is useful in practical calculations. We start with the question of how to find the normalization constant of a probability distribution. We can define a kernel with any function $f(x)$ that is non-negative and transform it into a probability distribution by normalizing.

$$p(x) = \frac{1}{Z} f(x) \tag{2.58}$$

where

$$Z = \int_x f(x)dx \tag{2.59}$$

The constant term Z is called the normalization constant or the partition function. Now, let's define an alternative parametrization of the exponential family in equation 2.60 where the kernel function is defined in equation 2.61.

$$p(x|\eta) = g(\eta)h(x)\exp\big(\eta^T t(x)\big) \tag{2.60}$$

$$K(x) = h(x)\exp\big(\eta^T t(x)\big) \tag{2.61}$$

The normalization constant for such a kernel function is:

$$Z = \int_x h(x)\exp\big(\eta^T t(x)\big) \tag{2.62}$$

Since a normalized probability distribution must sum to 1, the following deduction can be made:

$$\begin{aligned} 1 &= \int_x g(\eta)h(x)\exp\big(\eta^T t(x)\big) \\ &= g(\eta)\int_x h(x)\exp\big(\eta^T t(x)\big) \\ &= g(\eta)Z \end{aligned} \tag{2.63}$$

This shows that the normalization constant is a part of the parametric form of an exponential family distribution and it can be trivially found. Let's relate the function $g(\eta)$ to the log-normalizer function $a(\eta)$ we have used in our original parametrization. In order to utilize the mathematical properties of exponentials, we might consider moving every function into the exponential as shown in 2.64.

$$p(x|\eta) = \exp\big(\eta^T t(x) + \log h(x) + \log g(\eta)\big) \tag{2.64}$$

When use the identity $g(\eta) = 1/Z$ in equation 2.64 we obtain the original form of the family.

$$\begin{aligned} p(x|\eta) &= \exp\big(\eta^T t(x) + \log h(x) + \log\frac{1}{Z}\big) \\ &= \exp\big(\eta^T t(x) + \log h(x) - \log Z\big) \end{aligned} \tag{2.65}$$

We define the log-normalizer function $a(\eta)$ as $\log Z$. Another useful property of exponential families is related to the log-normalizer function. The derivatives of the log-normalizer gives the moments of the sufficient statistics $t(x)$ [49] as shown in equation 2.66.

$$\begin{aligned} \frac{\partial}{\partial\eta}a(\eta) &= \frac{\partial}{\partial\eta}\Big(\log\int h(x)\exp\Big(\eta^T t(x)\Big)dx\Big) \\ &= \frac{\int t(x)h(x)\exp\Big(\eta^T t(x)\Big)dx}{\int h(x)\exp\Big(\eta^T t(x)\Big)dx} \\ &= \int t(x)h(x)\exp\Big(\eta^T t(x) - a(\eta)\Big)dx \\ &= E(t(x)) \end{aligned} \tag{2.66}$$

3. GAUSSIAN-GAMMA HIDDEN MARKOV MODEL

All of the calculations we have introduced up to now such as the hidden Markov model in some way assume that we can use the Gaussian distribution to model asset returns. This assumption was very prevalent in most of the research in financial series including the calculation of the risk of sophisticated financial instruments. However, recent financial crises including the Great Recession period between 2007-2009 and the sudden financial impact of COVID-19 in the months of February and March of 2020, showed investors that extreme events that are not accounted for by a Gaussian distribution occasionally can occur. These extreme event may have catastrophic effects on the performance of a portfolio and should be modelled properly. In statistics, dealing with such extreme deviations from the median behavior of a probability distribution appears in the field of extreme value theory.

Kurtosis is a measure of how fast the tail of a probability distribution asymptotically approaches zero. While a high kurtosis means that more extreme events are possible, there is not a single agreed upon mathematical definition of how to measure the kurtosis of a sample of observations. Such probability distributions are known as heavy-tailed or fat distributions as opposed to light tailed distributions such as the Gaussian distribution. One important property that heavy-tailed distributions that must be considered is that the distribution may not have a finite variance or any higher moment. Some commonly used heavy-tailed distributions that have a single tail are the Pareto distribution and Lévy distribution, while distributions with two tails include the Cauchy distribution and the t-distribution.

Another important property of probability distributions is its stability. A distribution is stable if a linear combination of two independent random variables with this distribution also has the same distribution up to its location and scale parameters. The measure of stability in such distributions is defined by a parameter α. Stable distributions have an α value that satisfies $0 < \alpha < 2$ where $\alpha = 2$ corresponds to the

Gaussian distribution and $\alpha = 1$ corresponds to the Cauchy distribution. Except for the Gaussian distribution, all alpha stable distributions are heavy-tailed.

The class of heavy-tailed distributions that are frequently used in financial modelling is the generalised hyperbolic distribution which is a special case of a normal variance-mean mixture. Such mixtures treat the mean and variance parameters of a Gaussian random variable also as random. This expands the Gaussian distribution to include heavy-tailed distributions. In this work, we derive a hidden Markov model with observations based on the Student's t-distribution which is a special case of a generalised hyperbolic distribution.

The multivariate t-distribution has been used as the emission distribution of a hidden Markov model in works including [21] and [19] where its advantages compared to the Gaussian distribution are discussed. Most importantly, it is found that states are much more persistent in such a model. Intuitively, a heavy-tailed distribution allows deviations from the median of the distribution which causes the previous state to persist. This property suits dynamical processes with different regimes since we define regimes as persistent state behavior. For financial data regimes may correspond to the behavior of assets during normal economic activity and financial crises.

Hidden Markov models used in the literature with t-distribution as the emission distribution (SHMM) use the derivation of t-distribution as a scale mixture of the Gaussian distribution where the variance of the Gaussian distribution is treated as a random variable. This formulation allows the expectation-maximization algorithm to be tractable. However, rather than explicitly adding the variance of the Gaussian distribution into the model such works only use this formulation for mathematical convenience. In this work, we will treat the variance of the Gaussian distribution as an additional latent variable in our model. Our formulation allows us to have a more detailed interpretation of the estimated asset returns and covariance during asset

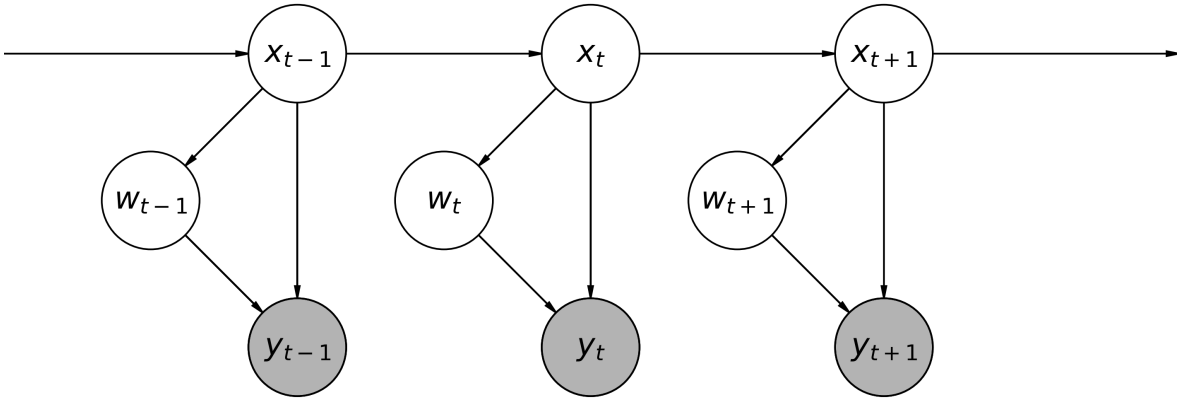

Figure 3.1. Gaussian-gamma hidden Markov model.

allocation which will be discussed in section 4.3.

We assume that the latent discrete state, discrete time first order Markov process $x_{1:T}$ is the same as in a hidden Markov model. However, for a complete description of the model let's define the process again as a N-state system represented by a categorical distribution.

$$f(x_t|x_{t-1}, A) = \prod_{i=1}^{N}\prod_{j=1}^{N} a_{ij}^{[x_{t-1}=i][x_t=j]} \tag{3.1}$$

where a_{ij} is the ith row and j^{th} column of transition matrix A. We will again use the notation that the initial probabilities of states at time 1 is represented by a vector π.

An observation at time t, defined as y_t, is assumed to be conditionally independent of previous observations $y_{1:t-1}$ given the current state x_t and has a k-dimensional

Gaussian distribution with mean μ and an unknown variance. We treat the variance of the observation process as a random variable and therefore define a scaling random variable w_t that has a Gamma distribution with parameters $\alpha = \beta = \frac{\nu}{2}$. Similar to the state x_t of the system, the scaling variable w_t is a latent variable. Furthermore, we assume that the value of parameters of our model depends on the current state of the system. The corresponding model can be shown as:

$$\begin{aligned} w_t|x_t, \nu &\sim \text{Gamma}\big(\frac{\nu}{2}, \frac{\nu}{2}\big) \\ &= \frac{\frac{\nu}{2}^{(\frac{\nu}{2})}}{\Gamma(\frac{\nu}{2})} w_t^{\frac{\nu}{2}-1} \exp\big(-(\frac{\nu}{2})w_t\big) \end{aligned} \tag{3.2}$$

$$\begin{aligned} y_t|x_t, w_t, \mu, \Sigma &\sim \text{Gaussian}\big(\mu, \frac{\Sigma}{\omega_t}\big) \\ &= \frac{\sqrt{\omega_t^k}}{\sqrt{(2\pi)^k|\Sigma|}} \exp\Big(-\frac{1}{2}\omega_t(y_t-\mu)^T\Sigma^{-1}(y_t-\mu)\Big) \end{aligned} \tag{3.3}$$

where Σ is considered to be a constant covariance parameter of the observation density encoding the correlations between different observations. The scale of the covariance parameter is adjusted according to ω_t. When $x_t = i$ the state dependent parameters are shown with the subscript i as μ_i, Σ_i and ν_i. This parameterization is equivalent to setting a conjugate prior on the variance of the Gaussian distribution. Such a formulation of the observation variable y_t has the convenient property that when we integrate over the latent scale variable w_t of the Gaussian distribution it becomes a generalized multivariate t-distribution which has a density shown in equation (3.4). Furthermore, the Gaussian-Gamma distribution density define an exponential family which enable efficient inference.

$$p(y_t|x_t,\mu,\Sigma,\nu) = \frac{\Gamma(\frac{\nu+k}{2})}{\Gamma(\frac{\nu}{2})\nu^{(k/2)}\pi^{(k/2)}|\Sigma|^{(1/2)}}\left(1+\frac{1}{\nu}(y_t-\mu)^T\Sigma^{-1}(y_t-\mu)\right)^{-\frac{\nu+k}{2}} \quad (3.4)$$

Using the model assumptions we have defined, the joint probability distribution of the variables $y_{1:T}$, $w_{1:T}$ and $x_{1:T}$ is defined as:

$$p(y_{1:T},w_{1:T},x_{1:T}|\theta) = p(x_1|\pi)\left[\prod_{t=2}^{T}p(x_t|x_{t-1},A)\right]\prod_{t=1}^{T}p(y_t|w_t,x_t,\mu,\Sigma)p(w_t|x_t,\nu) \quad (3.5)$$

where θ is the set of parameters of the model. The conditional independence assumptions of the model are summarized in Figure 3.1 as a graphical model where observed variables are shaded for emphasis. We define our model as Gaussian-Gamma hidden Markov model (GGHMM) in the rest of the text.

3.1. Inference

As in the case of the HMM, our main goal is to infer the posterior probability distribution of the state of the system at each time t. Assuming that our model sufficiently represents the system and the true parameters of the model are known, the posterior probability of x_t can be calculated without knowing the posterior distribution of w_t since we will be marginalizing it out. However, for a complete understanding of the model and it's requirement in parameter estimation, we will also be concerned with the inference of the posterior distribution of the latent variable w_t.

We formulate the filtering and smoothing problems for the estimation of latent variables in our model using a recursive Bayesian framework. In the GGHMM the filtered density includes the latent variable ω_t and shown as $p(x_t, w_t|y_{1:t}, \theta)$. Similarly

the smoothed density is defined as $p(x_t, w_t|y_{1:T}, \theta)$. Both problems are formulated similar to their HMM counterparts with the addition of new arguments to infer the extra latent variable w_t.

