Modelling Nonlinear Stochastic Dynamics in Financial Time Series

RAGNAR HAGEN LESCH

Doctor of Philosophy

ASTON UNIVERSITY, BIRMINGHAM

May 2000

Aston University, Birmingham

Modelling Nonlinear Stochastic Dynamics in Financial Time Series

Ragnar Hagen Lesch

Doctor of Philosophy, 2000

Thesis Summary

For analysing financial time series two main opposing viewpoints exist, either capital markets are completely stochastic and therefore prices follow a random walk, or they are deterministic and consequently predictable. For each of these views a great variety of tools exist with which it can be tried to confirm the hypotheses. Unfortunately, these methods are not well suited for dealing with data characterised in part by both paradigms.

This thesis investigates these two approaches in order to model the behaviour of financial time series. In the deterministic framework methods are used to characterise the dimensionality of embedded financial data. The stochastic approach includes here an estimation of the unconditioned and conditional return distributions using parametric, non- and semi-parametric density estimation techniques. Finally, it will be shown how elements from these two approaches could be combined to achieve a more realistic model for financial time series.

Keywords: time series analysis, stochastic and deterministic models, capital markets

Declaration

This thesis describes the work carried out between November 1996 and October 1999 in the Neural Computing Research Group at Aston University under the supervision of Prof. David Lowe.

The work reported in this thesis has been entirely executed by myself. This thesis has been composed by myself and has not, nor any similar dissertation, submitted in any previous application for a degree.

Acknowledgements

First and foremost I would like to thank my supervisor, Prof. David Lowe, for his stimulating approach to guiding me through the experience of science and research. I am especially grateful for his challenging ideas, constructive criticism and helpful support in mastering difficult problems. Besides this I admire also his personal advise and his warm humor, which helped tremendously in stressful moments.

I would like to express a very special thanks for the members of my PhD committee, Dr. Ian T. Nabney and Prof. Mahesan Niranjan, for a fair viva and appreciated final comments on my thesis.

This work has benefited greatly from discussions, encouragement and suggestions from all other members of the Neural Computing Research Group, especially David Saad and Manfred Opper. Furthermore, I would also like to thank Chris Bishop and Chris Williams for the lectures given during my first year at Aston.

I am very grateful to Hanni Sondermann and Vicky Bond for their valuable support and encouragement before, during and after my stay at the Neural Computing Research Group.

Special thanks go to all my fellow PhD students for the interesting discussions and the wonderful time we shared. Especially, I would like to thank Mehdi Azzouzi for his suggestions and conversations, which helped enormously to clarify parts of this thesis.

I also like to acknowledge Jean-François Cardoso, Aapo Hyvärinen and Erkki Oja for making available the source code of the ICA algorithms as well as Zoubin Gharamani for publishing the programs for learning linear dynamical systems.

Last but not least, I would like to thank Quantlab Financial, LLC, Houston, Texas, for their support by providing hardware, software and time to finalise this thesis.

Publications list

- Lesch, R. H. and Lowe, D.: "State Space Models in Finance", Paper presentation at the Sixth International Conference Forecasting Financial Markets (FFM 99), London 26-28 May 1999.
- Lesch, R. H., Caille, Y. and Lowe, D.: "Component Analysis in Financial Time Series", Proceedings of the IEEE/IAFE/INFORMS 1999 Conference on Computational Intelligence for Financial Engineering (CIFEr '99), IAFE, Port Jefferson, NY, p.183–190,
- Lesch, R. H. and Lowe, D.: "Towards a Framework for Combining Stochastic and Deterministic Descriptions of Nonstationary Financial Time Series", Proceedings of the 1998 IEEE Signal Processing Society Workshop: "Neural Networks for Signal Processing", Editors: T. Constantinides, S.-Y. Kung, M. Niranjan, E. Wilson. vol 8, pp 587–596, IEEE, New York,

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Analysing statistical regularities in financial time series for the purpose of, for instance, risk modelling and strategic investment management is a hard and challenging task for several reasons. One point is that there is still no widely accepted theory of financial markets, although major contributions have been made during the last decades. Furthermore, empirical characteristics of financial data present a serious hindrance: they are rather noisy, nonlinear and nonstationary. An analysis needs therefore robust but flexible models which can cope with these attributes. In turn, in the absence of prior knowledge, those models require an enormous amount of data to estimate reliably the relevant statistics. Unfortunately, for financial data on a daily or longer time scale this condition is hardly met. Consequently, a statistical analysis becomes an ill-posed problem.

Due to these difficulties the analysis of financial time series has been focussed so far on models with strongly simplifying assumptions. The linear stochastic framework, for instance, assumes the noise to be the dominant component while the dynamics is restricted to be linear. In contrast, in nonlinear deterministic models it is the nonlinearity which accounts for 'interesting' behaviour in a system while the noise is completely ignored. These two model classes have been applied in the financial domain, however, it is recognised today that they explain empirical phenomena in financial data only insufficiently. To overcome this deficit this thesis outlines a potential framework combining elements from both domains accounting, for instance, for noise in a nonlinear environment.

1.1 Capital markets

Capital markets are one of several domains which have been attracted a lot of attention from the machine learning community, especially in the context of time series analysis. This problem domain strongly exhibits the features of nonstationarity, noise and nonlinearity due to complex interactions of influential factors. This provides a challenge for the academic community as well as for practitioners to apply statistical methods for specific problems such

as portfolio management, currency exchange rate prediction, option pricing, risk analysis and many more.

To provide the background of capital markets as the application domain of this thesis, first the market's structure and function will be summarised. Afterwards the two main analysing paradigms are briefly discussed along with some established theories and models. Finally, studies will be outlined which show contradictory results to the theories, thus providing a motivation for this thesis.

1.1.1 Structure and function

Capital markets are places where venture capital is continuously allocated to potentially profitable investments. According to different types of investments, markets are divided into several segments in which certain assets are traded.

In *stock* markets the traded assets are company shares whose prices depend on the company's performance and its perception by the market participants. In contrast, the return of investment in *bond* markets is usually determined *a priori* over a specified time horizon. In *commodity* markets materials, such as metals, grain, and heating oil, are traded, while *currency* markets provide possibilities to exchange foreign currencies. Finally, in *futures* and *options* markets the conditions of trades can be fixed in advance, while execution is left to a later moment.

For an investment decision in those markets three relevant parameters are usually considered: the expected *return* and *risk* of an investment as well as its time horizon. Additional market constraints such as transaction costs remain ignored in the context of this thesis.

1.1.2 Analytical concepts

For the analysis of capital markets and, more specifically, the evaluation of profitable investments, there exist two complementary viewpoints. In the *fundamental* approach analysts try to estimate the book value[1] of an asset by determining the influence of business-relevant factors. A trading recommendation is then given according to the discrepancy between the real price and the book value.

Fundamental analysis is rule-oriented since explicit causal relationships are modelled using *a priori* knowledge. It is therefore more subjective but also more flexible for unseen situations in contrast to *technical* analysis. The latter's objective is to find statistical patterns in asset prices which requires a large amount of data for reliable results. For this purpose a huge number of methods have been developed in the areas of statistical pattern recognition, time series analysis and machine learning. All these approaches share the assumption that past

[1] The book value can be defined as the sum of all cash flows that owners of the share expect to receive in the future.

patterns will be repeated in the future creating opportunities to exploit them. Consequently, in the case of poor data or truly new situations a purely technical approach will fail. However, due to its less subjective nature and the now available computer power and amount of financial data, technical or *quantitative* analysis is becoming more and more important in the investment decision process.

1.1.3 Theories and models

One of the earliest technically oriented studies regarding the behaviour of asset prices was published by Bachelier in 1900, in which statistical methods, originated for analysing games of chance, were applied to describe stock price returns (Fama, 1965). The so-called Bachelier-Osborne model assumes that price changes from one transaction to the next are independent and identically distributed (i.i.d.), transactions are spread fairly uniformly over time and the distribution of price changes has a finite variance. In the limit of a large number of transactions the accumulated price change represents the sum of i.i.d. random variables. Therefore the central limit theorem suggests for accumulated returns a normal distribution with a variance proportional to the time scale of the summation.

Based on the Bachelier-Osborne model one of the most widely accepted theories about capital markets, the Efficient Markets Hypothesis (EMH), was developed in the 1960s (Cootner, 1964; Fama, 1970). Within the EMH the concept of the *rational investor* assumes that all market participants act rationally, are risk averse and have homogeneous expectations towards the risk and return for the assets they are interested in. A further important point is concerned with *information efficiency*. This refers to the notion that all publicly available information is processed and reflected immediately in the prices. Changes in the prices are therefore triggered only due to new information. Thus, by random occurrence of new information, the price changes themselves should be random, drawn from an i.i.d. process. This is usually assumed to be Gaussian according to the same argument as for the Bachelier-Osborne model. Nevertheless, other limit distributions can be derived for modified summation schemes of the price fluctuations.

1.1.4 Empirical studies

Since the EMH was established, researchers have been trying to determine to what extent capital markets are really efficient. A lot of studies indeed support the theories. However, also a significant number of surveys found anomalies not explainable by the concepts of the Efficient Market Hypothesis.

Regarding daily financial returns, for instance, it is currently widely recognised that their distributions have fatter tails and higher peaks around the mean than a Gaussian distribution (Fama, 1965; Sharp, 1970). As a consequence, models based on normality, such as the Modern

Portfolio Theory (Markowitz, 1952), might give unreliable results. For example, in reality an investor faces a higher risk in stock markets than perceived under the normality assumption.

Concerning the independence of successive returns, predictive structure has been detected on different time scales of the price series; the so-called calendar effects. For instance, the month-of-the-year effect is the statistical anomaly that stock returns in the US and UK are highest in specific months due to the fiscal year (Gultekin and Gultekin, 1983; Haugen and Lakonishok, 1988).

A similar phenomenon is observable on a daily basis. It has been shown that on average there is a smaller return with a higher variance on Mondays then on any other weekday (Hsieh, 1989; Abraham and Ikenberry, 1994).

On a day-to-day basis also significant autocorrelations in the returns have been found in financial markets (Brock, 1991; Hsieh, 1991). Furthermore, the magnitude of this autocorrelation seems to depend on the volatility at that time (LeBaron, 1992). Additionally, strong correlation was also found between the volatility and the volume in a stock market (LeBaron, 1993).

Although just a small fraction of interesting studies concerning market efficiencies, they demonstrate the existence of statistical structure in financial time series. Nevertheless, these empirical facts can at most weaken the concept of information efficiency since they represent just statistical inefficiency. Real investments according to these predictions are very often not profitable due to the costs to acquire information and perform the transaction.

1.2 Time series analysis

Time series analysis is one powerful tool for revealing statistical patterns in time series. It provides the methodological framework for analysing financial data in this thesis. This chapter introduces the background of this analytical framework by considering first the goals of time series analysis and one important issue arising from that: complexity. Empirical features of financial time series are considered which increase the complexity and make the analytical process therefore more difficult. Finally, specific time series models will be introduced which are relevant for the thesis.

1.2.1 Analysis goals

Four objectives can be distinguished within time series analysis (Gershenfeld and Weigend, 1994; Chatfield, 1996): forecasting, modelling, characterisation and control. *Forecasting* means to predict the future continuation of the time series using past and present information. An example is to predict tomorrow's share price based on today's price and other market data.

In contrast, *modelling* tries to capture the underlying dynamics of the system which has generated the time series. Thus an appropriate model takes care of the long-term behaviour and it becomes possible to simulate data from the model. Economists are interested, for instance, in modelling the relationships between macro-economic variables in order to understand the conditions for stable economic growth.

In *characterisation* or *feature extraction* the goal is to find descriptive properties of the time series which could be helpful for modelling and forecasting. Such invariants can be the complexity, the related number of degrees of freedom, stability, signal-to-noise ratio, dominant frequencies, the occurrence of turning points and the forecastability of the time series.

Finally, *control* refers to activities in which the knowledge gained in modelling and characterisation is actively used in order to, for instance, optimise portfolios or minimise risk by hedging strategies.

1.2.2 Complexity

One of the most important issues in time series analysis is the *complexity* of the system producing the time series and of the model used to describe this system. There exists no unique definition for complexity; however, intuitively *model complexity* refers to the effective number of model parameters and their interaction in order to represent the data and their inherent relationships (Bishop, 1995). In contrast, *data* or *system complexity* is associated with the number of factors and their interactions necessary to produce the data (Lowe and Hazarika, 1997; Gershenfeld and Weigend, 1994).

A mismatch in data and model complexity has serious consequences for the result of the modelling process. With too low a complexity, some relevant structure in the data remains ignored. In contrast, an over-complex model will besides incorporating the interesting aspects also fit the noise in the data. Thus, the training patterns are simply memorised instead of generalised. As a consequence a low training error is accompanied with a higher error on unseen, so-called *validation* data. The aim is therefore to match the data with the model complexity in order to represent the statistically relevant structure in the training data.

1.2.3 Empirical time series features

High data complexity typically arises from the presence of one or more of the main features of real-world time series: noise, nonlinearity and nonstationarity. *Noise* is present in all real-world time series and caused, for instance, by, *e.g.*, measurement errors, inherent uncertainty in the system or the effect of a large number of uncorrelated and unidentifiable factors. Noise can therefore be seen as a random fluctuation around the 'true' signal and, consequently, could be modelled probabilistically.

Nonlinearity is the generalisation of a linear dependency to a general smooth functional

form not excluding linear behaviour *per se*. A system is usually characterised as nonlinear when it reacts disproportionally to a variation in its input. A simple example of a nonlinear function is the *tangent hyperbolicus* which often acts as the activation function in nonlinear neural network models.

Nonstationarity is one of the most problematic issues in time series analysis since there exists no universal notion of nonstationarity and the definitions are rather subjective and depend on the viewpoint one might wish to take. Nevertheless, for the context of capital markets several empirical issues can be related to different concepts of nonstationarity.

The most rigorously defined is the concept of *statistical* nonstationarity which requires that the probability distribution describing the system is time-independent. However, in that sense seasonal and daily regimes for the electricity load demand are nonstationary, although these patterns occur regularly. Therefore it seems to be reasonable to classify the presence of quasi-periodic stationary regimes as *multi-stationarity* (Weigend *et al.*, 1995).

In contrast, *evolutionary* nonstationarity refers to the case when the underlying dynamics of a system is smoothly changing over time a-periodically. This behaviour can only be analysed when shorter segments of the time series are *quasi-stationary* such that the change can be tracked. This allows, under conditions for the smoothness of the dynamics and the involved noise, an adaptive modelling of the hidden mechanism which rules the system. An example for the evolution of a system is the capital market itself. Apart from the gradual changes in recording and publishing market and company information, its character has been changing qualitatively due to, *e.g.*, the introduction of derivatives in the 1970s, computer-based trading strategies throughout the 1980s and on-line brokering facilities in the 1990s.

1.2.4 Generative time series models

In order to analyse time series which exhibit the features of noise, nonlinearity and nonstationarity a large number of models and techniques have been developed in the machine learning community. It is therefore useful to classify time series models according to the assumptions they make about the underlying generative process. Two possible categorisations useful for the purpose of this thesis are, for instance, static *vs* dynamic and deterministic *vs* stochastic models.

A *static* model contains no temporal information and therefore explains the current time series value as a function of external inputs and noise. In contrast, for *dynamical* models the current values are conditional on the previous values. *Deterministic* models assume a complete dependence of the current time series value from external inputs or previous values while *stochastic* models take as well noise into account and therefore express dependencies via probability distributions.

The general case for a static time series model can be represented by a map f from a

multivariate input[2] and a multivariate noise term ϵ_t to the scalar observed value x_t:

$$x_t = f(\boldsymbol{u}_t, \epsilon_t). \tag{1.1}$$

This produces interesting behaviour only for the stochastic model class where, for simplicity, the noise contribution ϵ can be assumed to be normally distributed[3]. The deterministic version leads to a simple regression problem without noise and is therefore easily solvable, however, it is not very realistic.

In contrast, dynamical models allow temporal dependencies on a finite sequence $\boldsymbol{x}_{t-1} = (x_{t-1}, \ldots, x_{t-d})$ of d past time series values:

$$x_t = f(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t, \epsilon_t) \tag{1.2}$$

For a linear function f the linear stochastic dynamical model class appears. One representative is the linear autoregression model introduced by Yule (1927) as a technique to forecast future values in a time series using a weighted linear combination of past values:

$$x_t = \boldsymbol{a}' \boldsymbol{x}_{t-1} + c \tag{1.3}$$

with a weighting vector $\boldsymbol{a} \in \mathbb{R}^d$ and a bias $c \in \mathbb{R}$. In this framework noise is necessary to corrupt the linear dynamics of the system in order to produce some 'interesting' behaviour. A linear function f in a dynamical system without noise produces a time series which will either diverge, converge or oscillate periodically without external stimulus. Therefore, referring to linear systems one usually implies a stochastic dependence.

The simplicity of the linear approach made this model class very popular. Nevertheless, the applicability for real-world problems is limited. Several examples have been found for which linear models achieve only suboptimal results, for instance, the case of population growth dynamics (May, 1976).

Progress in this problem was made during the 1980s due primarily the availability of increased computational power together with more sophisticated algorithms accounting, for instance, for a nonlinear regressor function f. For example, Tong (1990) developed with the threshold autoregressive model one of the first nonlinear techniques using two local linearities activated via a threshold.

Another important step was the method of *state-space reconstruction* by using a *time-delay embedding* (Takens, 1981). With that the theory of *nonlinear deterministic dynamical systems* (NDDS), emerged as a useful tool for time series analysis. Within this framework several algorithms were introduced for characterising the intrinsic dimensionality of a system, the degree of nonlinearity and forecastability (Grassberger, 1983; Brock *et al.*, 1987; Wolf *et al.*, 1985).

[2]Systems incorporating external information via an input are called *non-autonomous* compared to *autonomous* systems which use only its own past data and possibly noise.

[3]Any non-normal distribution can be obtained via an arbitrary nonlinear transformation of a Gaussian.

The majority of research in time series analysis has considered either stochastic or nonlinear deterministic models for dynamical problems and (linear and nonlinear) stochastic models for static problems. In this thesis we will investigate nonlinear stochastic dynamical models by combining elements from both approaches.

Two further issues concerning the noise and stationarity need to be discussed. So far only *process* noise was considered which enters the system and therefore alters the continuation of the time series. This is different from the blurring by the measurement process. It is therefore necessary to distinguish *observational* noise η_t as a perturbation of the true signal which does not effect the dynamics of the generating process. For simplicity, the noise is here assumed to be additive. Thus, an observation y_t can be considered a noisy version of the underlying system variable or *hidden state* $\boldsymbol{x}_t$ which is obtained via the observation function g:

$$y_t = g(\boldsymbol{x}_t) + \eta_t. \tag{1.4}$$

All the discussed models concerning nonlinearity, noise and dynamics assume time-invariant functional relationships. This can lead to serious drawbacks in cases where these relationships change over time. Modelling this evolutionary nonstationary can be attempted with the time-window approach or by parameterising the model with the time t:

$$x_t = f(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t, t, \epsilon_t). \tag{1.5}$$

However, in that case fewer data points are available to estimate the parameters which could affect the reliability of the estimates and the statistics of the underlying dynamics. This means such a model can only detect successfully changes which are smooth enough to produce an arbitrary number of samples from which the structural properties can be estimated.