3.1.1. Filtering

The prediction step of the filtering algorithm is identical to the forward algorithm for a HMM since we have formulated the model such that the only dependence between variables at different time steps t is through the state variable x_t. Let's assume that the posterior distribution of the state x_{t-1} at time $t-1$ is known. Using the state evolution equation $f(x_t|x_{t-1}, A)$ we can obtain a prior distribution on the state x_t.

$$p(x_t|y_{1:t-1}, \theta) = \sum_{x_{t-1}} f(x_t|x_{t-1}, A)p(x_{t-1}|y_{1:t-1}, \theta) \tag{3.6}$$

The GGHMM differs from the HMM in its correction step since the posterior distribution over the latent variables include w_t as well as x_t. The joint posterior distribution on the latent variables x_t and w_t is shown in equation (3.7).

$$p(x_t, w_t|y_{1:t}, \theta) = \frac{p(y_t|w_t, x_t, \mu, \Sigma)p(w_t|x_t, \nu)p(x_t|y_{1:t-1}, \theta)}{p(y_t|y_{1:t-1}, \theta)} \tag{3.7}$$

Notice that the denominator $p(y_t|y_{1:t-1}, \theta)$ is obtained by marginalizing x_t and w_t out from the nominator in equation (3.7). At this point, we can obtain the posterior distribution of x_t by using the convenient property of integrating out w_t in order to obtain the density of t-distribution.

$$
\begin{aligned}
p(x_t|y_{1:t},\theta) &= \frac{\int_{w_t} dw_t\, p(y_t|w_t,x_t,\mu,\Sigma)p(w_t|x_t,\nu)p(x_t|y_{1:t-1},\theta)}{p(y_t|y_{1:t-1},\theta)} \\
&= \frac{p(y_t|x_t,\mu,\Sigma,\nu)p(x_t|y_{1:t-1},\theta)}{p(y_t|y_{1:t-1},\theta)}
\end{aligned}
\tag{3.8}
$$

where the density $p(y_t|x_t,\mu,\Sigma,\nu)$ is the generalized t-distribution shown in equation (3.4). Notice that using the chain rule, the posterior distribution of the latent variables can be factorized as shown in equation (3.9).

$$
p(x_t,w_t|y_{1:t},\theta) = p(x_t|y_{1:t},\theta)p(w_t|x_t,y_{1:t},\theta) \tag{3.9}
$$

Using the factorization shown in equation (3.9), the filtered posterior distribution of w_t can be obtained using equation (3.10).

$$
p(w_t|x_t,y_{1:t},\theta) = \frac{p(y_t|w_t,x_t,\mu,\Sigma)p(w_t|x_t,\nu)p(x_t|y_{1:t-1},\theta)}{p(x_t,y_t|y_{1:t-1},\theta)} \tag{3.10}
$$

where the joint density $p(x_t,y_t|y_{1:t-1},\theta)$ is equivalent to the nominator of equation (3.8). Let's show that with some rearrangement of parameters, the joint density $p(w_t|x_t,y_{1:t},\theta)$ actually defines an exponential family in equation (3.11). As shown in section 2.6, this property enables us to have some very efficient algorithms.

$$p(w_t|x_t, y_{1:t}, \theta) = \frac{p(y_t|w_t, x_t, \mu, \Sigma)p(w_t|x_t, \nu) \sum_{x_{t-1}} f(x_t|x_{t-1}, \theta)p(x_{t-1}|y_{1:t-1})}{p(y_t|x_t, \mu, \Sigma, \nu) \sum_{x_{t-1}} f(x_t|x_{t-1}, \theta)p(x_{t-1}|y_{1:t-1})} \tag{3.11}$$

where the denominator is a constant since y_t is observed and the nominator is made up of the product of two exponential family distributions. If we insert the multivariate t-distribution density in place of $p(y_t|x_t, \mu, \Sigma, \nu)$, the scaled multivariate Gaussian distribution density in place of $p(y_t|w_t, x_t, \mu, \Sigma)$ and the Gamma distribution density in place of $p(w_t|x_t, \nu)$, we obtain the following functional form for the posterior distribution of w_t.

$$p(w_t|x_t = i, y_{1:t}, \theta) = \frac{w_t^{\frac{\nu_i - 2 + k}{2}} \exp\left(- \left[\frac{1}{2}(y_t - \mu_i)^T \Sigma^{-1} (y_t - \mu_i)\right] w_t \right)}{\Gamma(\frac{\nu_i + k}{2}) \left(\frac{1}{2}(y_t - \mu_i)^T \Sigma^{-1} (y_t - \mu_i) + \frac{\nu_i}{2} \right)^{-\frac{\nu_i + k}{2}}} \tag{3.12}$$

3.1.2. Moments of the Latent Scaling Variable

Since we will be needing to calculate some functions of w_t in the learning and asset allocation sections, let's analyze the posterior distribution of w_t shown in equation 3.12 further. Firstly, notice that since w_t is a latent variable that is not observed and there is no dependence between consecutive time steps, the only information we have about w_t is through the observation of y_t and estimate of x_t. Any observation other than y_t does not have any influence over our estimates of w_t. While we expect that consecutive w_t values do not vary significantly in real data sets, the GGHMM model does not take this dependence into account in order to simplify the mathematical formulation of the model. This assumption mathematically means that we cannot improve our estimate of the posterior distribution by using a smoothing procedure and $p(w_t|x_t = i, y_{1:t}, \theta)$ is equal to $p(w_t|x_t = i, y_{1:T}, \theta)$. We will comment on this limitation of our model in

chapter 6.

Let's now calculate the moments of the sufficient statistics of w_t defined by equation 3.12. As shown in section 2.6, let's define the functions $t(x)$, $\eta(\theta)$ and $a(\eta)$. We have a 2-dimensional sufficient statistics shown as:

$$\begin{aligned} t_1(w_t) &= -w_t \\ t_2(w_t) &= \log(w_t) \end{aligned} \tag{3.13}$$

The parameters $\eta(\theta)$ corresponding to these sufficient statistics are:

$$\begin{aligned} \eta_1(\theta) &= \Big[\frac{1}{2}(y_t - \mu_i)^T \Sigma^{-1}(y_t - \mu_i) + \frac{\nu_i}{2}\Big] \\ \eta_2(\theta) &= \Big(\frac{\nu_i - 2 + k}{2}\Big) \end{aligned} \tag{3.14}$$

Most importantly the log-normalizer function $a(\eta)$ is defined as:

$$\begin{aligned} a(\eta) &= \log\Bigg[\Big(\frac{1}{2}(y_t - \mu_i)^T \Sigma^{-1}(y_t - \mu_i) + \frac{\nu_i}{2}\Big)^{-(\frac{\nu_i + k}{2})} \Gamma(\frac{\nu_i + k}{2})\Bigg] \\ &= \log\Bigg[\eta_1^{-(\eta_2+1)} \Gamma(\eta_2 + 1)\Bigg] \end{aligned} \tag{3.15}$$

Now, using the log-normalizer function, we can calculate the first moments of the sufficient statistics $-w_t$ and $\log(w_t)$.

$$\begin{aligned}\frac{\partial}{\partial\eta_1}a(\eta) &= \frac{1}{\eta_1^{-(\eta_2+1)}\Gamma(\eta_2+1)}\Gamma(\eta_2+1)(-1)(\eta_2+1)\eta_1^{-(\eta_2+2)} \\ &= (-1)(\eta_2+1)\eta_1^{-1} \\ &= E(-w_t)\end{aligned} \tag{3.16}$$

We can transform the parameters η_1 and η_2 into their θ forms in order to obtain an expression for the expectation of w_t.

$$\begin{aligned}\frac{\partial}{\partial\eta_1}a(\eta) &= (-1)\Big(\frac{\nu_i+k}{2}\Big)\Big[\frac{1}{2}(y_t-\mu_i)^T\Sigma^{-1}(y_t-\mu_i)+\frac{\nu_i}{2}\Big]^{-1} \\ &= E(-w_t)\end{aligned} \tag{3.17}$$

Since the expectation operator scales linearly when multiplied by a constant factor, we find that:

$$E(w_t) = \Big(\frac{\nu_i+k}{2}\Big)\Big[\frac{1}{2}(y_t-\mu_i)^T\Sigma^{-1}(y_t-\mu_i)+\frac{\nu_i}{2}\Big]^{-1} \tag{3.18}$$

Similarly, let's find the first moment of $\log(w_t)$:

$$\begin{aligned}\frac{\partial}{\partial\eta_2}a(\eta) &= \frac{1}{\eta_1^{-(\eta_2+1)}\Gamma(\eta_2+1)}\Big(\eta_1^{-(\eta_2+1)}(-1)\Gamma(\eta_2+1)\ln(\eta_1)+\Gamma(\eta_2+1)\eta_1^{-(\eta_2+1)}\psi_0(\eta_1)\Big) \\ &= \psi_0(\eta_1)-\ln(\eta_1)\end{aligned} \tag{3.19}$$

where the function $\psi_0(z)$ is the digamma function which describes the derivative of the Gamma function $\Gamma(z)$ through the identity:

$$\Gamma'(z) = \Gamma(z)\psi_0(z) \tag{3.20}$$

We can transform the parameters η_1 and η_2 into their θ forms in order to obtain an expression for the expectation of $\log(w_t)$.

$$\begin{aligned} \frac{\partial}{\partial \eta_2} a(\eta) &= \psi_0(\eta_1) - \ln(\eta_1) \\ &= \psi_0\left(\frac{\nu_i + k}{2}\right) - \ln\left(\frac{1}{2}(y_t - \mu_i)^T \Sigma^{-1}(y_t - \mu_i) + \frac{\nu_i}{2}\right) \end{aligned} \tag{3.21}$$

$$E\left(log(w_t)\right) = \psi_0\left(\frac{\nu_i + k}{2}\right) - \ln\left(\frac{1}{2}(y_t - \mu_i)^T \Sigma^{-1}(y_t - \mu_i) + \frac{\nu_i}{2}\right) \tag{3.22}$$

3.1.3. Smoothing

In this section, our goal is to obtain a smoothed estimate of the latent variables $x_{1:T}$ which utilizes the full information available at time T. As we have mentioned, the smoothed posterior density of w_t is equal to the filtered posterior since we do not have any sequential dependence between consecutive w_t values. The smoothed estimate we aim to calculate in this section is defined as $p(x_t|y_{1:T}, \theta)$.

Notice that similar to our considerations in an HMM, we can use the filtered

estimates $p(x_t = i, y_{1:t}|\theta)$ together with a density $p(y_{t+1:T}|x_t = i, \theta)$ in order to calculate the joint probability density of the model:

$$p(x_t, y_{1:T}|\theta) = p(x_t, y_{1:t}|\theta)p(y_{t+1:T}|x_t, \theta) \tag{3.23}$$

which can be normalized by using the incomplete data likelihood $p(y_{1:T}|\theta)$ in order to obtain the smoothed estimate $p(x_t, |y_{1:T}, \theta)$.

Assuming we know $p(y_{t+1:T}|x_t, \theta)$, We can find a joint distribution $p(y_{t:T}|x_{t-1}, \theta)$ by initially using the emission density $p(y_t|x_t, \mu, \Sigma, \nu)$ of the multivariate t-distribution in order to find the likelihood of $y_{t:T}$ for each state x_t. We call this operation the update step since it involves the addition of an observation y_t to a known joint probability distribution.

$$p(y_{t:T}|x_t, \theta) = p(y_t|x_t, \mu, \Sigma, \nu)p(y_{t+1:T}|x_t, \theta) \tag{3.24}$$

The second step we defined as postdiction in the HMM section is identical to section 2.5. We estimate the likelihood of $y_{t:T}$ given that we have an estimate of x_{t-1}.

$$p(y_{t:T}|x_{t-1}, \theta) = \sum_{x_t} p(y_{t:T}|x_t, \theta)f(x_t|x_{t-1}, A) \tag{3.25}$$

3.2. Learning

We formulate a maximum likelihood estimation procedure for our model based on the expectation-maximization algorithm. The well known Baum-Welch algorithm is derived for the hidden Markov model which has the same state transition structure as our model. Thus, we build upon this algorithm to derive update equations for the model parameters π, A, μ, σ and ν. The maximum likelihood estimation problem can be stated as:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \quad \log p(y_{1:T}|\theta) \tag{3.26}$$

The expectation maximization algorithm is a method of maximizing equation (3.26) in an iterative manner. Using Jensen's inequality we can find a lower bound to the incomplete data likelihood $p(y_{1:T}|\theta)$ and maximize this lower bound at each iteration. The lower bound turns out to be in the form of an expectation over the posterior distribution of the latent variables $x_{1:T}$ and $w_{1:T}$ which is shown in equation 3.27.

$$\begin{aligned} \log p(y_{1:T}|\theta) &\geq \sum_{x_{1:T}} \int_{w_{1:T}} dw_{1:T}\, p(x_{1:T}, w_{1:T}|y_{1:T}, \theta) \log \frac{p(y_{1:T}, x_{1:T}, w_{1:T}|\theta)}{p(x_{1:T}, w_{1:T}|y_{1:T}, \theta)} \\ &= E_{p(x_{1:T}, w_{1:T}|y_{1:T}, \theta)} \left[\log \frac{p(y_{1:T}, x_{1:T}, w_{1:T}|\theta)}{p(x_{1:T}, w_{1:T}|y_{1:T}, \theta)} \right] \end{aligned} \tag{3.27}$$

Let's define the estimated parameters at each iteration with a superscript k, shown as $\theta^{(k)}$. For computational efficiency, the posterior distribution over the la-

tent variables are calculated with the set of parameters $\theta^{(k)}$, which then reduces the maximization problem into:

$$\begin{aligned}\theta^{(k+1)} &= \underset{\theta}{\text{argmax}} \quad \sum_{x_{1:T}} \int_{w_{1:T}} dw_{1:T}\, p(x_{1:T}, w_{1:T}|y_{1:T}, \theta^{(k)}) \log p(y_{1:T}, x_{1:T}, w_{1:T}|\theta) \\ &= \underset{\theta}{\text{argmax}} \quad Q(\theta, \theta^{(k)})\end{aligned} \tag{3.28}$$

When we insert the joint probability distribution of the model defined in equation 3.5 into $Q(\theta, \theta^{(k)})$, we obtain:

$$\begin{aligned}Q(\theta, \theta^{(k)}) = &\sum_{x_1} p(x_1|y_{1:T}, \theta^{(k)}) \log p(x_1|\pi) \\ &+ \sum_{t=2}^{T} \sum_{x_t} \sum_{x_{t-1}} p(x_t, x_{t-1}|y_{1:T}, \theta^{(k)}) \log p(x_t|x_{t-1}, A) \\ &+ \sum_{t=1}^{T} \sum_{x_t} \int_0^{\infty} dw_t\, p(x_t, w_t|y_{1:T}, \theta^{(k)}) \log p(y_t|x_t, w_t, \mu, \Sigma) p(w_t|x_t, \nu)\end{aligned} \tag{3.29}$$

The details of this derivation is similar to the considerations in the Baum-Welch algorithm which are shown in detail in section 2.5. In summary, it is based on the chain rule of probability applied to the posterior distribution of the latent variables. Only the third line of equation 3.29 is different from the hidden Markov model, therefore we will concentrate on that line. Notice that using the chain rule the posterior distribution of the latent variables can be factorized as:

$$p(x_t, w_t|y_{1:T}, \theta^{(k)}) = p(x_t|y_{1:T}, \theta^{(k)}) p(w_t|x_t, y_{1:T}, \theta^{(k)}) \tag{3.30}$$