1.3 Thesis structure

Financial time series provide a challenging environment for statistical pattern analysis due to the inherent noise, nonlinearity and nonstationarity. Proposed models for dealing with these problems consider often just one of these features in order to ease the modelling problem. Unfortunately, this might leave out important aspects of the data leading to suboptimal results. The aim is therefore to establish a framework which combines relevant aspects and allows therefore an adaptive modelling of a probability distribution driven by a nonlinear dynamics.

This thesis introduces a framework to analyse financial time series with statistical methods developed in the machine learning community. Within this framework common assumptions about the nature of financial returns are tested whose results lead to proposing a model for time-varying distributions of financial price changes.

The specific problem of extracting statistical regularities in financial time series will be examined from both, the determinstic and the stochastic viewpoint. We will report briefly some of these methods, show their results for a selected but representative set of financial time series and discuss the limitations of both approaches. A framework will be outlined which combines elements from both viewpoints and has great theoretical abilities to model the generator behind the actual time series.

Before considering the two paradigms for time series modelling in more detail, the next Chapter introduces some practical aspects of the modelling methodology in the context of this thesis. This includes issues such as data preprocessing, model selection, parameter estimation and model evaluation.

The first part of the thesis deals with nonlinear deterministic dynamical models ignoring any noise affecting the underlying dynamics. Chapter 3 introduces briefly the background and tests for nonlinearity detection. The aim is to investigate if this methodology is applicable for financial time series in the context of nonstationarity and noise. Specifically, we estimate the correlation integral and based on that explore possibilities to determine fractal dimensions of financial return time series in the context of noise and nonstationarity.

In the second part noise is taken explicitly into account by modelling the fluctuations in financial time series by a distribution. For such a distribution we investigate their properties concerning independence, identity and normality.

The natural starting point for a stochastic analysis is therefore the estimation of the unconditional probability density of financial returns. Chapter 4 discusses density estimation using parametric distribution such as the stable Paretian, Cauchy, Laplace and Weibull distribution and introducing the Bootstrap maximum likelihood approach to estimate the parameters for some of the distributions. Besides investigating semi-parametric methods such as Gaussian mixture models, we also propose and explore the use of mixture models with combined Gaussian and Laplace basis functions.

After analysing the shape of the unconditional distribution, the hypothesis of independence in financial returns is tested by estimating their conditional distribution given previous values. For this purpose we propose a mixture model approach in order to estimate the static two-dimensional return distribution. Furthermore, we enhance a non-parametric dynamic independence test with a Bootstrap component.

All the stochastic methods mentioned operate directly in the observation space and try to estimate a distribution either unconditionally or conditional upon previous values. Assuming an underlying cause for the observations which in turn represent perturbed versions of the true underlying value, Chapter 5 investigates if such static factor models, *e.g.*, principal and independent component analysis, can perform dimensionality reduction and feature extraction. Specifically, we apply single-channel versions of these algorithms to financial data using delay coordinate vector embedding and discuss acquired problems and results.

However, these hidden factor models are static, ignoring time-dependencies and hence cannot explain explicitly phenomena such as persistence in the second-order moment (volatility clustering). Therefore the third part of the thesis considers a model for distributions with time-varying parameters.

There a nonlinear state space model will be proposed which encompasses particle filtering and smoothing of the involved distributions. Since an *a priori* model is in general not available, a learning scheme to estimate the model parameters from the data is presented. This model allows nonlinearities for the hidden dynamics as well for the observation process and non-Gaussian distributions and therefore should be very flexible for modelling a large range of real-world time series. A comparison with a linear state space model in terms of results and problems will be provided using artificial data.

The efficacy and validity of the proposed scheme is considered in the final part of the thesis, where the evidence is accumulated and the relative success and failure of the framework is discussed together with suggestions for extensions for future work.

Chapter 2

Practical modelling aspects

An empirical analysis of a time series usually consists of data preprocessing, model selection, parameter estimation and model evaluation. These issues will be discussed here in detail. First, some initial exploration and important preprocessing techniques will be introduced which help to choose an appropriate model. Afterwards strategies will be reviewed for estimating the parameters of the selected model. Finally, techniques will be outlined for determining the quality of the model with respect to the data and other models.

2.1 Data preprocessing

Data preprocessing includes all transformations performed on the raw time series data with the purpose of bringing them in a proper form for a further analysis. In this thesis a *time series* $\mathcal{X}$ denotes a finite time-discrete realisation of a variable $X(t) \in \mathbb{R}^d$ which varies with time $t \in \mathbb{N}_0$ and is observed at equally spaced time points:

$$\mathcal{X} = \{\boldsymbol{x}_t\}_{t=1}^{T} . \tag{2.1}$$

In the financial context such a time series can represent, *e.g.*, prices p_t of an asset sampled every day at the market's closing. Since prices follow often a geometric growth process and usually exhibit a variance which depends on the price's level a logarithmic transformation of the prices is frequently applied to achieve a stabilised and less dependent variance (Chatfield, 1996).

For some problems it is more interesting to predict the change rather than the level of the quantity. Using this differencing approach the time series becomes more normally distributed and statistically stationary than the log prices. This thesis therefore uses daily log prices and returns r_t derived as their first-order differences:

$$r_t = \log p_t - \log p_{t-1}. \tag{2.2}$$

Such returns represent simply the yield from holding an asset over the one-step time period (Fama, 1965). In general it is recommended to perform a transformation of the data to make

them normal, using, for instance, the Box-Cox transform which includes the log as a special case:

$$y_t = \begin{cases} \frac{x_t^\lambda - 1}{\lambda}, & \lambda > 0, \\ \log x_t, & \lambda = 0. \end{cases} \tag{2.3}$$

In case the log transformation is inappropriate, the parameter λ can, for instance, be determined as the value which minimises the deviation from a Gaussian in the form of the third-order cumulant.

Beside such basic transformations it is necessary to inspect the data visually as part of an *exploratory* analysis before any further modelling activities. In a *time series plot* one can spot, *e.g.*, trends, seasonal behaviour and other important characteristics such as *outliers*. Further visual techniques are *scatter plots* for investigating the relationship between two variables and *histograms* for summarising the distribution of the data.

Outliers can be defined as values in the time series seemingly not consistent with the remaining data. They usually stem from measurement errors. However, sometimes, despite being technically correct, they are the result of exceptional circumstances which should be modeled in a specific way and not in the context of the usual behaviour of a system.

One example of a rare event is the global stock market crash on the 19^{th} of October 1987. On that day the S&P 500 index declined by about 20%, a value outside 20 standard deviations and more than twice the value of the second biggest negative return (estimated over a time range of 70 years). Such a value dominates heavily empirical estimates of higher-order statistics, such as the skewness or kurtosis. Therefore it should be viewed as an outlier and consequently excluded when one is interested in analysing the typical stock market behaviour.

Beside confirming outliers, a visual inspection can help to segment the full dataset into subsets with obviously different characteristics such as volatility. This approach has been applied to all datasets in order to avoid averaging statistical properties over different regimes. As an example, Figure 2.1 shows the full dataset of the Dow Jones Industrial Average index for about 100 years together with a suggested segmentation separating different levels of volatility.

2.2 Parameter estimation

After preprocessing the data a model has to be chosen, for instance, from those introduced in Chapter 1. This choice has to be made according to the aim of the analysis and the prior knowledge available about the data. Once a parameterised model has been selected the next step is to fit this model to the data by estimating its parameters, a process called *learning*. For the example of estimating the probability density function, which will be discussed in detail in Chapter 4, we briefly summarise the Bayesian inference approach. After that the Bootstrap approach is sketched as one alternative parameter estimation technique.

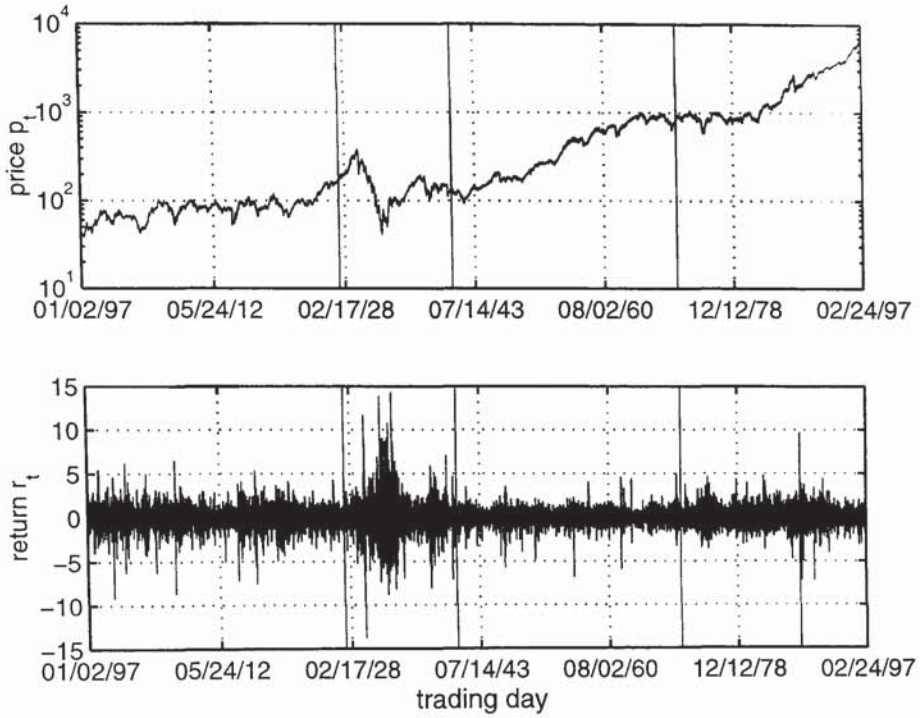

Figure 2.1: The daily closing prices of the Dow Jones Industrial Average (DJIA) for over 100 years: prices on a log scale (top) and returns (bottom). The segmentation into the four datasets of approximately second-order stationarity (with respect to a time scale of several years) indicated by the vertical bars

2.2.1 Bayesian inference

In density estimation a simple model with parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots)$ is given by $p(\boldsymbol{x} \mid \mathcal{X}, \boldsymbol{\theta})$, the probability to obtain a value $\boldsymbol{x}$ given the dataset $\mathcal{X}$ of all observations. The task at hand is now to estimate the $\boldsymbol{\theta}$. Due to the stochastic nature of the model and the finiteness of the data (sampling error) such an estimate is better represented by a distribution $p(\boldsymbol{\theta} \mid \mathcal{X})$ than just a single point. This distribution is called *posterior* since it is computed after the full dataset $\mathcal{X}$ has been seen. In contrast, the *prior* distribution $p(\boldsymbol{\theta})$ does not depend on the dataset since it represents the *a priori* knowledge about the parameters and is therefore available before the data have been seen. This natural uncertainty in the parameter estimate is taken into account in the *partial* Bayesian approach by integrating the model distribution over all possible model parameters:

$$p(\boldsymbol{x} \mid \mathcal{X}) = \int p(\boldsymbol{x} \mid \mathcal{X}, \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid \mathcal{X})\, d\boldsymbol{\theta}. \tag{2.4}$$

The *full* Bayesian approach goes a step further and integrates out the uncertainty of the model structure referring to its architecture and configuration. However, since this approach quickly becomes impractical for higher input dimensions and complex model classes, models are usually selected based on qualitative criteria. Here we adopt therefore at most a partial Bayesian approach.

Nevertheless, for a high-dimensional parameter space even the partial Bayesian approach needs to be simplified since the integral defined above may be difficult to compute. Assuming an arbitrary large number of observations a highly peaked posterior $p(\boldsymbol{\theta} \mid \mathcal{X})$ can be expected around its *most probable* solution $\boldsymbol{\theta}^*$. This allows an approximation by a delta function $\delta(\boldsymbol{\theta} - \boldsymbol{\theta}^*)$ leading to $p(\boldsymbol{x} \mid \mathcal{X}) = p(\boldsymbol{x} \mid \mathcal{X}, \boldsymbol{\theta}^*)$. The remaining question is how to calculate $\boldsymbol{\theta}^*$? One choice is to compute the mode of the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{X})$ by maximising the posterior, therefore referred to as the *maximum a posteriori* (MAP) approach:

$$\boldsymbol{\theta}^{\text{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta} \mid \mathcal{X}). \tag{2.5}$$

This is still problematic if the posterior $p(\boldsymbol{\theta} \mid \mathcal{X}) \propto p(\mathcal{X} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$ depends on a complex prior $p(\boldsymbol{\theta})$. In the case of a flat prior or one which is strongly dominated by the likelihood $p(\mathcal{X} \mid \boldsymbol{\theta})$ the prior can be ignored. Then the most probable parameters are found by simply maximising this likelihood:

$$\boldsymbol{\theta}^{\text{ML}} = \arg\max_{\boldsymbol{\theta}} p(\mathcal{X} \mid \boldsymbol{\theta}) \tag{2.6}$$

Here we adopt this *maximum likelihood* (ML) approach and write the likelihood $\mathcal{L}$ as a function of the model parameters

$$\mathcal{L}(\boldsymbol{\theta}) = p(\mathcal{X} \mid \boldsymbol{\theta}) \tag{2.7}$$

or, instead of maximising $\mathcal{L}(\boldsymbol{\theta})$ directly, we minimize the negative *total log likelihood* function $J(\boldsymbol{\theta}) = -\log \mathcal{L}(\boldsymbol{\theta})$. For this minimisation several gradient-based optimisation algorithms have been proposed. For instance, the Newton-Raphson method uses the inverse of the second-order partial derivatives matrix $\mathcal{H} = \left(\frac{\delta^2 J}{\delta \boldsymbol{\theta}^2}\right)$ of the negative log-likelihood. This is not only computationally very expensive but may also suffer from numerical problems leading to slow or no convergence at all (Gupta and Mehra, 1974).

To avoid this difficulty the Gauss-Newton method, also known as *scoring*, has been suggested, which takes the expectation of $\mathcal{H}$, the Fisher information matrix, with respect to the whole sample space. Besides the high computational costs, this approach experiences problems for singular or near singular estimates for this matrix causing the likelihood to actually decrease or to converge only very slowly.

In order to avoid these practical problems very often the prior and posterior distributions are assumed to be Gaussian which simplifies the inference process since the distributions involved can be expressed analytically rather than approximately. In such a case Bayesian inference is equivalent to the Kalman filter which will be discussed further in the context of state space models in Chapter 6.

2.2.2 The Bootstrap approach

One problematic point with maximum likelihood is the assumption regarding a highly peaked likelihood function, usually idealised to a delta function representing a single point. Instead

of such an estimator a Gaussian approximation could be used, for which its variance gives an idea about the certainty of the estimate of the most likely solution. However, for the general case where the sampling statistics is unknown the Gaussian approximation might not be appropriate. In this case the Bootstrap approach is a powerful alternative tool. By using the Bootstrap approach empirical error bars can be estimated as an approximation for the width of the likelihood function (Zoubir and Boashash, 1998; Hall, 1992).

Since the Bootstrap approach will be used here on several occasions its main concept is briefly introduced in the following. A non-parametric bootstrap is based on a set $\mathcal{X} = \{x_n\}_{n=1}^{N^*}$ of N^* data points for which a statistic θ of interest needs to be estimated. Such an estimate $\hat{\theta} = \hat{\theta}(\mathcal{X})$ can be obtained with the following procedure:

1. Create N *Bootstrap* sample sets of N^* samples each by uniform sampling from $\mathcal{X}$ with replacement.

2. Compute the statistic of interest $\theta^{(i)}$ for each Bootstrap sample $i = 1, \ldots, N$.

3. For $N \geq 100$ the $\theta^{(i)}$ represent approximately the distribution of θ. Assuming a normal distribution for θ the estimator $\hat{\theta}$ is given as the mean $\hat{\theta}$ of all $\theta^{(i)}$ with empirical standard error $\hat{\sigma}_\theta$:

$$\hat{\theta} = \frac{1}{N}\sum_{i=1}^{N}\theta^{(i)} \qquad \hat{\sigma}_\theta = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\theta^{(i)} - \hat{\theta})^2}. \tag{2.8}$$

   For a non-normal case the percentile approach can be used: all $\theta^{(i)}$ are ordered ascendingly; the lower bound for the $(1-\alpha) \times 100$ confidence interval is the value at position $(\alpha/2) \times N$ in this ranking, the corresponding upper value is given at position $(1-\alpha/2) \times N$.

2.3 Model evaluation

After estimating the model parameters it is necessary to test the validity of the established model. One approach is that already mentioned in Section 1.2.2; estimation of the training and generalisation performance with several quantitative measures. Some of the most common ones are briefly summarised next. After this another approach will be discussed which evaluates the quality of a model compared to a hypothesis about the nature of the time series under investigation.

2.3.1 Performance measures

Using a quantitative measure allows to compare the quality of the model on training and validation data. This is necessary in order to determine if the model has generalised the

structure or simply memorised the peculiarities in the training data. In general a model can fit the training data very well using an arbitrary number of model parameters and sufficient training time. However, using too many parameters, or equivalently an overcomplex model, will achieve a poor performance on unseen data, since the model will have fitted also the noise in the training data. This behaviour is called overfitting and can be avoided by regularising and restricting the model complexity.

Nevertheless, overfitting can also take place when one model is repeatedly validated on the same set for different parameters and finally the best model is chosen. Then the validation set became actually part of the training process and does not give necessarily good estimates of the generalisation error.

One example of a commonly used performance measure for predictive models is the mean squared error (MSE) assuming that the distribution of the forecasting error is Gaussian. The MSE is the average squared error for the predictions $\hat{\boldsymbol{x}}_t = \mathbb{E}[\boldsymbol{x}_t \mid \boldsymbol{\Phi}_{t-1}]$:

$$E_{\mathrm{MSE}} = \frac{1}{T}\sum_{t=1}^{T}(\boldsymbol{x}_t - \hat{\boldsymbol{x}}_t)^2. \tag{2.9}$$

The related normalised mean squared error (NMSE) represents the normalisation by the naive predictor which takes for each time series value of the test set the mean of the training set:

$$E_{\mathrm{NMSE}} = \frac{\sum_{t=1}^{T}(\boldsymbol{x}_t - \hat{\boldsymbol{x}}_t)^2}{\sum_{t=1}^{T}(\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}_X^{\mathrm{train}})^2}. \tag{2.10}$$

A more appropriate quality measure for probabilistic models is to use the likelihood $\mathcal{L}(\boldsymbol{\theta}) = p(\mathcal{X}_T \mid \boldsymbol{\theta})$ which gives the probability that a particular model has generated the observed data. In fact, it can be shown that the mean square error is part of the likelihood for a Gaussian model. However, it is difficult to compare the log-likelihood for models of different complexity. A related diagnostic measure for non-Gaussian distributions has been developed by Diebold *et al.* (1998):

$$\hat{P}(\boldsymbol{X}_t \leq \boldsymbol{x}_t \mid \boldsymbol{\Phi}_{t-1}) = \int_{-\infty}^{\boldsymbol{x}_t} P(\boldsymbol{x} \mid \boldsymbol{\Phi}_{t-1})\, d\boldsymbol{x} \tag{2.11}$$

which is the probability of the model to obtain a value less than or equal to the observation $\boldsymbol{x}_t$. For a valid predictive distribution, $\hat{P}$ should be uniformly distributed with zero autocorrelation. This is relevant in case the conditional distribution $p(x_{t+1} \mid \boldsymbol{\Phi}_t)$ is not Gaussian-like but bimodal, for instance. Due to its Gaussian assumption the mean squared error is then practically useless.