Using this factorization and the expanding property of logarithms, we can rewrite the third line 3.29 as:

$$\begin{aligned}
&\sum_{t=1}^{T}\sum_{x_t}\int_0^\infty dw_t\, p(x_t, w_t|y_{1:T}, \theta^{(k)})\log p(y_t|x_t, w_t, \mu, \sigma)p(w_t|x_t, \nu) \\
&= \sum_{t=1}^{T}\sum_{i=1}^{N} p(x_t = i|y_{1:T}, \theta^{(k)}) \int_0^\infty dw_t\, p(w_t|x_t = i, y_{1:T}, \theta^{(k)})\log p(y_t|x_t = i, w_t, \mu, \Sigma) \\
&+ \sum_{t=1}^{T}\sum_{i=1}^{N} p(x_t = i|y_{1:T}, \theta^{(k)}) \int_0^\infty dw_t\, p(w_t|x_t = i, y_{1:T}, \theta^{(k)})\log p(w_t|x_t = i, \nu)
\end{aligned} \tag{3.31}$$

In order to find an updated value for observation parameters, we insert the probability density functions of $p(y_t|x_t = i, w_t, \mu, \Sigma)$ and $p(w_t|x_t = i, \nu)$. Then we need to take the derivative of equation (3.31) and set it to zero. The update equations for μ_i and Σ_i are shown below:

$$\mu_i^{(k+1)} = \frac{\sum_{t=1}^{T} p(x_t = i|y_{1:T}, \theta^{(k)})\, \langle w_{t,i}\rangle\, y_t}{\sum_{t=1}^{T} p(x_t = i|y_{1:T}, \theta^{(k)})\, \langle w_{t,i}\rangle} \tag{3.32}$$

$$\Sigma_i{}^{(k+1)} = \frac{\sum_{t=1}^{T} p(x_t = i|y_{1:T}, \theta^{(k)})\, \langle w_{t,i}\rangle\, (y_t - \mu_i^{(k+1)})(y_t - \mu_i^{(k+1)})^T}{\sum_{t=1}^{T} p(x_t = i|y_{1:T}, \theta^{(k)})\, \langle w_{t,i}\rangle} \tag{3.33}$$

where we denote the expected value of $w_{t,i}$ as $\langle w_{t,i}\rangle$ for brevity. Notice that we have shown that the calculation of this expectation is relatively simple because of the exponential family form of the posterior distribution of $w_{t,i}$ shown in 3.1.2. Therefore, equations (3.32) and (3.33) have similar analogs in Baum-Welch update equations,

and their calculation is straightforward.

Unfortunately, the update equation for ν_i does not have a simple form similar to equations (3.32) and (3.33) because of the presence of the Gamma function $\Gamma(z)$ in the gamma distribution $p(w_t|x_t = i, \nu)$. We have to introduce the digamma function $\psi_0(z)$ which is defined as the logarithmic derivative of the gamma function. Taking the derivative of equation (3.31) with respect to ν_i and setting it equal to zero yields:

$$0 = \log\Big(\frac{\nu_i}{2}\Big) + 1 - \psi_0\Big(\frac{\nu_i}{2}\Big) + \mathrm{C} \tag{3.34}$$

where C denotes a constant with respect to ν_i and can be shown as:

$$\mathrm{C} = \frac{\sum_{t=1}^{T} p(x_t = i|y_{1:T}, \theta^{(k)}) \int_0^\infty dw_t \, p(w_t|x_t = i, y_{1:T}, \theta^{(k)}) \big(\log(w_{t,i}) - w_{t,i}\big)}{\sum_{t=1}^{T} p(x_t = i|y_{1:T}, \theta^{(k)})} \tag{3.35}$$

Notice that the nominator of the equation above includes the expected value of $\log(w_t)$ and w_t which we previously found in section 3.1.2.

In order to find a value $\nu_i^{(k+1)}$, we propose to find the root of the equation (3.34) by using the bisection (binary search) method. We prefer this method over other root finding algorithms such as the Newton–Raphson method because it enables us to constrain the space of possible ν_i values to be in $(1, \infty)$. This is important for the robustness of the parameter estimation process since the student's t-distribution is only defined for $\nu > 0$ and its mean is defined for $\nu > 1$. When the mean of the emission density is not defined, all other parameters diverge. An alternative method of approximating the value of $\nu_i^{(k+1)}$ is shown in [50].

4. APPLICATION OF GAUSSIAN-GAMMA HIDDEN MARKOV MODEL

4.1. Modern Portfolio Theory

The mathematical framework for diversification was initially developed by Harry Markowitz in his essay Portfolio Selection published in *The Journal of Finance* in 1952 [2]. The framework is generally referred to as the Modern Portfolio Theory or Markowitz Mean-Variance Analysis. The essay introduces basic principles for a rational choice of portfolio among many asset classes such as equities or stocks, bonds, cash, foreign currency, and commodities.

4.1.1. Financial Interpretations

The goal of constructing an optimal portfolio is to allocate wealth in such a way that any negative return period in a set of assets is mitigated by positive returns in another set of assets that are negatively correlated. As mentioned above, this is called diversification and it is a popular risk management method in quantitative finance. Contrary to the belief that financial experts predict the single asset class that will perform the best and allocate all of their wealth into it, investment strategies are based on statistical arguments about the long-term characteristics of financial assets.

The Modern Portfolio Theory, or Markowitz Mean-Variance Analysis (MVA), attempts to find an optimal selection of assets for a single investment period. The main argument in the framework is that each asset can be modelled as a tuple of expected return and standard deviation. This enables us to visualize each asset or portfolio as a point in the risk-return spectrum as shown in Figure 4.1. The red dots in the figure correspond to individual assets, while the blue dots correspond to possible portfolios constructed with the assets in question.

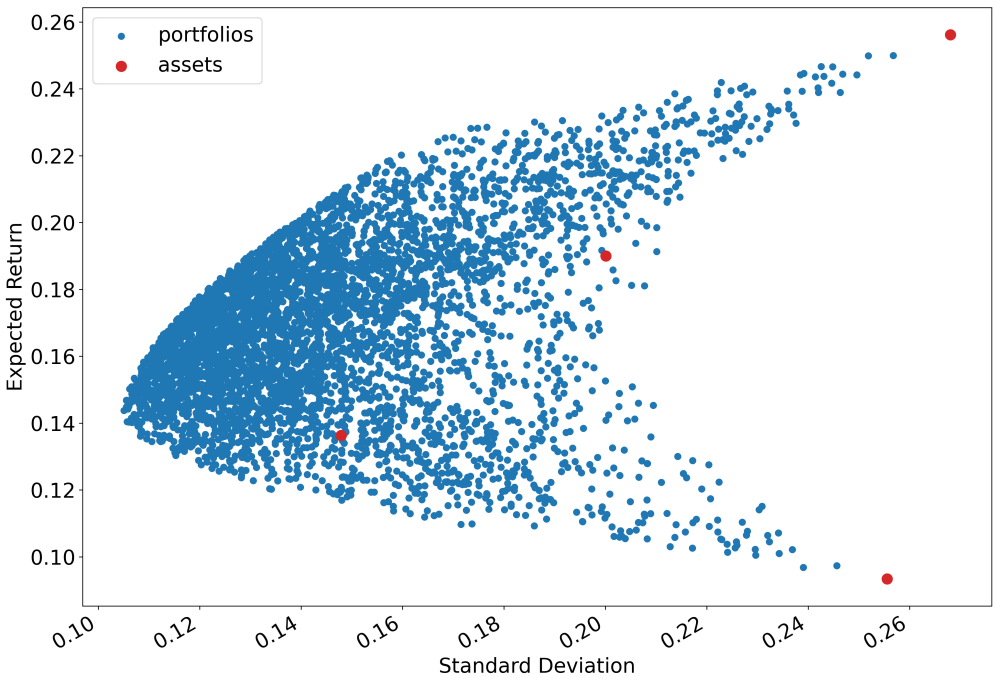

Figure 4.1. Visualization of real assets and portfolios.

Notice that choosing an asset based only on its expected return carries a significant uncertainty with it. This is due to the fact that risk and reward are inherently related in financial markets. However, while two assets may have a significant uncertainty associated with them, their uncertainties may be related. For example, in general when stock prices fall, gold prices increase since investors are transferring their money from stocks to gold. This observation may enable us to construct a portfolio with both stocks and gold in order to balance their uncertainties and create a new product that has significantly less standard deviation when compared to stock indices but a higher expected return compared to gold indices.

As it can be seen in Figure 4.1, the optimal portfolio is found to have a parabolic shape in the expected portfolio return axis. The upper part of the parabolic shape is called the efficient frontier since the portfolios corresponding to that curve offer the highest rate of expected return for a given level of standard deviation or risk. Therefore, any portfolio that lie below the efficient frontier curve is considered sub-optimal. Since the risk aversion characteristics of any investor is different, MVA finds a set of optimal portfolios corresponding to different risk tolerance levels of the investor. Optimality is defined in terms of higher expected returns for a chosen level of risk or lower risk for a chosen level of expected return.

4.1.2. Mathematical Framework

Let's assume that the historical return dynamics of assets completely represent the future return dynamics. Therefore, the expected return of an asset is assumed to be the historical mean return value, represented as α. Furthermore, the risk associated with an asset is defined as the standard deviation of the historical return data, represented as σ. Notice that this is equivalent to treating the return data set as independent and identically distributed which we discuss in detail in section 2.1. These assumptions lead to a mathematical representation of an asset as a point in the risk-return spectrum. MVA assumes that the return series r_i for any asset i is a random variable sampled from an unknown distribution with known expected return and variance:

$$\begin{aligned} \mathrm{E}(r_i) &= \alpha_i \\ \mathrm{Var}(r_i) &= \sigma_i^2 \end{aligned} \tag{4.1}$$

Since there are multiple assets to choose in a financial market, m is defined as the number of risky assets available. The return at time t of all available assets can be represented as an m-variate random vector:

$$\mathbf{r_t} = \begin{bmatrix} r_{t,1} \\ r_{t,2} \\ \vdots \\ r_{t,m} \end{bmatrix} \tag{4.2}$$

Since $\mathbf{r}$ is structured as a multivariate distribution, the expectation and covariance of the return series can be summarized as follows:

$$\begin{aligned} \mathrm{E}(\mathbf{r}) &= \alpha \\ &= \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{bmatrix} \end{aligned} \tag{4.3}$$

$$\begin{aligned} \mathrm{Cov}(\mathbf{r}) &= \mathbf{\Sigma} \\ &= \begin{bmatrix} \Sigma_{1,1} \dots \Sigma_{1,m} \\ \vdots \quad \dots \quad \vdots \\ \Sigma_{m,1} \dots \Sigma_{m,m} \end{bmatrix} \end{aligned} \tag{4.4}$$

A portfolio can be defined as an m-variate random vector of weighted assets. Since the number of data points for each asset is assumed to be large, the mean return of each asset can be assumed to have a Gaussian distribution using the central limit theorem. Then, considering the summation of multiple Gaussian random variables still behaves as a Gaussian random variable, a portfolio becomes a multivariate random variable $\mathcal{N}(\alpha_w, \sigma_w^2)$ where α_w is the expected return of the portfolio and σ_w^2 is the variance of the portfolio. The index ω describes the m-vector ω, with each element ω_i corresponding to the fraction of wealth held in the ith asset. Positive values for ω_i are equivalent to long positions on the related asset, while negative values are for short positions.

$$\omega = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_m \end{bmatrix} \tag{4.5}$$

While there are complicated financial instruments that enable leveraging the principal capital of an investor, let's assume that we are constrained to only invest our wealth in long only positions which corresponds to the mathematical constraint on ω such that $\sum_{i=1}^{m} \omega_i \leq 1$ and $0 \leq \omega_i$ for all i.

The mathematical construction of an optimal portfolio differs in the case where the sum of all weights equal 1 and where the sum of all weights is less than 1 in the literature. This is due to the fact that there are risk-free investments such as deposit accounts that will have a deterministic outcome rather than an uncertain outcome that is represented by a probability distribution. An investor may decide to allocate some of her wealth to risky assets which corresponds to $\sum_{i=1}^{m} \omega_i < 1$ where the rest of the capital is allocated to risk-free assets. While this distinction may be useful in practise, since our aim is to learn the dynamics of asset returns without supervision we will treat risk-free investments in the same framework and try to learn its relation to other asset classes. It is useful to point out that the expected return of the risk-free asset should be lower than any other asset class while also having a lower variance or risk than any other asset class.

The m-variate Gaussian random vector of weighted assets corresponding to a portfolio can be shown as:

$$\mathbf{p}_\omega = \begin{bmatrix} \omega_1 r_1 \\ \omega_2 r_2 \\ \vdots \\ \omega_m r_m \end{bmatrix} \tag{4.6}$$

The expected return of the portfolio α_ω can simply be calculated from the weighted sum of the expected return of assets:

$$\begin{aligned} \alpha_\omega &= \omega^T \alpha \\ &= \sum_{i=1}^{m} \omega_i \alpha_i \end{aligned} \tag{4.7}$$

The variance of the portfolio must reflect the weight of each asset in the portfolio as well as their covariance relationship. The multivariate variance is defined through considering the pairwise contributions of all assets.

$$\begin{aligned} \sigma_w^2 &= \omega^T \Sigma \omega \\ &= \sum_{j=1}^{m} \sum_{i=1}^{m} \omega_i \Sigma_{ij} \omega_j \end{aligned} \tag{4.8}$$

The problem of finding an optimal portfolio may be constructed in two different ways. While the two methods are logically equivalent, they differ in terms of which parameter they want to optimize. The first is the risk minimization problem which requires the investor to choose an expected return of portfolio before minimizing the risk. While the other method is the expected return maximization problem which works for a chosen level of risk parameter. In practise, experience shows that the problem of creating an optimal portfolio is done in the latter way where an investor defines a maximum level of risk. Thus, we will introduce the optimization procedure using the expected return maximization problem.

4.1.3. Portfolio Optimization

As mentioned above, we assume that the investor has a maximum risk level or expected standard deviation, σ_w^2, for the returns of the portfolio. This assumption corresponds to having an inequality constraint in the formulation of the optimization problem on top of the constraints on ω mentioned in section 4.1.2. We can set up the optimization problem as:

$$\begin{aligned} \max_{w} \quad & \omega^T \alpha \\ \text{s.t.} \quad & \omega^T \Sigma \omega \leq \sigma_w^2 \\ & \sum_{i=1}^{m} \omega_i = 1 \\ & 0 \leq \omega_i \end{aligned} \tag{4.9}$$

where the objective is to maximize the expected return of the portfolio.

A convex optimization problem subject to linear equality constraints can be solved using the method of Lagrange multipliers. However, since we have an inequality constraint a more generalized method compared to the method of Lagrange multipliers is required. The Karush-Kuhn-Tucker (KKT) conditions of mathematical optimization may be viewed as a more generalized framework for introducing constraints into the objective function. The KKT conditions are necessary conditions for optimality and are stated as: Stationarity, Primal Feasibility, Dual Feasibility and Complementary Slackness. The detailed explanations of each condition is out of the scope of this work. The stationarity condition is defined as:

$$\nabla f(x^*) = \sum_{i=1}^{n} \delta_i \nabla g_i(x^*) + \sum_{j=1}^{l} \lambda_j \nabla h_j(x^*) \tag{4.10}$$

where $g_i(x^*)$ is the ith inequality constraint and $h_j(x^*)$ is the jth equality constraint. $f(x^*)$ is the objective function evaluated at x^* and δ_i and λ_j are called KKT multipliers. It is important to point out that the above equation is valid only for maximization problems. A minimization problem would require a different approach. Additionally, primal feasibility condition imposes the following constraints:

$$g_i(x^*) \leq 0 \tag{4.11}$$

$$h_j(x^*) = 0 \tag{4.12}$$

for all $i \in 1, 2, \ldots, n$ and $j \in 1, 2, \ldots, l$. Let's transform the optimization problem according to KKT conditions:

$$\begin{aligned} \max_{w} \quad & f(x) = \omega^T \alpha \\ \text{s.t.} \quad & g(x) = \omega^T \Sigma \omega - \sigma_w^2 \leq 0 \\ & h(x) = \omega^T \mathbf{1}_m - 1 = 0 \end{aligned} \tag{4.13}$$

where the constraint $\omega^T \mathbf{1}_m - 1 = 0$ corresponds to $\sum_{i=1}^{m} \omega_i - 1 = 0$ in vector notation. Now, we can use the stationarity condition to obtain the optimal weight distribution, w^*.

According to KKT conditions, the ω that satisfies the stationarity condition must be optimal:

$$\alpha = 2\delta \Sigma \omega^* + \lambda \tag{4.14}$$

Some simple mathematical manipulations on the above equation yields the optimum asset allocation, ω^*:

$$\omega^* = \frac{1}{2\delta}\Sigma^{-1}(\alpha - \lambda) \tag{4.15}$$

Notice that ω^* is found as a function of δ and λ. The values of δ and λ can be obtained by inserting the expression for ω^* to the corresponding constraint equation as shown below:

$$\frac{1}{2\delta}\sum_{i=1}^{m}\sum_{j=1}^{m}\Sigma_{ij}^{-1}(\alpha - \lambda)_j = 1 \tag{4.16}$$

and

$$\frac{1}{4\delta^2}(\alpha - \lambda)^T\Sigma^{-1}(\alpha - \lambda) - \sigma_w^2 \leq 0 \tag{4.17}$$

Let's analyze equation 4.17 further. By rearranging the values in the equation we obtain:

$$\frac{1}{2\sigma_w}\sqrt{(\alpha - \lambda)^T\Sigma^{-1}(\alpha - \lambda)} \leq \delta \tag{4.18}$$

Since this is an inequality, there isn't a single δ value to choose, any value that satisfies the equation above corresponds to a mathematically optimal portfolio. We need to solve equation 4.16 together with 4.17. However, when we observe the parabolic shape of the efficient frontier shown in figure 4.1, we see that as the standard deviation or risk of the portfolio increase, the expected return also increase. Thus, we can find the portfolio weights that maximize the expected return by treating the inequality in equation 4.17 as an equality.

4.2. Asset Allocation with GHMM

In section 4.1, we have introduced an asset allocation framework. However, in the classical mean-variance analysis framework the historical sample statistics are used in order to model the dynamics of each asset class. As discussed, treating a time series as an i.i.d. data set does not accurately represent the sequential structure of the process. Using the HMM, we can estimate the expected return of each asset class as well as their covariance structure while also considering the sequential structure of the data. On top of that, we can find multiple characteristic behavior states of the process in order to take the non-linear nature of financial assets. The HMM has been used in various works in the literature to estimate the dynamics of financial series, these works include [51], [13] and [52].

The simplest way to allocate wealth based on a HMM is to predict the most likely state at time t where we want to construct a portfolio. At time $t-1$, we will be able to calculate the filtered posterior probability distribution over the states x_{t-1}. Using the probability distribution $p(x_{t-1}|y_{1:t-1})$, we can use the prediction procedure shown in equation 2.19 in order to obtain $p(x_t|y_{1:t-1})$. Notice that we will only be able to use the filtering procedure during asset allocation since we must make online calculations. Thus, we solve equation 4.19 in order to find i^* defined as the estimated most likely state at time t.

$$i^* = \underset{i}{\text{argmax}} \quad p(x_t = i | y_{1:t-1}) \tag{4.19}$$

Once we solve for i^*, we assume that the parameters μ_{i^*} and Σ_{i^*} learned for the HMM represent the return dynamics of assets at time t. It is straightforward to allocate wealth based on these estimates using the framework in section 4.1.

4.3. Asset Allocation with GGHMM

Since financial data is heavy-tailed, by considering the standard deviation as a risk measure, the framework of mean-variance analysis does not take into account the possible extreme deviations from the median behavior. This is a major issue in portfolio management that causes the mean-variance analysis to be abandoned. Since then there hasn't been an industry standard in portfolio optimization. Multiple different frameworks are being developed based on the notion of extreme risk. Some of the important works in this area are [53] and [54]. These works usually try to infer the actual risk involved with an asset and construct a portfolio such that the portfolio is robust with respect to extreme events for a certain interval of time. However, because of the simple and elegant form of mean-variance analysis, it is our belief that the framework should not be completely disregarded and can be slightly adjusted to include the risk of extreme events.

Let's recall that using the GGHMM, we model the financial returns as:

$$y_t | x_t, w_t, \mu, \Sigma \sim \text{Gaussian}\big(\mu, \frac{\Sigma}{\omega_t}\big) \tag{4.20}$$

such that the covariance is actually time dependent. This dynamic model on the covariance structure allows extreme events that a Gaussian distribution cannot take into account. We assume that while extreme events in finance can unexpectedly occur, they also have the property that they persist, even if the extreme period is short-lived. Thus, our goal is to avoid persistently underestimating the risk involved in an asset during a period of crisis. Notice that this is different from constructing a portfolio that is robust to extreme events for any period which the literature usually covers. GGHMM differs from such a robust portfolio in terms of its dependence on dynamically changing behavior of the markets. The model does not take an extreme event into account before it actually occurs, but it is robust with respect to avoiding such events that occur consecutively.

In practise, since we cannot calculate w_t before observing y_t we need a reasonable way of predicting its value. Having a better estimate of w_t is the key to measure the current risk involved with each asset. Related to our assumption that extreme events persists, we assume that the observation y_{t-1} can be used to calculate the expected value $\langle w_t \rangle$ instead of y_t. Thus, we assume that the expected value of w_t is an accurate representation of the distribution of w_t. Lastly, similar to our asset allocation strategy in section 2.5, we estimate the most likely state at time t using equation 4.19 and use the corresponding mean parameter μ_i and covariance parameter $\Sigma_i / \langle w_t \rangle$.

5. EXPERIMENTS AND RESULTS

In this chapter, we test the performance of our proposed Gaussian-Gamma hidden Markov model with synthetically generated data and historical financial data sets. First, we present the theoretical capabilities of our model by evaluating its performance on state identification for synthetically generated multivariate heavy-tailed time series data and compare it to a Gaussian hidden Markov model. These results will reflect our theoretical expectation on the performance of a GHMM and GGHMM in determining financial regimes. Afterwards, we continue with testing the performance of the model on a multivariate financial data set including the return data of Istanbul Stock Exchange National 100 Index, USD/TRY exchange rates, gold exchange rate and interest rate reported by Central Bank of the Republic of Turkey and Nasdaq Composite Index.

Since the state of the financial markets are unknown, we measure the performance of our model with respect to two different tasks. The first task is to identify regime shifts using the complete data set in order to identify business cycles. This task is periodically performed by experts of the National Bureau of Economic Research (NBER) in the United States to be used in conjunction with other economic research tasks. It is shown in [55] that the GHMM yields comparable results to the economics experts in this task. Thus, it is of interest to create better models. The second task is to measure the performance of the GGHMM in asset allocation by forecasting the economic state of the market. We will use an expanding window of training data in this task to have an accurate representation of reality. The GGHMM and GHMM will be compared to the performance of the plain mean-variance allocation strategy.

In each section, we start by presenting a single realization of the experiment which we believe to be representative of the capabilities of the model. Results include visualizations of the data set as well as the performance of both models on the respective task. We continue our discussion with presenting statistical results of multiple realizations of the experiment together with strengths and weaknesses of each model.

5.1. Synthetic Data

In order to generate a multivariate heavy-tailed time series data set we use a state-space model with an identical structure to the GGHMM. The parameters of the model are randomly generated in each realization of the experiment with a 2-dimensional latent Markovian state space. Our aim is to understand the effects of maximum training epochs used for each realization and dimensionality of the observation space on the state identification task using smoothed estimates of the latent states. We presents results for 3-dimensional and 5-dimensional observation spaces of length 500. Each realization is indexed by dates in order to emphasize the sequential nature of the data set.

5.1.1. Evaluation Criteria

There are various evaluation criteria for classification and clustering tasks such as accuracy, precision and recall. However, we need to consider the underlying decision making problem in order to select an appropriate metric. While our main goal is to identify the state at each time t, since we know that these states are persistent, the difficulty of this task arise from state change-points.

Let's discuss the potential effectiveness of each evaluation metric in order to understand their significance. Accuracy, defined in equation 5.1, measures the correctness of all state identifications. This measure does not reflect the persistent nature of our dynamical system since correctly guessing that there wasn't any change in the system is not as informative as detecting a change.

$$\text{Accuracy} = \frac{\sum \text{True Positive} + \sum \text{True Negative}}{\sum \text{Total Population}} \tag{5.1}$$

Instead of focusing on identifying every state, let's direct our attention to how

well we can identify a specific state, which is analogous to being able to identify a financial crisis in real data sets. This change in our focus leads us to the precision and recall metrics which are defined in equations 5.2 and 5.3, respectively. Precision is a measure of the correctness of our positively identified states and recall is a measure of how correct we were actually able to identify a specific state in reality. These two metrics are a better measure of the performance of our models considering the crisis identification task. Thus, similar to a financial crisis regime, we select the state with higher overall standard deviations as the relevant state to be identified and report the corresponding precision and recall values for each realization of the experiment.

$$\text{Precision} = \frac{\sum \text{True Positive}}{\sum \text{True Positive} + \sum \text{False Positive}} \tag{5.2}$$

$$\text{Recall} = \frac{\sum \text{True Positive}}{\sum \text{Condition Positive}} \tag{5.3}$$

Since we do not want our models to be dependent on a specific data set, we generate multiple different realization of the experiments with randomly generated data sets and compare the models based on an equally weighted average performance on these different realizations. The results report the mean and standard deviation in each evaluation criteria.

5.1.2. Visualization of Experiments

Let's start our discussion by showing a single realization of the experiment where the observation space is 5-dimensional. The observations and their cumulative product are visualized in Figure 5.1, where all observations generated by one of the states is

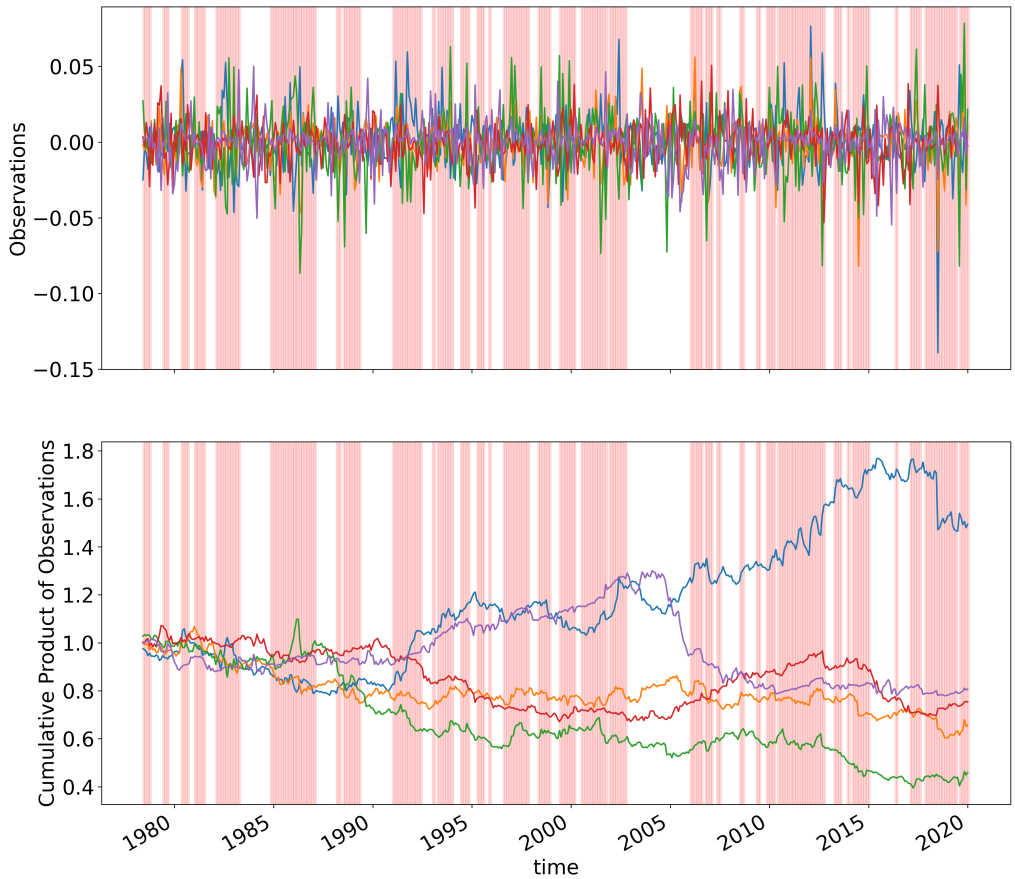

Figure 5.1. Visualization of synthetically generated data with corresponding states.

highlighted with red. We have tried to generate data that has a similar structure to the behavior of financial assets in the long-term. The observation space where the learning algorithm runs is assumed to be equivalent to the return space in finance and the cumulative product of the observations are assumed to simulate the price space. The realization we have visualized is chosen since it is able to demonstrate a case where the GGHMM performs significantly better than the GHMM.