Besides calculating such error measures on training and validation data, it makes sense to compare the performance of the model with other conventional techniques and naive predictors in order to check whether superior performance has been achieved.

2.3.2 Hypothesis testing

An approach to evaluate descriptive in contrast to forecasting models can be based on accepting or rejecting a hypothesis. A so-called *null hypothesis* is made about the generating process of the data (Theiler *et al.*, 1992). Besides this hypothesis a discriminating statistic θ is needed which quantifies some aspect of the time series by a certain real number. Then, the statistic is computed for the original data and compared with the one expected under the null hypothesis. If the difference is significant, the null hypothesis can be rejected.

For financial prices a very general null hypothesis is the random walk which means that the returns are generated by an i.i.d. random process. A further restriction can be made regarding a specific distribution of this process such as a Gaussian. In this thesis we will test descriptive statistics for the original data against three types of datasets, which are called surrogate data, according to the term used by Theiler.

The first type of surrogate data corresponds to Gaussian white noise with the sample mean and variance of the original time series. Such a dataset contains therefore no temporal information. A second type is the original time series with the sign of each data point is flipped randomly, suggested by Weigend (1999). This destroys directional structure while amplitude correlations remain intact.

To test if a certain diagnostic statistic is due to linear or nonlinear structure in the data a third type of surrogate data can be employed. The nonlinear temporal structure is removed by randomising the phase in a Fourier representation of the data while all linear correlations are still present.

A fourth type of surrogate data can be used when all temporal structure needs to be destroyed while the distribution should be kept intact. Then a surrogate can be created by sampling uniformly with replacement from the original time series.

Chapter 3

Deterministic Modelling

In the first part of the thesis, we adopt the deterministic approach for analysing financial time series. The focus is therefore on the nonlinearity in the dynamics while noise is assumed to be statistically insignificant. In the following we briefly introduce the background of nonlinear dynamical deterministic systems and describe and apply tests to detect nonlinearity and to characterise the underlying deterministic generator. We investigate if this methodology is applicable for financial time series especially in the context of nonstationarity and noise. Specifically, we will estimate correlation integrals, and based on that, fractal dimensions. Working mainly with univariate time series the embedding approach will be applied using the delay coordinate method and the singular spectrum approach.

3.1 Introduction

An explanation why the theory of dynamical systems has attracted so much interest in the recent years in finance is twofold. At first, there is the recognition of the failure of the Random walk model to explain empirical phenomena such as volatility clustering and extreme events, such as crashes. Second, nonlinear dynamical deterministic systems can produce 'random' looking time series despite each value being completely determined by previous one. This qualitatively different behaviour compared to linear systems stems from the nonlinearity and the unproportional reaction to changes in the system's input. Furthermore, with the additional property of sensitivity to initial conditions their long-term behaviour becomes unpredictable.

With the appearance of some practical algorithms to quantify the behaviour of nonlinear deterministic systems, *e.g.*, via dimension and Lyapunov estimates (Grassberger, 1983; Wolf *et al.*, 1985) and sufficient computing power to realise these algorithms the capital markets became a challenging field of study in the aftermath of the stock market crash in 1987. Elements of dynamical systems theory started to emerge in economics, finance and social sciences in order to explain phenomena such as crowd behaviour leading to panics and crashes (Loistl

and Betz, 1994; Vaga, 1990; Peters, 1991). However, a clear definition for chaos actually still does not exist. Specific studies have dealt, for instance, with detecting nonlinearity, one pre-requisite for chaos, and reported slightly positive identification (Hsieh, 1991; Scheinkman and LeBaron, 1989).

The motivation for reviewing the deterministic concept is to provide some tools for data analysis in order to get an idea about the dimensionality of the system which has been creating the data. This can then be used in the modeling process, for instance, for matching the system complexity with the model complexity.

The main character of this approach is the assumption of a low-dimensional hidden deterministic generator which produces the observations. With the embedding approach the aim is to reconstruct a space topologically equivalent to the phase space, the *embedding space*, in order to investigate properties of the system. This will be outlined next in more detail.

3.2 Dynamical systems

In contrast to stochastic processes, the evolution of a deterministic system can theoretically be completely described by a set of differential equations of the form

$$\dot{\boldsymbol{x}}(t) = \frac{d\boldsymbol{x}(t)}{dt} = \mathcal{F}(\boldsymbol{x}) \tag{3.1}$$

where $\boldsymbol{x}$ is an element of the phase space $\boldsymbol{S} \subset \mathbb{R}^D$, which is the space of all states the system can evolve in. With that the current state of the system is fully determined by its previous one. Since observations made from real world systems are usually at discrete times, the differential equation (3.1) is modified to an explicit functional dependency of the current state $\boldsymbol{x}_t$ at time t on the previous state:

$$\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_{t-1}) \tag{3.2}$$

Unfortunately, for real-world systems a noise-free trajectory is impossible due to, for instance measurement and truncation errors. Furthermore for more complex systems the underlying equations are usually unknown, the phase space is not accessible and its dimension is often not known neither. However, usually one or more variables generated by the system over time are observable. This can be understood as a projection from the higher-dimensional phase space to the (usually) one-dimensional time domain, resulting in a scalar time series.

In the context of capital markets such a projection could be, for instance, the currency exchange rate between two countries. Economists believe that the exchange rate reflects numerous factors influencing the system of economic interaction of the two countries, *e.g.*, differences in the general economic situation, interest rates, consumer prices, working productivity, trade balance and political stability. While it is not feasible to know all the relevant

factors nor to measure them precisely, the exchange rate is measurable at a relatively high frequency and accuracy.

In this thesis we are interested in characterisation and forecasting based on the scalar time series. Therefore it is useful to reconstruct the unknown phase space by deriving and combining relevant information from the immediate past for every point in the time series. This process is called *embedding* and represents a mapping of the observed time series into a higher-dimensional *embedding space*. According to Whitney's embedding theorem such a reconstruction results in a topologically equivalent space under appropriate conditions and can therefore be used for an analysis of the system.

Eckmann and Ruelle (1985) suggested as an embedding to take the current value x_t of the time series together with some higher-order differences[1]:

$$\boldsymbol{x}_t = (x_t, \nabla^1 x_t, \nabla^2 x_t, \ldots, \nabla^{d_E-1} x_t) \tag{3.3}$$

where d_E denotes the dimension of the embedding, therefore called *embedding dimension*. Unfortunately, for noisy time series this will amplify the noise to the level of the signal's amplitude (Loistl and Betz, 1994). Due to the high amount of noise in financial data this method seems not to be applicable and is therefore not used here.

A more robust approach is the concept of *delay coordinate vectors*, introduced by Packard *et al.* (1980)[2]. There an *embedding* or *delay vector* $\boldsymbol{x}_t \in \mathbb{R}^{d_E}$ is constructed by putting together d_E past values:

$$\boldsymbol{x}_t = (x_t, x_{t-\tau}, \ldots, x_{t-(d_E-1)\tau}) \tag{3.4}$$

with the *embedding delay* $\tau \in \mathbb{N}, \tau \geq 1$ representing the time difference between the components within the embedding vector.

A third possibility for an embedding is to use principal and independent components which will be discussed in more detail later in the context of static factor models. These techniques give a valid embedding since they represent linear and noise-free transformations of the data. In that way no information is lost and the dynamics are not altered.

For all these embedding approaches there are certain requirements concerning the embedding parameters in order to achieve a proper embedding. For instance, the embedding dimension d_E has to fulfill the relation to the (unknown) fractal dimension D_F of the system introduced by Mané (1981)

$$d_E \geq 2D_F + 1. \tag{3.5}$$

The fractal dimension D_F represents the number of degress of freedom used by the system and is therefore unknown. It was proven by Takens (1981) that for an embedding dimension fulfilling the above condition the dynamics of a stationary system can be reconstructed. This

[1] The k-th order difference is defined for x_t as $\nabla^k x_t = \nabla^{k-1} x_t - \nabla^{k-1} x_{t-1}$ with $\nabla^0 x_t = x_t$

[2] The use of previous time series values for prediction can be traced back to Yule (1927) who used this approach for the famous sunspot forecasting example

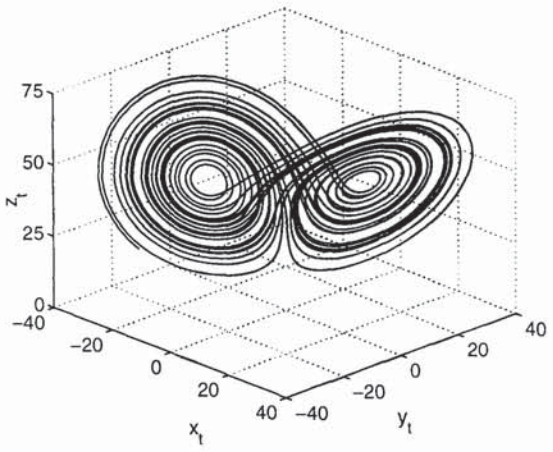

Figure 3.1: Phase space (left) and embedding space (right) with embedding dimension $d_E = 2$ (for visual purpose) and delay $\tau = 3$ reconstructed for the sampled y-coordinate of the Lorenz time series

is done by embedding an infinite long and noise free time series generated by the system into a d_E-dimensional embedding space via Equation (3.4).

For an appropriate value for the embedding delay τ there exist two main constraints. If τ is too small, the vector components are too close together and the resulting embedding vectors are grouped around the identity line in $\mathbb{R}^{d_E}$. With a τ too big, information will be lost from values between x_t and $x_{t-\tau}$ or, even more important, the coordinates belong to different states in the phase space.

As a practical choice it was suggested to take, for instance, the first zero-crossing of the autocorrelation function as the embedding delay (Schuster, 1994). Still, this only ensures linear independence between the components. A similar method which takes also nonlinearity into account is to choose the first local minimum of the *mutual information* of delayed versions of the time series (Fraser and Swinney, 1986).

In order to test for nonlinear relationships in the data the visual approach of *phase plots* can be used in which the coordinates of the state space vectors are plotted against each other. Strong deterministic structure can be detected with phase plots easily since the state space vectors will be restricted to a lower-dimensional set, called *attractor*, in contrast to stochastic systems which will fill out the whole space eventually.

One example of a three-dimensional nonlinear deterministic system is the Lorenz model developed in the context of weather simulation and forecast (Lorenz, 1963). This model is given by set of three differential equations: $\dot{x} = s(y - x)$, $\dot{y} = rx - y - xz$ and $\dot{z} = xy - bz$ with parameters $r = 28$, $b = 8/3$ and $s = 10$. The three-dimensional phase space and the embedding space reconstructed from a scalar time series [3] is shown in Figure 3.1.

[3] A time series of 4000 points was generated via a $4th$-order Runge-Kutta integration using $dt = 0.01$ and a sampling of every tenth y-coordinate.

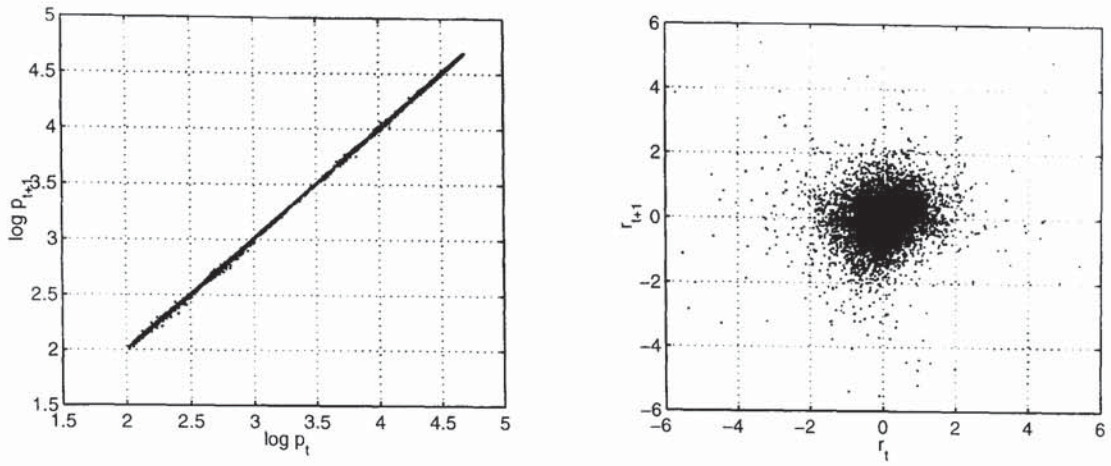

Figure 3.2: Two-dimensional embedding for consecutive S&P 500 prices p_t (left) and returns r_t (right) thus $\tau = 1$

In comparison to the plots for the Lorenz model, Figure 3.2 shows for the S&P 500 index the results of the two-dimensional embedding for the log prices and for the corresponding returns with embedding delay $\tau = 1$. This value has been chosen since financial return time series do usually not possess significant autocorrelation. In contrast to the low-dimensional Lorenz system no 'regular' structure can be recognised here. The embedded returns seem to be a Gaussian-like cloud of points, therefore the embedded log prices are located around the diagonal.

3.3 Dimension estimation

Two main techniques are commonly used to determine the dimensionality of a system for characterising and forecasting purposes. The first method is the delay coordinate method combined with approximations of the fractal dimension of the system via information and correlation dimension. The second uses the singular systems approach to approximate the rank of the covariance matrix as the dimensionality of the system representing the signal separated from the noise. Here we focus on the first approach only.

The concept of fractal dimensions was developed by Hausdorff (1919) in order to quantify nonlinear correlation in dynamical systems which leads to a dynamics which restricts the trajectory of the system to a subspace called a *manifold* of a lower fractional dimension. Mandelbrot introduced for this the term 'fractal dimension' and showed its applicability for characterising several artificial and natural systems (Mandelbrot, 1982).

For the purpose of this thesis we need to restrict the attention to practical estimators of the fractal dimension D_F of a system,c for which several methods have been developed. The intuitive way to determine dimensional measures is to compute the number of points within a hypercube of a given radius or *vice versa* to determine the radius necessary to contain a

fixed number of neighbours and to determine the scaling behaviour of these numbers.

3.3.1 Information dimension

The information dimension is one practical estimator for the fractal dimension of a system. Central in this approach is the distribution of point distances in the embedding space as an estimate for the probability of obtaining a point in a certain region of that space. For this purpose the local density $n_i(r)$ at a d-dimensional point $\boldsymbol{x}_i$ in the embedding space is defined as the expectation of the number of points in a neighborhood around $\boldsymbol{x}_i$ of radius r:

$$n_i(r) = \lim_{N\to\infty} \frac{1}{N-1} \sum_{j=1, j\neq i}^{N} \Theta(r - \|\boldsymbol{x}_i - \boldsymbol{x}_j\|) \tag{3.6}$$

with the Heaviside unit-step function Θ

$$\Theta(s) = \begin{cases} 1 & s > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.7}$$

and the Euclidean norm as the distance measure $\|\cdot\|$. The scaling behaviour of the expectation of the log of this local density can now be investigated in a log-log plot for different radii r. The slope of this scaling behaviour for r approaching zero defines the information dimension:

$$D_1 = -\lim_{r\to 0} \frac{\mathbb{E}[\log n_i(r)]}{\log r}. \tag{3.8}$$

Due to the log used inside the expectation, the information dimension can also be seen as a measure for how much information is necessary to localise a point on the attractor.

3.3.2 Correlation dimension

Grassberger (1983) suggested approximating the fractal dimension by the *correlation* dimension. The so-called correlation integral $C(r)$ is defined as the expectation of the local density $n_i(r)$:

$$C(r) = \mathbb{E}[n_i(r)] = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} n_i(r). \tag{3.9}$$

In contrast to the information dimension, here the expectation is taken before applying the log transform. With that the correlation integral $C(r)$ is equivalent to inverse cumulative histogram of the distribution of distances. It can therefore be calculated with the histogram approach for density estimation which will be discussed in Section 4.2.2. The correlation dimension D_2 is now defined as the scaling behaviour of the correlation integrals in the limit of $r \to 0$:

$$D_2 = \lim_{r\to 0} \frac{\log C(r)}{\log r} = \lim_{r\to 0} \frac{\log \mathbb{E}[n_i(r)]}{\log r}. \tag{3.10}$$

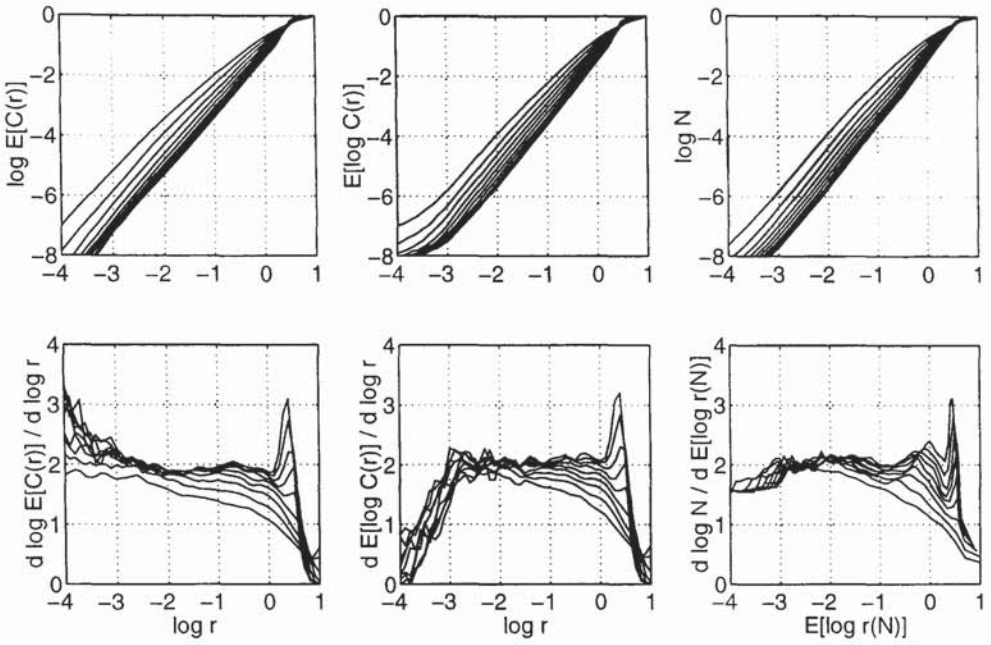

Figure 3.3: Correlation integral estimates for various algorithm variants for the Lorenz time series with $d_E = 2, \ldots, 10$, $\tau = 3$ and $t = 19$. Top row: Log of the mean correlation integral $\log \mathbb{E}[C(r)]$ (left) and expected log correlation integral $\mathbb{E}[\log C(r)]$ against $\log r$ (middle) and log number of neighbours $\log N$ against expected log radius $\mathbb{E}[\log r(N)]$ (right), bottom row: the corresponding local gradient with respect to $\log r$.

3.3.3 Nearest neighbour approach

Unfortunately, in experimental situations with only a limited number of available data both methods suffer from averaging over all $n_i(r)$ for a given radius r. For a small enough radius r_{min} there are not enough data points to represent the correct saling behaviour. The same happens near the attractor border for a radius r_{max}. There the scaling behaviour is distortet. However, inside the region $[r_{min}, r_{max}]$ the scaling law should hold. So if this interval could be determined an improvement in the estimation of the fractal dimension can be made (Holzfuss, 1987). Therefore the method of nearest neighbours (NN) was proposed where the radius is determined which contains a given number of neighbours:

$$D_1 = \lim_{N \to 0} \frac{\log N}{\mathbb{E}[\log r(N)]}. \tag{3.11}$$

This avoids choosing a too small or too big radius and rather estimates these limits automatically, as for instance r_{min} can be determined as the average radius from a data point to the closest next one.