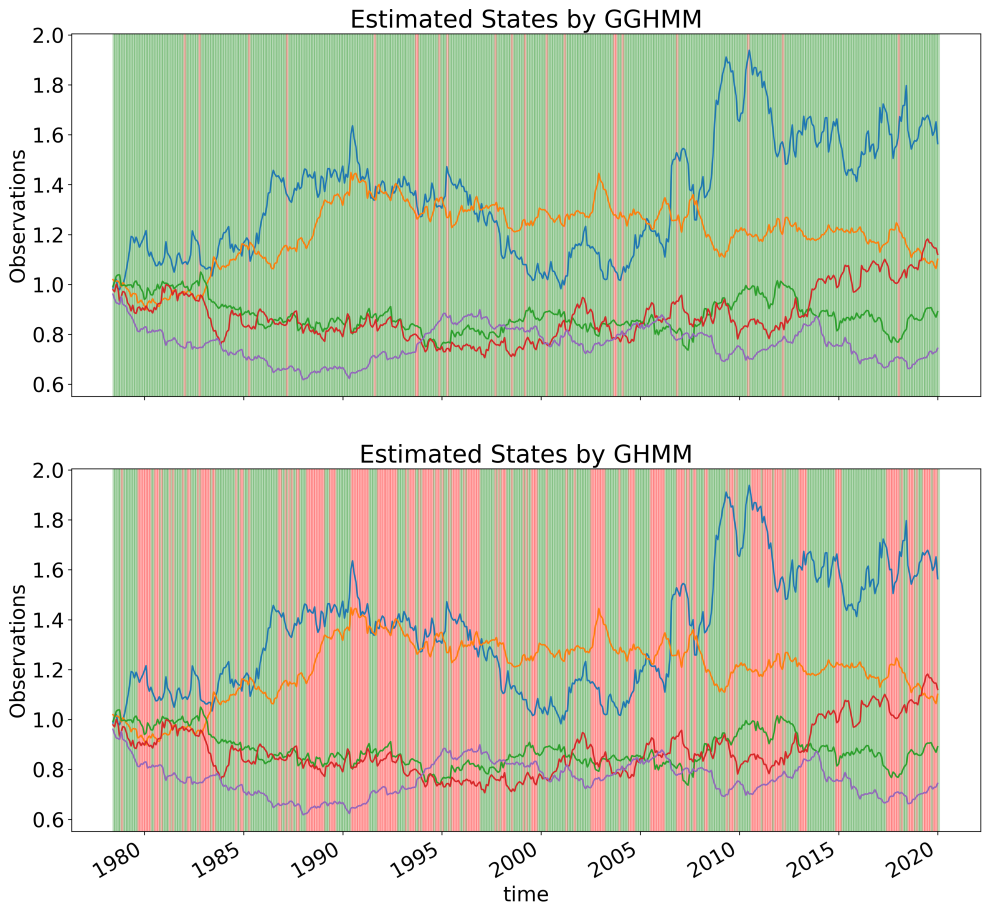

Figure 5.2. Visualization of state estimates by GGHMM and GHMM.

In Figure 5.2, we present the accuracy of estimated states with maximum like-

lihood for GGHMM and GHMM where correct estimates are highlighted with green and wrong estimates are highlighted with red. We have chosen to plot the cumulative product of the observations since it is easier to distinguish the random variables in that space. For this realization, the GGHMM has an accuracy of 0.958, while the accuracy of the GHMM is merely 0.578. There is a clear difference between the performance of each model in terms of their accuracy. Furthermore, the precision and recall metrics provide insight into how GGHMM performs better than the GHMM. Both models have quite a high precision score with 0.941 and 0.907 for GGHMM and GHMM respectively. On the other hand, there is a large difference in terms of their recall scores with 0.976 and 0.159 for GGHMM and GHMM respectively. Keeping these performance metrics in mind, when we analyze Figure 5.2, we may notice that wrong estimates by the GHMM correspond to extreme events that cause large changes. These extreme events cannot be handled by the GHMM because the model assigns zero probability that these events may actually occur. It turns out that when the mean vector of each state is relatively similar with the covariance structure creating a difference in behavior, light-tailed distributions such as the Gaussian distribution is unable to account for extreme events. We will discuss this result in detail in the next section.

Since we are using a maximum likelihood estimation of 2-dimensional hidden states, our estimation criteria becomes picking state x_i where $p(x_i) > 0.5$, which is in reality quite a low threshold for decision making. Thus, we need to study the distribution of the likelihoods that are chosen by the maximum likelihood procedure. The mean and variance of these likelihood samples may define how much confidence each model has in its maximum likelihood estimate. The GGHMM has a mean of 0.967 probability in its chosen maximum likelihood states and a standard deviation of 0.087. The GHMM has a mean of 0.967 probability in its chosen maximum likelihood states and a standard deviation of 0.092. These results show that both models assign very high likelihoods to their respective estimates. Combined with such a low recall scores, this shows that not only the GHMM fails to identify crisis states, it is also assigning high likelihoods to its wrong estimates.

5.1.3. Performance Statistics of Experiments

The performance of both models highly depend on the random data generation mechanism. As we have mentioned, the intersection of the theoretical probability distribution of each state has a great influence on the performance. This fact is actually quite intuitive when we consider the one-dimensional observation space case. If the two states have mutually exclusive probability distributions it would be easy to distinguish between the two. The problem gets progressively more difficult as their sample space intersect. Thus, we may define an easy and a hard data generation mechanisms. While both of the cases randomly assign the true parameters, for the hard case we constrain the mean behavior of both states to be close in terms of a dispersion metric. This corresponds to a unique problem where the regimes of the process have very similar average behavior while the covariance structure and the extreme events define their difference. We explore the performance of each model when the data is generated according to the hard case.

Table 5.1. Performance statistics of equally trained models.

	GGHMM		**GHMM**	
	Mean	Std	Mean	Std
Accuracy	0.849	0.150	0.907	0.145
Precision	0.782	0.126	0.831	0.171
Recall	0.727	0.131	0.911	0.207

Initially, we have recorded the results of 100 different realizations of the experiment where both of the models are trained for a maximum of 20 epochs and data is generated . The observation space is 3-dimensional and each realization has a length of 500. The results are presented in Table 5.1. We see that the GHMM has a significantly better performance in terms of mean accuracy and standard deviation in accuracy throughout the realizations. The same result carries over to their performance in terms of precision and recall. Thus, under the current conditions the GHMM is a better choice.

On the other hand, because of the increased complexity of the GGHMM when compared to the GHMM, we arguably need to train the model with more maximum training epochs in order have comparable performance statistics. It is likely that the parameters did not converge to a set which maximizes the likelihood of the incomplete data set in 20 epochs. This fact is related to the increased standard deviation in the results of the GGHMM. The convergence properties of the EM algorithm for the case of Gaussian emissions are studied in [56]. Since the GHMM is relatively simpler when compared to the GGHMM, we conclude that under equal maximum training epochs the GHMM displays better results.

Table 5.2. Performance statistics of 20 to 50 epoch realizations.

	GGHMM		**GHMM**	
	Mean	Std	Mean	Std
Accuracy	0.957	0.063	0.901	0.139
Precision	0.953	0.099	0.868	0.241
Recall	0.948	0.116	0.896	0.218

Let's investigate the effects of maximum training epochs in the performance statistics of both models. We have recorded the results of 100 different realization of the experiment where the GGHMM is trained until its results converge according to a threshold difference between each iteration and the GHMM is trained for a maximum of 20 epochs in order to study the performance of the GGHMM when it is allowed to be trained for a larger maximum training epoch. The results are shown in Table 5.2. Since we left the maximum training epochs for the GHMM unchanged, we expect that the mean and standard deviation in performance measures are not significantly different from the first set of 100 realizations. Results show that we can make a significant improvement for the GGHMM by increasing the maximum training epochs. The GGHMM has a better performance in every metric compared to both the GHMM for these realizations and the GGHMM for the previous 100 realizations. Furthermore, convergence in the GGHMM is reached in 80 epochs on average.

However, considering that a single epoch of training in GGHMM is significantly longer than a single epoch in GHMM because of the increased complexity of the calculations, both models may have their advantages. For most problems, the training time of the model is another important selection metric because of time constraints. While the GGHMM is able to display better performance, it is at the cost of significantly more training time. The assessment of how this fact affects the choice between the two models is dependent on the problem and the priorities of the researcher.

Table 5.3. Performance statistics of equally trained models in 5-dimensional observation space.

	GGHMM		**GHMM**	
	Mean	Std	Mean	Std
Accuracy	0.957	0.073	0.867	0.156
Precision	0.941	0.100	0.893	0.179
Recall	0.961	0.057	0.880	0.227

Next, we investigate the effects of the dimensionality of the observation space. Thus, we have recorded the results of 100 different realizations of the experiment with a 2-dimensional latent Markovian state-space, a 5-dimensional observation space and each realization has a length of 500. As we have established in the previous 200 realizations of the experiment, a better performance is recorded for the GGHMM when we let the model train until convergence. Since the average epochs for GGHMM was found to be 80 for convergence, we use this value as an upper bound to constrain the experiment. Therefore, we train the GHMM and GGHMM for a maximum of 20 and 80 training epochs, respectively. The results are shown in Table 5.3. Notice that we cannot show any improvement in the performance for the GHMM while the performance of the GGHMM is similarly good compared to the 3 dimensional case. Thus, we conclude that the dimensions of the observation space does not have a significant effect on the performance for these realizations.

Overall, we are able to show that the GGHMM performs better than the GHMM under specific circumstances. While both models have their advantages, there may

be some benefit in using a GGHMM for the identification of financial crises over the GHMM. On the other hand, the increased training time for the GGHMM also needs to be considered when deploying such a model in production.

5.2. Financial Data and Asset Allocation

While we are able to find a significant improvement on the state identification task for synthetically generated data sets, since financial data are generated from an unknown source that has a complex dynamical structure, it is difficult to evaluate if the dynamic structure of financial data is consistent with the specific conditions that the GGHMM performs better than the GHMM. Although in reality we may have days to train both models to their full potential, since the GGHMM theoretically requires more training epochs to yield a performance on par with an GHMM, more training epochs will be allocated to the GGHMM to present representative results in real data.

Our data set consists of the daily returns of Istanbul Stock Exchange National 100 Index, USD/TRY exchange rates, gold exchange rate and interest rate reported by Central Bank of the Republic of Turkey and Nasdaq Composite Index starting from 2007 to the end of April 2020. These financial instruments are carefully chosen in order to represent assets that are readily available to an investor in Turkey. The historical returns of each asset in Turkish lira is shown in the cumulative return space in Figure 5.3.

5.2.1. Regime Identification

In this section, we investigate how well the GGHMM performs in regime shift identification compared to the GHMM. Thus, we use the complete data set for training both models and visualize the identified regimes. The results of this section also present an upper limit to the performance of each model in the asset allocation section. We will also present some of the parameters learned by both models in order to discuss the structure of financial data sets. Both models use weekly resampled data sets

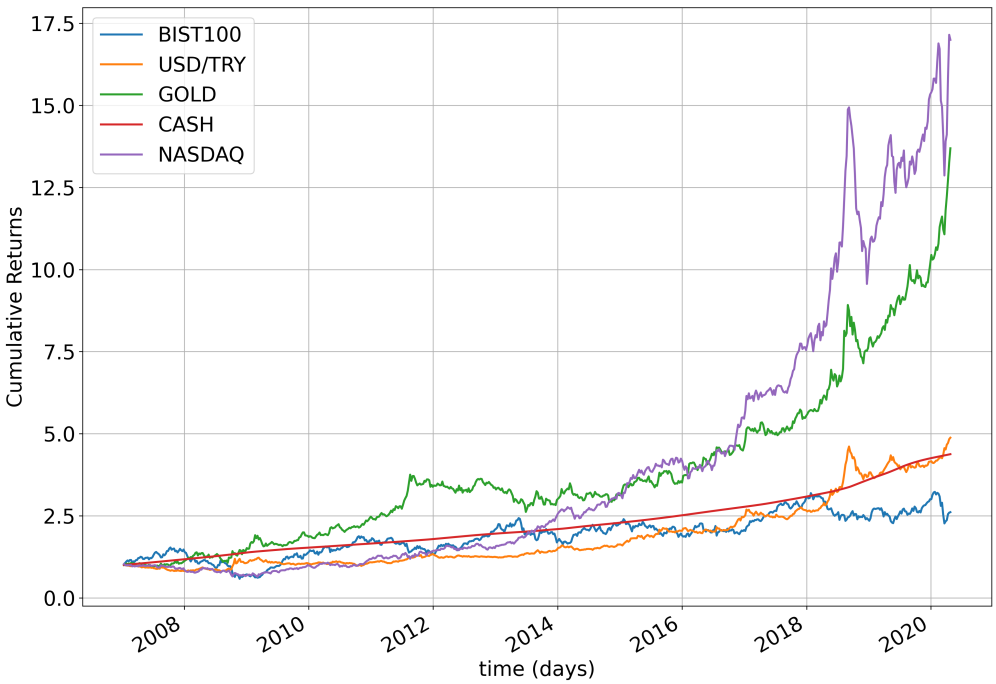

Figure 5.3. Cumulative returns of each asset.

since identifying regime shifts in between weeks does not make economical sense and maximum training epochs are set to 75.

The learning algorithm for both models are designed to be unbiased towards data sets, which means that there are no underlying assumptions about the particular financial data other than the structural assumptions of the models. Furthermore, we assume that the transition parameters π and A as well as the emission parameters of each state μ_i, Σ_i and ν_i are unknown and must be learned from data. While we do not impose any assumptions on how the models should cluster the time series data, our first observation is that the most identifiable difference between clusters is their magnitude of standard deviations. Thus, since we will be assuming that the latent state space is 2-dimensional, the corresponding states will be referred to as the turbulent and non-turbulent states for clarification.

The states identified by the GHMM are shown in Figure 5.4. The model is able to clearly identify the regime shifting behavior of financial markets. The states that are highlighted by red correspond to the financial crisis of 2007-2008 and the Turkish currency and debt crisis that occurred in 2018. Particularly, the first crisis period identified by the model starts from January 12th of 2007 to March 13th of 2009 and the second crisis identified starts from May 4th of 2018 to October 25th of 2019. However, the model does not identify the COVID-19 crisis that occured in 2020 as a turbulent regime. This is arguably because of the fact that this was not an economical crisis but a global pandemic that had unprecedented behavioral characteristics.

In order to have a more detailed understanding of the results, let's start by presenting μ_i's learned by the model which are annualized for a more intuitive understanding. The annualized mean returns for the turbulent and non-turbulent regimes are shown in Table 5.4. There is a significant difference in the returns of equities between the two states. We see that BIST100 has an annual 9% loss during the turbulent regime. Furthermore, NASDAQ has a significantly lower annual return when compared to the interest rates in Turkey during the turbulent regime despite being

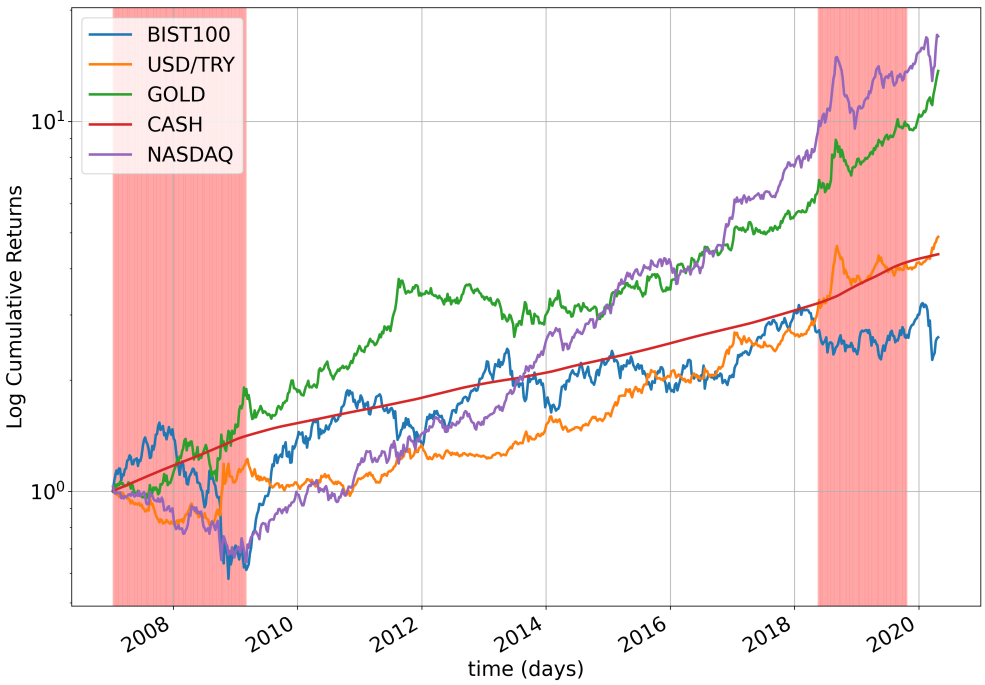

Figure 5.4. State identification with smoothed estimates of GHMM.

priced in USD. An interpretation of this behavior could be that the GHMM defines turbulent regimes in terms of periods where equities are not profitable.

Table 5.4. Annualized Mean Return Results for GHMM.

	Turbulent	**Non-Turbulent**
BIST100	-0.091	0.18
USD/TRY	0.16	0.12
GOLD	0.32	0.18
CASH	0.17	0.090
NASDAQ	0.066	0.30

Let's present the annualized standard deviation of each asset as well as their correlations for each state in Table 5.5 and 5.6. As expected, the standard deviations of assets are higher during the turbulent regime. However, the correlations between assets seem to have similar dynamics. While BIST100 and NASDAQ have a slightly positive correlation during non-turbulent regimes, this relationship reverses to have a slightly negative correlation during turbulent regimes. This is likely because of the fact that the NASDAQ index is priced in terms of USD which almost has a perfect negative correlation with BIST100. The fact that the correlation between USD/TRY and NASDAQ increase in turbulent regimes reinforces our argument.

The states identified by the GGHMM are shown in Figure 5.5. Visually there is no significant difference from the GHMM in state identification. Similarly, the model is able to identify the regime shifting behavior of financial markets. The start and end dates of identified financial crises are slightly different. The first crisis period identified by the model starts from January 12th of 2007 to February 27th of 2009 and the second crisis identified starts from May 18th of 2018 to October 18th of 2019. Unfortunately, the GGHMM is also unable to identify the COVID-19 crisis as a turbulent regime.

The values of the learned parameter μ_i which correspond to annualized mean returns for the turbulent and non-turbulent regimes seem to be significantly different

Table 5.5. Covariance characteristics learned by GHMM for the turbulent state.

		Correlations			
	Standard Deviations	**BIST100**	**USD/TRY**	**GOLD**	**CASH**
Turbulent					
BIST100	0.32				
USD/TRY	0.21	-0.94			
GOLD	0.27	-0.79	0.73		
CASH	0.0031	0.78	-0.92	-0.59	
NASDAQ	0.25	-0.10	0.36	-0.14	-0.67

Table 5.6. Covariance characteristics learned by GHMM for the non-turbulent state.

		Correlations			
	Standard Deviations	**BIST100**	**USD/TRY**	**GOLD**	**CASH**
Non-Turbulent					
BIST100	0.24				
USD/TRY	0.11	-0.93			
GOLD	0.17	-0.69	0.62		
CASH	0.0020	0.95	-0.97	-0.55	
NASDAQ	0.18	0.07	0.17	-0.17	-0.21

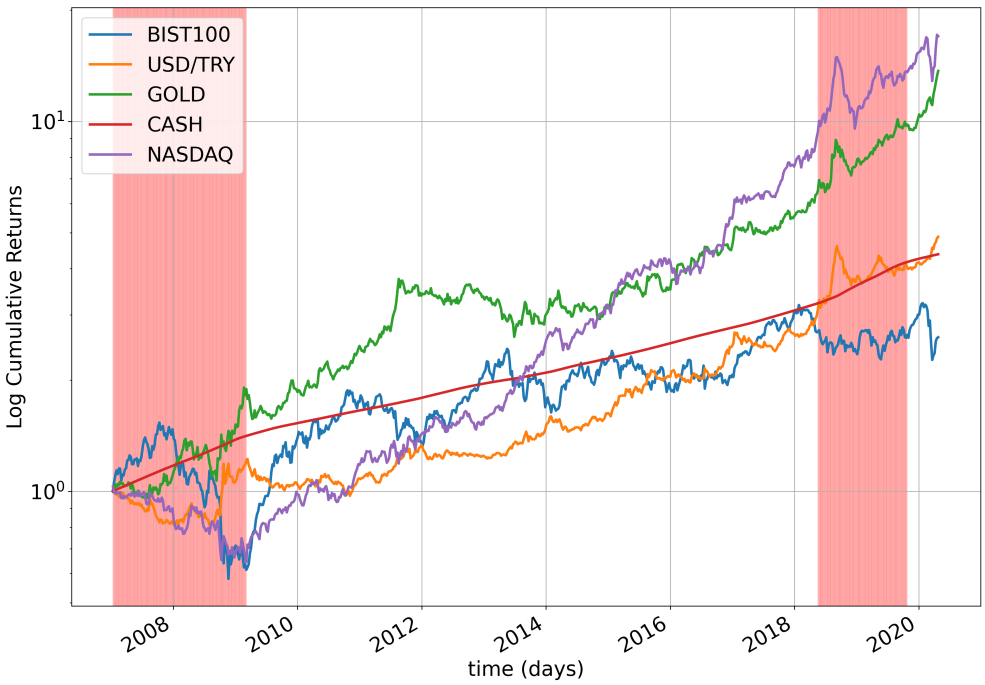

Figure 5.5. State identification with smoothed estimates of GGHMM.

from the previous means learned by the GHMM. The results are reported in Table 5.7. Let's start by noting that the annualized returns for cash which is equivalent to the interest rate, is found to be exactly the same. This is likely since interest rates have a significantly smaller standard deviation when compared to other asset classes. The most significant difference seems to be in the annualized mean of USD/TRY exchange rate. While the GHMM finds large positive returns for the exchange rate at both regimes, the GGHMM reports that in reality the mean value is much closer to zero. A financial interpretation of this observation is that large positive returns in the exchange rate are in reality extreme events. Since GHMM cannot handle extreme events it inevitably shifts the mean towards extreme values but the GGHMM classifies these events as extreme. Thus, using the smoothed estimates found from the GGHMM may enable investors to make more informed decisions about the risk and return characteristics of assets.

Table 5.7. Annualized Mean Return Results for GGHMM.

	Turbulent	**Non-Turbulent**
BIST100	0.045	0.23
USD/TRY	0.032	0.074
GOLD	0.24	0.15
CASH	0.17	0.090
NASDAQ	-0.0023	0.29

Let's present the annualized standard deviation of each asset as well as their correlations for each state in Table 5.8 and 5.9. Similar to the GHMM results, the standard deviations of assets are higher during the turbulent regime. One interesting observation is that the Istanbul stock exchange index becomes significantly more negatively correlated to gold index during turbulent states. Furthermore, the USD/TRY exchange currency rate and gold becomes more correlated during such periods. This observation is consistent with our financial intuition that investors prefer to sell equities and buy gold and USD during volatile financial periods in Turkey. It is also worth to note the degrees of freedom parameter, ν, for each state. The parameter takes a value

of 5.30 and 9.27 for the turbulent and non-turbulent regimes, respectively. Since both ν values are larger than 2, the variance for both distributions is defined. Furthermore, we know that the Student's t-distribution converges to the Gaussian distribution as ν increases, then we can conclude that the turbulent regime has a more heavy-tailed distribution.

Table 5.8. Covariance characteristics learned by GGHMM for the turbulent state.

		Correlations			
	Standard Deviations	**BIST100**	**USD/TRY**	**GOLD**	**CASH**
Turbulent					
BIST100	0.33				
USD/TRY	0.19	-0.89			
GOLD	0.26	-0.80	0.69		
CASH	0.0033	0.85	-0.98	-0.73	
NASDAQ	0.26	-0.0014	0.38	-0.17	-0.46

Table 5.9. Covariance characteristics learned by GGHMM for the non-turbulent state.

		Correlations			
	Standard Deviations	**BIST100**	**USD/TRY**	**GOLD**	**CASH**
Non-Turbulent					
BIST100	0.23				
USD/TRY	0.11	-0.93			
GOLD	0.17	-0.68	0.58		
CASH	0.0021	0.96	-0.99	-0.60	
NASDAQ	0.18	-0.026	0.28	-0.17	-0.27

When we compare the standard deviation values learned by the GHMM and GGHMM for each state we see that the numeric values of these standard deviations are quite similar. However, we would like to emphasize that the shape of a heavy-tailed distribution and a light-tailed distribution such a Gaussian with equal variance values

can significantly differ. This fact is visualized in Figure 5.6. While the heavy-tailed density has a narrower distribution around the mean, it has a higher density around the tails. We claim that this is a more accurate representation of financial return series. While most return values are measured around the mean value, very extreme deviations from this mean value can occur.

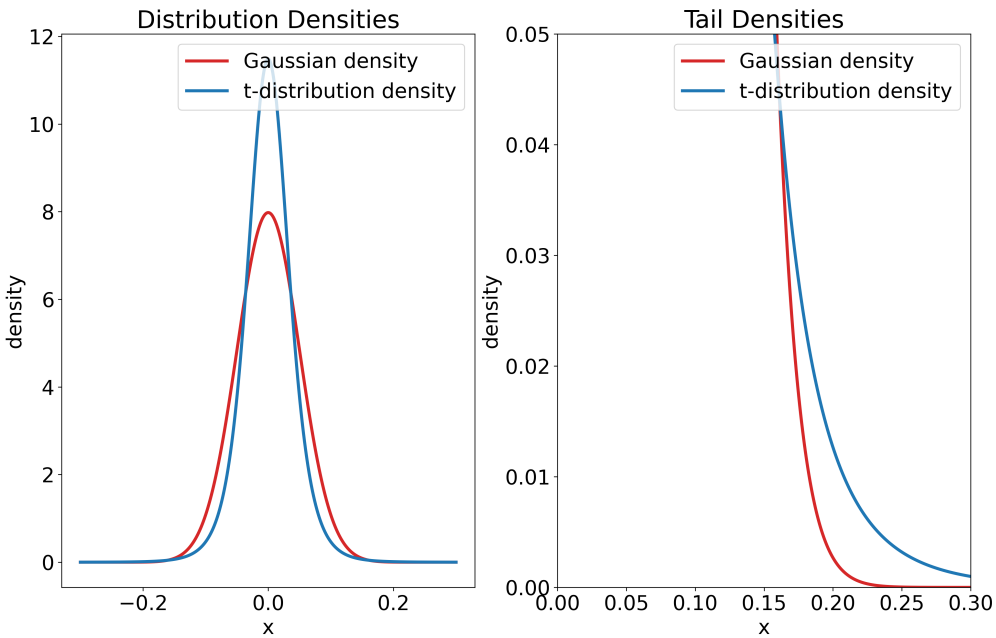

Figure 5.6. Difference in densities of equal variance distributions.