3.3.4 Comparison

Figure 3.3 shows, for the Lorenz example, the correlation integrals and number of neighbours versus the cube radius together with the corresponding slopes. Note that the algorithms used

only 4000 data points and that the estimated local gradient amplifies small datasize effects. Comparing these methods we found, first, that all algorithms estimate the fractal dimension in the region $[-3, 0]$ as being approximately 2.0 for a growing embedding dimension, however with different accuracy.

For instance, the information dimension shows better scaling properties than the correlation dimension and the method of nearest neighbours gives clearer results in terms of a longer linear part and slope closer to the true value. In the figure it can be seen, furthermore, that the slopes for the information dimension and the method of nearest neighbours are less spread and better behaved at the borders of the scaling region than for the correlation dimension. It also seems that the nearest neighbour approach is more constant in the slopes for smaller radii.

Similar results have been achieved as well for other low-dimensional examples of nonlinear deterministic maps. Therefore the algorithm for estimating the information dimension estimated via the nearest neighbour method will be primarily used for the simulations with financial data.

Noise impact on the dimension estimate

To be more realistic and to allow some noise in the time series it is useful to assess the impact of noise for the estimation of the fractal dimension. The experiment was therefore repeated using 1% and 25% additive observational noise as well as 1% process noise for the Lorenz time series. Figure 3.4 shows the results using the NN approach where it can be seen that the noise has the effect of increasing the calculated dimension. While the 1% observational noise lifts the estimates for the dimension within the linear region just slightly above 2, the 1% process noise results in a divergence of the dimension estimate in a similar way as the 25% observational noise.

Surrogate datasets

As introduced in Section 2.3.2 it is necessary to apply such nonlinear algorithms on surrogate datasets in order to discuss the results properly considering, for example, the cause of a found convergence in the estimate. Therefore two different types of surrogates have been created for the Lorenz time series.

For the first version the phases in a Fourier transform were shuffled. The second series was created by an autoregressive model of order 4 with the AR coefficients estimated from the original data. Figure 3.5 shows a time series segment of the original and the surrogate data as well as the slopes for the different test cases against the embedding dimension. There a saturation in the slope of the correlation integral can be observed for the original time series beginning with embedding dimension $d_E = 5$ (note that this confirms the embedding

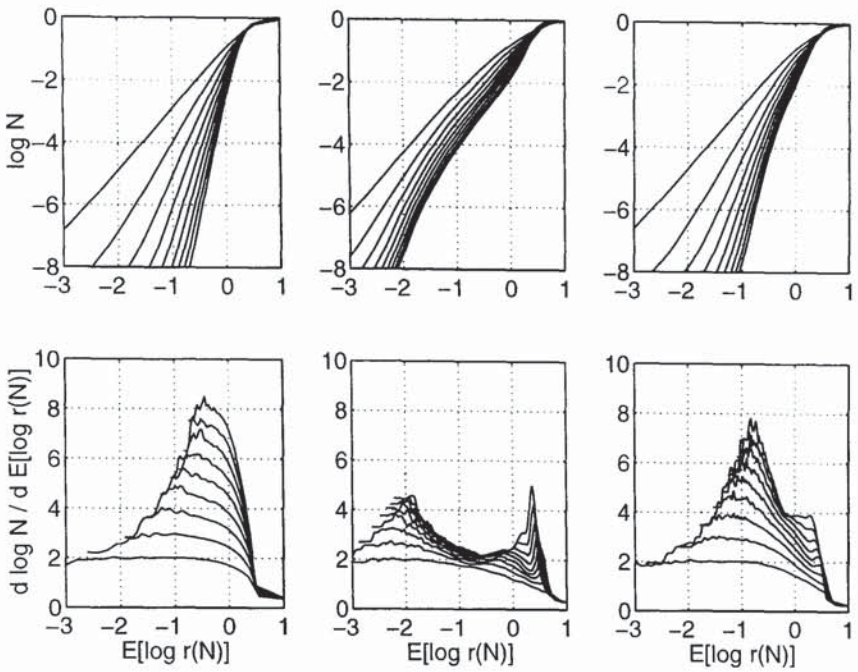

Figure 3.4: Estimates for the correlation integral for a noisy Lorenz time series: 1% process (left), 1 % (middle) and 25 % (right) observational noise

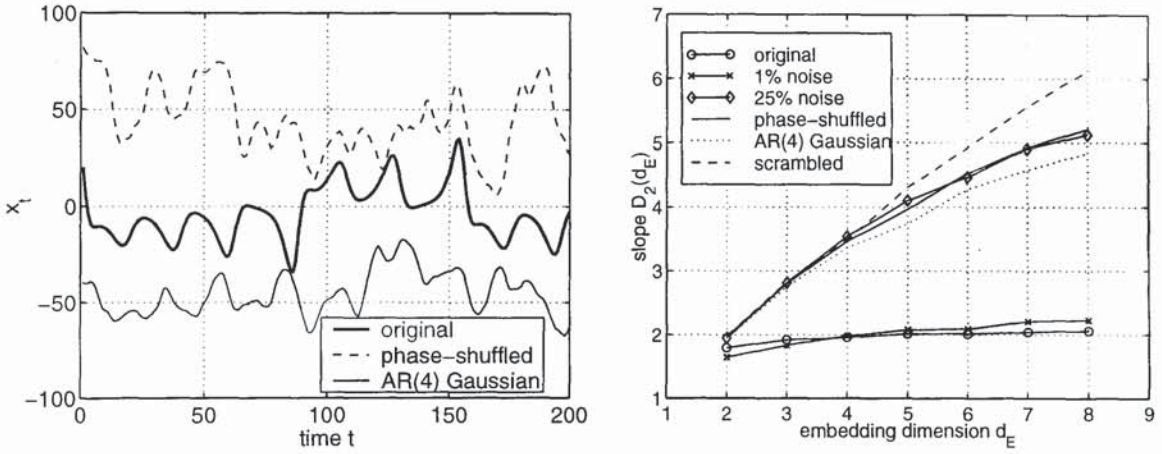

Figure 3.5: Segment of 200 points of the original Lorenz time series together with two surrogate datasets (left) and the D_1 estimates for the fractal dimension (by the NN approach) for growing embedding dimension for the original time series and the two surrogates (right). One surrogate is a phase-shuffled version, the other is an AR(4) filtered Gaussian noise series with the same linear correlation structure as the original Lorenz time series

requirement $d_E \geq 2D + 1$) resulting in a value of 2.03 ± 0.02 averaged over $d_E = 5, \ldots, 8$, quite close to the analytical value of $D_2 = 2.05$ (Grassberger, 1983). In contrast, the dimension estimate for the surrogate datasets seems to grow almost linearly with the embedding dimension as expected. Concerning the noise it can be noticed that for the low-level of 1% only a slight increase in the estimated dimension can be noticed. However, for 25% noise the result is not different from one obtained for a surrogate dataset. This shows that noise with an arbitrary amplitude will finally mask the deterministic content in the data completely.

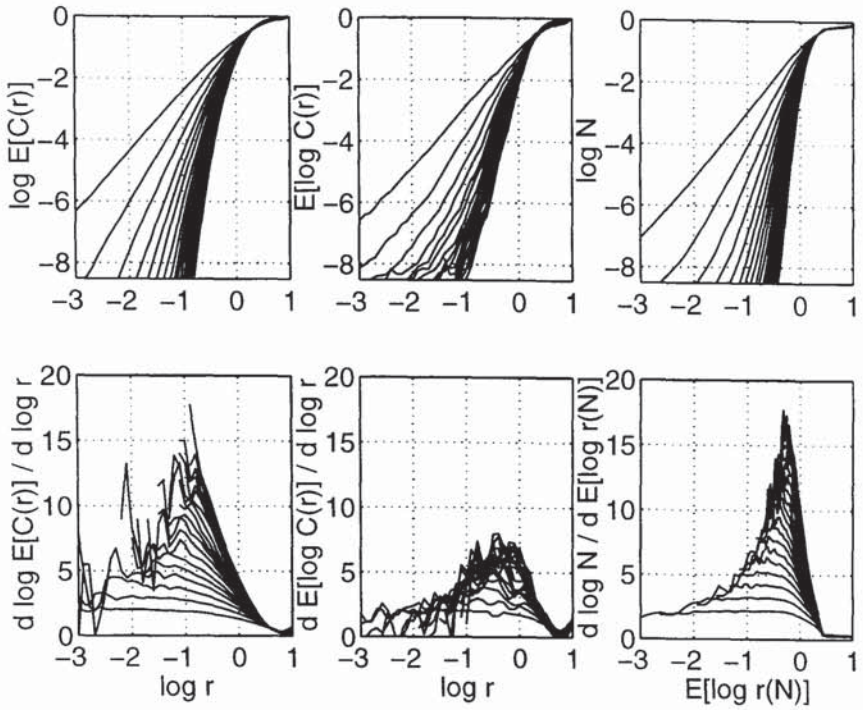

Figure 3.6: Correlation integral estimates for the three algorithms for IBM returns with $d_E = 2, \ldots, 20$, $\tau = 1$ and $t = 1$. Top: mean correlation integral $\log \mathbb{E}[C(r)]$ (left) and expected correlation integral $\mathbb{E}[\log C(r)]$ against $\log r$ (middle) and number of neighbours N against $\mathbb{E}[\log r(N)]$ (right). Bottom: corresponding local gradients with respect to $\log r$

Financial dataset

In order to test the ability of these algorithms with financial data the IBM returns time series was choosen (set 1 with 4981 data points) and the correlation integrals $C(r)$ and the cube sizes $r(N)$ were computed for an embedding dimension $d_E = 2, \ldots, 20$ and a range of radii r and numbers of neighbours N.