5.2.2. Asset Allocation

For the asset allocation task, we should disclaim that our aim is not to optimize the models in order to generate a profitable investment strategy but to present how these models can be used in asset allocation. We should note that our whole data set consists of slightly more than 13 years of data which cannot be representative of long-term asset relations. We use an expanding window that initially contains 10 years of data starting from 2007 to the end of 2016. The parameters are retrained annually and the asset allocation is updated every week which is called a rebalance in the finance literature. Our aim is to learn the expected return and standard deviation

of the assets while taking the sequential nature of the data into account. Furthermore, it would be beneficial to learn the dynamics of a financial crisis and use our model in asset allocation to reduce the impact of the crisis. The financial crisis of 2007-2008 is conveniently included in our initial training window. At least two other accepted financial crises are present in the test window including the Turkish currency and debt crisis that occurred in 2018 and COVID-19 crisis that occurred in 2020.

For both models the mean-variance optimization algorithm uses an annual maximum standard deviation of 6% when calculating optimal portfolio weights. However, while this choice is somewhat arbitrary, other values yield similar results in comparison. We expect that since the allocation period is out-of-sample, the portfolio standard deviation will exceed this amount. Let's start by presenting the results of the backtest that uses the GHMM in order to estimate the return characteristics of assets and make decisions based on the maximum likelihood state identified by the model. The results are shown in Figure 5.7. The results are shown beside a portfolio constructed by using historical sample expected return and covariance for comparison.

The portfolio constructed with a GHMM has an annualized return of 23% and annualized standard deviation of 7.8%. Another measure of risk used in investment management is the maximum drawdown which is defined as the maximum percentage loss the portfolio has experienced. In this case, the GHMM portfolio has a maximum drawdown of 8.9%. For comparison, the benchmark has an annualized return of 25% and an annualized standard deviation of 8.8%. The maximum drawdown of the benchmark is 10%.

In order to compare two portfolios, a frequently used measure is the Sharpe ratio which is also related to the efficient frontier of a mean-variance optimized portfolio [57]. The Sharpe ratio is calculated as:

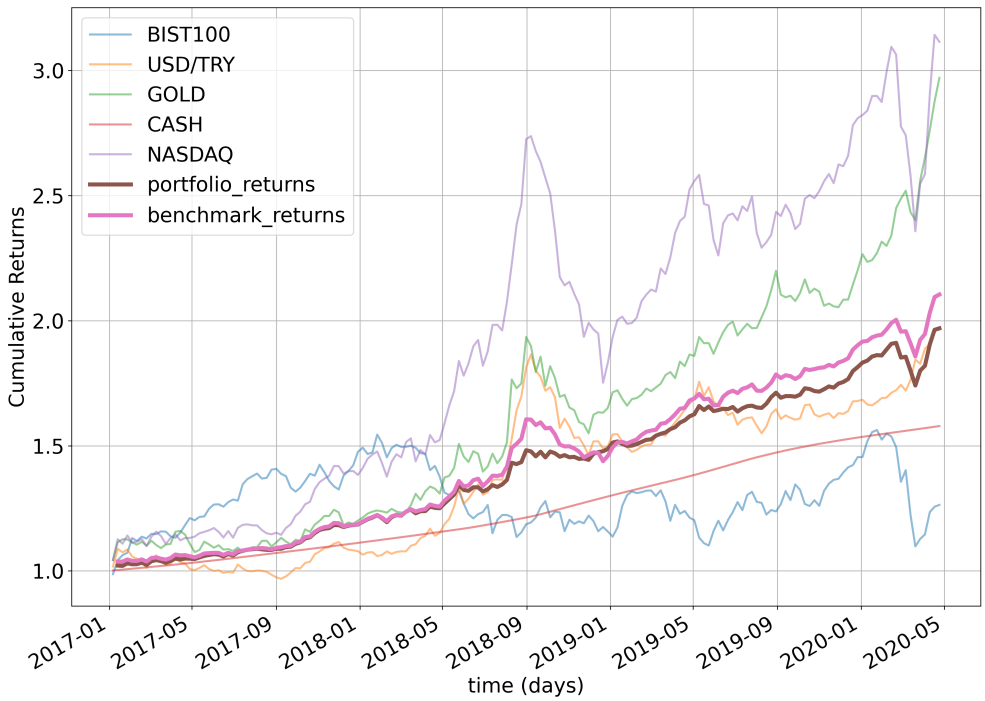

Figure 5.7. Asset allocation based on GHMM.

$$S = \frac{r_p - r_f}{\sigma_p} \tag{5.4}$$

where r_p and σ_p are the annualized return and standard deviation of the portfolio. r_f stands for the annualized return of a risk-free portfolio where we assume that a risk-free portfolio consists of returns that can be gained from investing in a interest rate based asset such as a deposit account. We can interpret this ratio as the expected return that is gained per unit risk taken. In terms of Sharpe ratio, we see that the GHMM portfolio is a slightly worse investment choice with a 0.25 compared to the benchmark portfolio which has a 0.31.

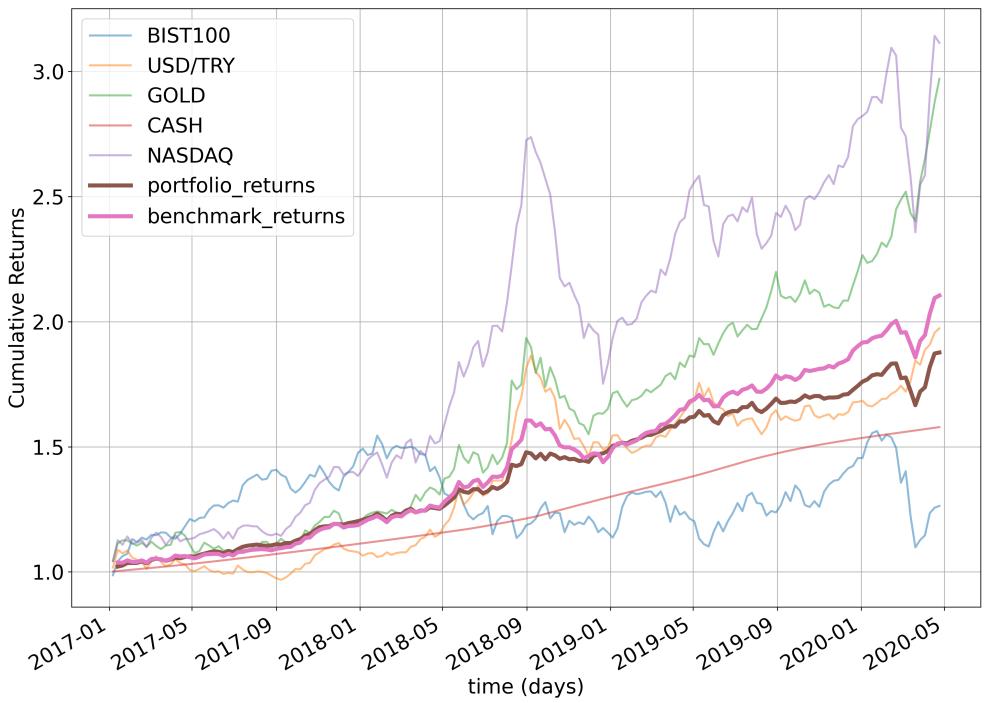

Figure 5.8. Asset allocation based on GGHMM.

We present the results of the backtest that uses the GGHMM in Figure 5.8. The portfolio constructed with a GGHMM has an annualized return of 21%, annualized standard deviation of 7.6% and a maximum drawdown of 9%. The Sharpe ratio for the portfolio is found to be 0.20.

In comparison, we are unable to show significant differences between using the GGHMM and GHMM in asset allocation. Furthermore, while we have shown that both models are able to identify market regimes, the prediction of market regime does not immediately result in better asset allocation strategies compared to using sample mean and covariance in mean-variance optimization.

6. CONCLUSION

In this work, we have introduced non-stationary dynamical systems along with statistical models and inference methods to understand their behavior. We have focused our coverage of the topics discussed in this work to understanding the non-stationary dynamics of financial markets, since they pose a significant challenge to researchers in computational fields and even incremental advancements may be highly lucrative.

We have emphasized our prior knowledge on financial markets in order to justify the importance of each section in this work. In summary, the sequential structure of data should be taken into account for time series data sets. The sequential dependencies of these data sets can be simplified by assuming that they satisfy the Markov property. However, this assumption may be overly simplistic and fails to capture long range dependencies between sequences. In these cases the state-space model can be used to impose a dependency between all data points in a sequential set while also taking advantage of the simplifications offered by Markov chains.

The hidden Markov model, which is a special case of state-space models where the latent layer is assumed to take on values in a discrete space, is introduced in this work as a fundamental model used for many non-stationary dynamical systems in practise. Since we have focused our coverage on financial signals, we assume that the observation density of the hidden Markov model is normally distributed and the state and parameter inference algorithms are presented accordingly. Our proposed model can be considered as a variant of the GHMM where we increase the complexity of the model in order to accommodate our prior knowledge on financial signals. The increased complexity of our model can be efficiently handled by formulating the observation density as an exponential family, thus we additionally review the basic properties of exponential families before we study our proposed model.

We have presented the Gaussian-Gamma hidden Markov model which can poten-

tially be used to model various non-stationary dynamic systems with multiple regimes and heavy-tailed distributions. Our proposed model is capable of representing heteroscedastic processes within each state and is highly flexible. Furthermore, the model is mostly analytically tractable aside from requiring an auxiliary root finding algorithm. This allows it to be trained relatively quickly compared to the state-of-the-art deep neural networks and requires much less data in order to be trained.

In order to demonstrate the performance of our model in financial applications, we have introduced the standard portfolio optimization framework and its drawbacks with respect to modelling time series data. Using the GHMM and GGHMM, these drawbacks can be mitigated. Thus, we introduce some simple adjustments to the standard portfolio optimization framework in order to use the GHMM and GGHMM in conjunction with portfolio optimization. We have shown the application of GGHMM in state identification for both synthetically generated data and real financial data. Results show that the GGHMM has a performance comparable with the GHMM in the state identification task. In addition to its identification performance, we have shown that the parameters learned by the model are capable of representing heavy-tailed data sets.

While we cannot show superior performance in the asset allocation task, in the context of finance 3 years of testing data is insufficient for a conclusive result in this task. Decades of historical data are required for a significant result to be presented. Such data sets are available in U.S. stock and bond markets and should be used to have a better understanding of the capabilities of the GGHMM.

Another drawback of our model is that we have used the expected value of the random scale variable that determines the current volatility of the system in order to present a simple asset allocation framework and keep the model analytically tractable. However, in practise, more research is required into the estimation of the latent variable. Such research requires the calculation of intractable likelihoods which leads to the use of approximate inference methods. Monte Carlo sampling based methods called particle

filters are especially suitable in this context. The interested reader should consult references [58], [59], [60] and [61] for an introduction.

In terms of improvements to the GGHMM, one obvious way of improvement can come from explicitly modelling the dependencies between the sequential latent scale variables. However, since this would introduce intractable calculations we have left it as a future research project. Another source of improvement may be to also learn the number latent states in the system. Such models use the hierarchical Dirichlet process as their latent representation of the state transition system [62]. For an unknown number of states, the intuition that the states are persistent in financial systems can be modelled using the work in [63]. For more efficient learning in such models, practical considerations are presented in [64]. To conclude, we believe that our proposed model along with its application in the task of portfolio management offers a way of integrating our current knowledge about the statistical properties of financial markets with the established framework of mean-variance analysis in asset allocation.

REFERENCES

1. Zuckerman, G., *The Man Who Solved the Market: How Jim Simons Launched the Quant Revolution*, Penguin, UK, 2019.

2. Markowitz, H., "Portfolio Selection", *The Journal of Finance*, Vol. 7, No. 1, pp. 77–91, Mar. 1952.

3. Cootner, P. H., *The random character of stock market prices*, Cambridge, Mass., M.I.T. Press, 1964.

4. Ang, A. and A. G. Timmermann, *Regime Changes and Financial Markets*, CEPR Discussion Papers 8480, C.E.P.R. Discussion Papers, Jul. 2011.

5. Ait-Sahalia, Y. and D. Xiu, "Increased correlation among asset classes: Are volatility or jumps to blame, or both?", *Journal of Econometrics*, Vol. 194, No. 2, pp. 205 – 219, 2016, financial Statistics and Risk Management.

6. Rabiner, L. R., *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, p. 267–296, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990.

7. Durbin, R., S. R. Eddy, A. Krogh and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, 1998.

8. Smyth, P., "Clustering Sequences with Hidden Markov Models", *Proceedings of the 9th International Conference on Neural Information Processing Systems*, NIPS'96, p. 648–654, MIT Press, Cambridge, MA, USA, 1996.

9. Bishop, C. M., *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag, Berlin, Heidelberg, 2006.

10. Kantas, N., A. Doucet, S. Singh and J. Maciejowski, "An Overview of Sequential Monte Carlo Methods for Parameter Estimation in General State-Space Models", *Proc. International Federation of Automatic Control Meet. System Identification*, Vol. 15, 01 2009.

11. Doucet, A., S. Godsill and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering", *Statistics and Computing*, Vol. 10, No. 3, p. 197–208, Jul. 2000.

12. Jacquier, E., N. G. Polson and P. E. Rossi, "Bayesian Analysis of Stochastic Volatility Models", *Journal of Business and Economic Statistics*, Vol. 12, No. 4, pp. 371–389, 1994.

13. Kritzman, M., S. Page and D. Turkington, "Regime Shifts: Implications for Dynamic Strategies (corrected)", *Financial Analysts Journal*, Vol. 68, No. 3, pp. 22–39, 2012.

14. Kinlaw, W. and D. T. Mark P. Kritzman, *A Practitioner's Guide to Asset Allocation*, Wiley, 2017.

15. Zhou, X. and G. Yin, "Markowitz's Mean-Variance Portfolio Selection with Regime Switching: A Continuous-Time Model", *SIAM J. Control and Optimization*, Vol. 42, pp. 1466–1482, 01 2003.

16. Zhang, M. and P. Chen, "Mean-variance portfolio selection with regime switching under shorting prohibition", *Operations Research Letters*, Vol. 44, 08 2016.

17. Cemgil, A. T., C. Fevotte and S. J. Godsill, "Variational and stochastic inference for Bayesian source separation", *Digital Signal Processing*, Vol. 17, No. 5, pp. 891 – 913, 2007, special Issue on Bayesian Source Separation.

18. Chatzis, S., D. Kosmopoulos and T. Varvarigou, "Robust Sequential Data Modeling Using an Outlier Tolerant Hidden Markov Model", *IEEE transactions on*

pattern analysis and machine intelligence, Vol. 31, pp. 1657–69, 10 2009.

19. Bulla, J., "Hidden Markov models with t components. Increased persistence and other aspects", *Quantitative Finance*, Vol. 11, No. 3, pp. 459–475, 2011.

20. Zhang, H., Q. M. Jonathan Wu and T. M. Nguyen, "Modified student's t-hidden Markov model for pattern recognition and classification", *IET Signal Processing*, Vol. 7, No. 3, pp. 219–227, 2013.

21. Bernardi, M., A. Maruotti and L. Petrella, "Multivariate Markov-Switching models and tail risk interdependence", online, 07 2018, `https://arxiv.org/pdf/1312.6407.pdf`, accessed in June 2020.

22. Barber, D., A. T. Cemgil and S. Chiappa, *Bayesian Time Series Models*, Cambridge University Press, USA, 2011.

23. Norris, J. R., *Markov Chains*, Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press, 1997.

24. Andrieu, C., N. Freitas, A. Doucet and M. Jordan, "An Introduction to MCMC for Machine Learning", *Machine Learning*, Vol. 50, pp. 5–43, 01 2003.

25. Box, G., G. Jenkins and C. Reinsel, *Time Series Analysis: Forecasting and Control: Fourth Edition*, Wiley, 2013.

26. Andrew W. Lo, A. C. M., *A Non-Random Walk Down Wall Street*, Princeton University Press, 1999.

27. Hyndman, R. and G. Athanasopoulos, *Forecasting: principles and practice*, OTexts, 2014.

28. Lütkepohl, H., *New introduction to multiple time series analysis*, Springer, 2005.

29. Sims, C. A., "Macroeconomics and Reality", *Econometrica*, Vol. 48, No. 1, pp. 1–48, 1980.

30. Barber, D. and A. T. Cemgil, "Graphical Models for Time-Series.", *IEEE Signal Process. Mag.*, Vol. 27, No. 6, pp. 18–28, 2010.

31. Fama, E. F., "Efficient Capital Markets: A Review of Theory and Empirical Work", *The Journal of Finance*, Vol. 25, 1970.

32. Shleifer, A., *Inefficient Markets: An Introduction to Behavioral Finance*, Oxford University Press, 2000.

33. Lloyd, S., "Least squares quantization in PCM", *IEEE Transactions on Information Theory*, Vol. 28, No. 2, pp. 129–137, mar 1982.

34. Bilmes, J., "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models", *Technical Report ICSI-TR-97-021, University of Berkeley*, Vol. 4, 06 2000.

35. van der Maaten, L. and G. Hinton, "Visualizing Data using t-SNE", *Journal of Machine Learning Research*, Vol. 9, pp. 2579–2605, 2008.

36. Mandelbrot, B., "The Variation of Some Other Speculative Prices", *The Journal of Business*, Vol. 40, No. 4, pp. 393–413, 1967.

37. Engle, R. F., "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation", *Econometrica*, Vol. 50, No. 4, pp. 987–1007, 1982.

38. Bollerslev, T., "Generalized autoregressive conditional heteroskedasticity", *Journal of Econometrics*, Vol. 31, No. 3, pp. 307 – 327, 1986.

39. Sipos, I. R., A. Ceffer and J. Levendovszky, "Parallel Optimization of Sparse Port-

folios with AR-HMMs", *Computational Economics*, Vol. 49, No. 4, pp. 563–578, April 2017.

40. Petropoulos, A., S. Chatzis and S. Xanthopoulos, "A Novel Corporate Credit Rating System Based on Student's-t Hidden Markov Models", *Expert Systems with Applications*, Vol. 53, 01 2016.

41. Jelinek, F., *Statistical Methods for Speech Recognition*, MIT Press, Cambridge, MA, USA, 1998.

42. Stigler, J., F. Ziegler, A. Gieseke, J. C. M. Gebhardt and M. Rief, "The Complex Folding Network of Single Calmodulin Molecules", *Science*, Vol. 334, No. 6055, pp. 512–516, 2011.

43. Cover, T. M. and J. A. Thomas, *Elements of Information Theory*, John Wiley and Sons, Inc., 1991.

44. Wu, C. F. J., "On the Convergence Properties of the EM Algorithm", *Ann. Statist.*, Vol. 11, No. 1, pp. 95–103, 03 1983, `https://doi.org/10.1214/aos/1176346060`.

45. Robert, C. P. and G. Casella, *Monte Carlo Statistical Methods (Springer Texts in Statistics)*, Springer-Verlag, Berlin, Heidelberg, 2005.

46. Mehrabi, N., F. Morstatter, N. Saxena, K. Lerman and A. Galstyan, "A Survey on Bias and Fairness in Machine Learning", online, 09 2019, `https://arxiv.org/pdf/1908.09635.pdf`, accessed in June 2020.

47. Wang, S. X., *Maximum weighted likelihood estimation*, Ph.D. Thesis, 2001.

48. Beyaztas, U. and H. L. Shang, "Forecasting functional time series using weighted likelihood methodology", *Journal of Statistical Computation and Simulation*, 08 2019.

49. Blei, D., "The Exponential Family", online, 11 2016, www.cs.columbia.edu/ blei/fogm/2016F/doc/exponential_families.pdf, accessed in June 2020.

50. Shoham, S., "Robust clustering by deterministic agglomeration EM of mixtures of multivariate t-distributions", *Pattern Recognition*, Vol. 35, pp. 1127–1142, 05 2002.

51. Hassan, M. R. and B. Nath, "Stock market forecasting using hidden Markov model: a new approach", *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*, pp. 192–196, 2005.

52. Guidolin, M. and A. Timmermann, "Asset allocation under multivariate regime switching", *Journal of Economic Dynamics and Control*, Vol. 31, No. 11, pp. 3503 – 3544, 2007.

53. Novak, S., *Extreme Value Methods with Applications to Finance*, CRC Press, Boca Raton, FL, 2011.

54. Taleb, N. N., "The statistical consequences of fat tails: Papers and commentary", *The technical inconcerto*, Vol. 1, 2018.

55. Chauvet, M. and J. Piger, "A comparison of the real-time performance of business cycle dating methods", *Journal of Business and Economic Statistics*, pp. 42–49, 2008.

56. Xu, L. and M. I. Jordan, "On Convergence Properties of the Em Algorithm for Gaussian Mixtures", *Neural Comput.*, Vol. 8, No. 1, p. 129–151, Jan. 1996.

57. Sharpe, W. F., "Mutual Fund Performance", *The Journal of Business*, Vol. 39, No. 1, pp. 119–138, 1966.

58. Godsill, S., "Particle Filtering: the First 25 Years and beyond", *ICASSP 2019* -

2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7760–7764, 2019.

59. Yildirim, S., S. S. Singh, T. Dean and A. Jasra, "Parameter Estimation in Hidden Markov Models With Intractable Likelihoods Using Sequential Monte Carlo", *Journal of Computational and Graphical Statistics*, Vol. 24, No. 3, pp. 846–865, 2015.

60. Doucet, A., N. d. Freitas, K. P. Murphy and S. J. Russell, "Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks", *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, UAI '00, p. 176–183, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000.

61. Andrieu, C., A. Doucet and R. Holenstein, "Particle Markov chain Monte Carlo methods", *Journal of the Royal Statistical Society Series B*, Vol. 72, No. 3, pp. 269–342, 2010.

62. Teh, Y. W., M. I. Jordan, M. J. Beal and D. M. Blei, "Hierarchical Dirichlet Processes", *Journal of the American Statistical Association*, Vol. 101, No. 476, pp. 1566–1581, 2006.

63. Fox, E., E. Sudderth, M. Jordan and Willsky, *The sticky HDP-HMM: Bayesian nonparametric hidden Markov models with persistent states*, Tech. Rep. 2, MIT Laboratory for Information & Decision Systems, Cambridge, MA 02139, 01 2007.

64. Ulker, Y., B. Gunsel and A. T. Cemgil, "Annealed SMC Samplers for Nonparametric Bayesian Mixture Models", *IEEE Signal Processing Letters*, Vol. 18, No. 1, pp. 3–6, 2011.

APPENDIX A: PROBABILITY DENSITY REFRESHER

A.1. Multivariate Gaussian Distribution

Assuming we have a k-dimensional data set, the multivariate Gaussian distribution is defined as:

$$f_x(x_1, \ldots, x_k) = \frac{1}{\sqrt{(2\pi)^k |\Sigma|}} \exp\Big(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu) \Big) \tag{A.1}$$

A.2. Gamma Distribution

$$f_x(\alpha, \beta) = \frac{\beta^\alpha x^{\alpha-1}}{\Gamma(\alpha)} \exp\Big(-\beta x \Big) \tag{A.2}$$

APPENDIX B: MULTIVARIATE T-DISTRIBUTION DERIVATION

Let's start by introducing a relevant property of determinants. For an nxn matrix A, the following property holds:

$$\det(c\mathrm{A}) = c^n \det(\mathrm{A}) \tag{B.1}$$

Then, using this property we can scale a covariance matrix Σ by dividing it with a scalar value w and the determinant becomes:

$$\det\Big(\frac{\Sigma}{w}\Big) = w^{-n}\det(\Sigma) \tag{B.2}$$

Using this scaled covariance matrix, the density of a multivariate Gaussian becomes:

$$f_x(x_1, \ldots, x_k) = \frac{\sqrt{\omega^k}}{\sqrt{(2\pi)^k|\Sigma|}}\exp\Big(-\frac{1}{2}\omega(x-\mu)^T\Sigma^{-1}(x-\mu)\Big) \tag{B.3}$$

Now, we assume that w is a random variable and it has a prior Gamma distribution. This becomes a Gaussian scale mixture where there are multiple Gaussian distributions with equal location parameters but varying scale parameters. Since we

do not know the value of the scale parameters, we are interested in integrating w out of the density and obtain a mixture density that is a weighted average over all possible scale values as shown in equation B.4.

$$\int_0^\infty dw \frac{\sqrt{\omega^k}}{\sqrt{(2\pi)^k|\Sigma|}} \exp\Big(-\frac{1}{2}\omega(x-\mu)^T\Sigma^{-1}(x-\mu)\Big) \frac{\beta^\alpha x^{\alpha-1}}{\Gamma(\alpha)} \exp\Big(-\beta x\Big) \tag{B.4}$$

$$= \frac{1}{\sqrt{(2\pi)^k|\Sigma|}} \frac{\beta^\alpha}{\Gamma(\alpha)} \int_0^\infty dw\, w^{\frac{k}{2}+\alpha-1} \exp\Bigg(-\Big[\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)+\beta\Big]w\Bigg)$$

In order to calculate the integral in equation B.4, we use the following identity:

$$\int_0^\infty x^a \exp\Big(-bx\Big) dx = b^{-a-1}\Gamma(a+1) \tag{B.5}$$

where the real part of b must be greater than 0 and the real part of a must be greater than -1. Then equation B.4 becomes:

$$\frac{1}{\sqrt{(2\pi)^k|\Sigma|}} \frac{\beta^\alpha}{\Gamma(\alpha)} \int_0^\infty dw\, w^{\frac{k}{2}+\alpha-1} \exp\Bigg(-\Big[\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)+\beta\Big]w\Bigg) \tag{B.6}$$

$$= \frac{\beta^\alpha\Gamma(\alpha+\frac{k}{2})}{\sqrt{(2\pi)^k|\Sigma|}\Gamma(\alpha)} \Big[\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)+\beta\Big]^{-(\alpha+\frac{k}{2})}$$

Let's assume that the value of both α and β are equal to $\frac{\nu}{2}$. If we insert these

values into equation B.6, we obtain:

$$\frac{(\frac{\nu}{2})^{(\frac{\nu}{2})}\Gamma(\frac{\nu+k}{2})}{\sqrt{(2\pi)^k|\Sigma|}\Gamma(\frac{\nu}{2})}\left[\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)+\frac{\nu}{2}\right]^{-(\frac{\nu+k}{2})} \tag{B.7}$$

The density shown in equation B.7 is actually equivalent to the density of the multivariate t-distribution. We can show this fact by multiplying the density by 1 as shown below:

$$\begin{aligned}&\frac{(\frac{\nu}{2})^{-(\frac{\nu+k}{2})}}{(\frac{\nu}{2})^{-(\frac{\nu+k}{2})}}\frac{(\frac{\nu}{2})^{(\frac{\nu}{2})}\Gamma(\frac{\nu+k}{2})}{\sqrt{(2\pi)^k|\Sigma|}\Gamma(\frac{\nu}{2})}\left[\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)+\frac{\nu}{2}\right]^{-(\frac{\nu+k}{2})}\\ &=\frac{\Gamma(\frac{\nu+k}{2})}{\Gamma(\frac{\nu}{2})\nu^{\frac{k}{2}}\pi^{\frac{k}{2}}|\Sigma|^{\frac{1}{2}}}\left[\frac{1}{\nu}(x-\mu)^T\Sigma^{-1}(x-\mu)+1\right]^{-(\frac{\nu+k}{2})}\end{aligned} \tag{B.8}$$

where the last line is the k-dimensional multivariate t-distribution.