Figure 3.6 contains the results for the original data in terms of the correlation integrals and their slopes. There it can be noticed that the information dimension for both algorithm variants does not saturate for an increasing embedding dimension. However, the slope for the correlation dimension seems to reach a plateau at around 6 although the estimates look more 'erratic' than those for the information dimension. The most reliable algorithm seems to be the nearest neighbour method since there the slopes are smoother compared to the other two approaches.

In order to assess the significance of these results the three algorithms were also applied to surrogate data created by shuffling the IBM returns. Figure 3.7 shows slopes for the information dimension similar to those for the original data. In contrast, the correlation dimension seems to grow stronger with the embedding dimension for the surrogate than for the original data. However, since the underlying correlation integrals exhibit a less strongly linear scaling behaviour in the log-log plot these results have to be treated carefully.

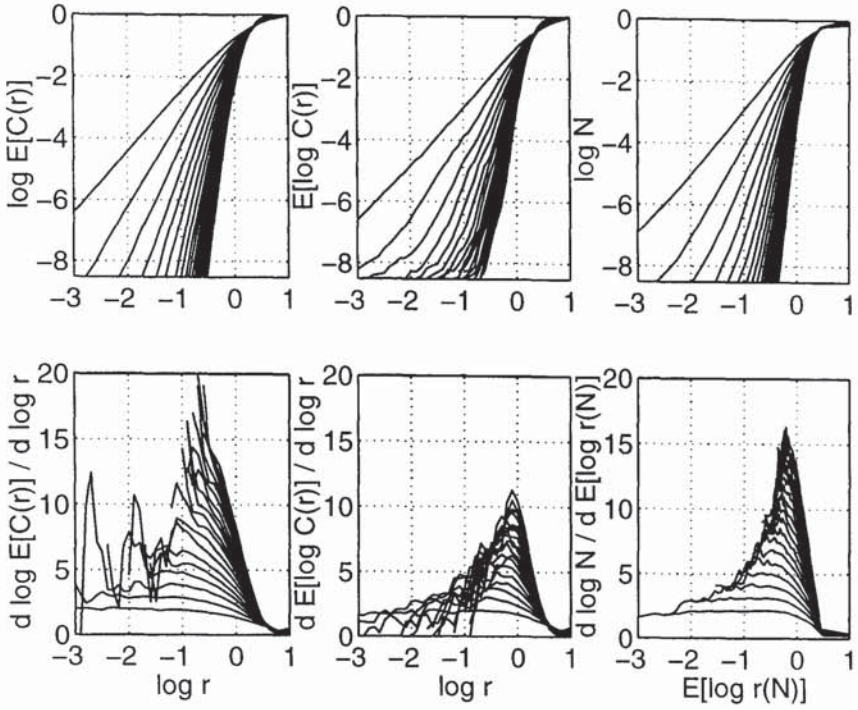

Figure 3.7: Correlation integral estimates for the three algorithms for surrogate IBM returns with $d_E = 2, \ldots, 20$, $\tau = 1$ and $t = 1$. Top: mean correlation integral $\log \mathbb{E}[C(r)]$ (left) and expected correlation integral $\mathbb{E}[\log C(r)]$ against $\log r$ (middle) and number of neighbours N against $\mathbb{E}[\log r(N)]$ (right). Bottom: corresponding local gradients with respect to $\log r$

Computing the slope automatically by looking for the linear scaling region within the log-log plot then it can be seen in Figure 3.8 that the best linear scaling region has been achieved by the nearest neighbour method for computing the information dimension. This can be concluded from the small error bar (the standard deviation of a least-squares fit through all data points in the linear scaling region) and the relatively smooth increase of the slope compared to the more 'erratic' behaviour of the other methods.

In a summary, the algorithms tested for dimensional estimates are not able to distinguish between the financial data in their original and surrogate form. The correlation dimension estimates seem to vary a lot, is less well behaved than the estimates for the other approaches. The information dimension estimate seems to grow slower for the original data than for the surrgate ones, although a saturation cannot be confirmed. All methods suffer heavily from the small size of the linear region which violates first the intention to determine the 'usual' scaling behaviour which actually requires a scaling behaviour which exists over the majority of the range. The second point is with such small linear ranges the calculated dimension estimates are prone to error due to the small number of points.

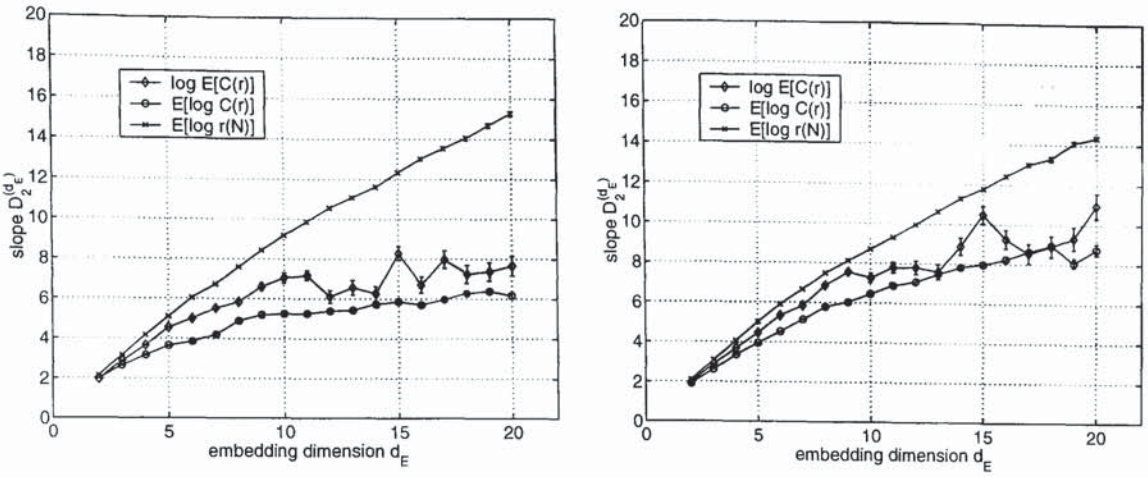

Figure 3.8: Slopes for the three algorithms for the fractal dimension estimates of original (left) and surrogate (right) IBM returns (The algorithm for the correlation dimension calculates $\log \mathbb{E}[C(r)]$, the one for the information dimension determines $\mathbb{E}[\log C(r)]$ and the method of the nearest neighbours computes $\mathbb{E}[\log r(N)]$).

Pointwise correlation dimension for nonstationary time series

Beside the noise, another important empirical issue needs to be tested as well; nonstationarity. Skinner *et al.* (1993) suggested the method of a pointwise estimate for the correlation dimension (PD2). There the scaling behaviour is determined for a number of reference points in order to keep it computationally feasible. These reference points are fairly evenly spread over time. In the stationary case the mean of the dimension estimates over all reference points is called the averaged pointwise dimension and approximates the information dimension (Holzfuss, 1987).

We extend this approach to using reference vectors within a moving time window. Figure 3.9 shows the results using this algorithm on nonstationary data. The time series consists of three consecutive segments of 4000 points each, starting with the phase-shuffled surrogate, followed by the original Lorenz time series and the AR(4) Gaussian surrogate. The pointwise dimension was estimated for embedding parameters $d_E = 2, \ldots, 8$, $\tau = 3$, $t = 10$, $N_{ref} = 10\%$ and a moving time window of 1000 points. Removing the five 'outliers' greater than four in the middle segment an average dimension is calculated as $p\hat{D}_2^{(8)} = 2.09 \pm 0.22$.

This algorithm has been applied to the financial return time series using a maximum embedding dimension of $d_E = 20$, an embedding delay $\tau = 1$ and a set of 40% reference points. As an example, Figure 3.10 shows the results for the IBM dataset. It can be seen that the correlation integral approach ($N(r)$) estimates consistently higher slopes respectively dimensions than the nearest neighbour approach ($r(N)$). An interesting finding is that the latter produces a similar mean but with a smaller variance for the scrambled data. In contrast, the $N(r)$ approach achieves a slightly higher average on the scrambled data with a similar mean.

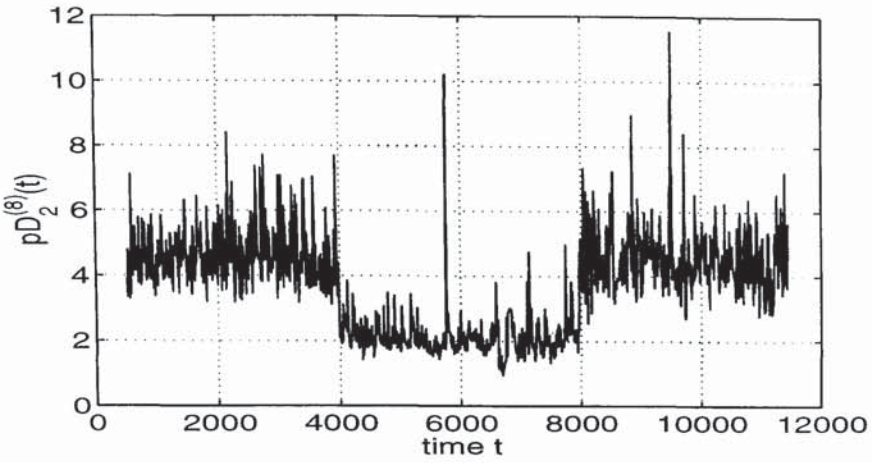

Figure 3.9: Estimates for the pointwise correlation dimension for a nonstationary time series consisting of three subsets each of 4000 points: the Lorenz time series as the middle segment, the phase-shuffled surrogate on the left and the linear surrogate of this series on the right side

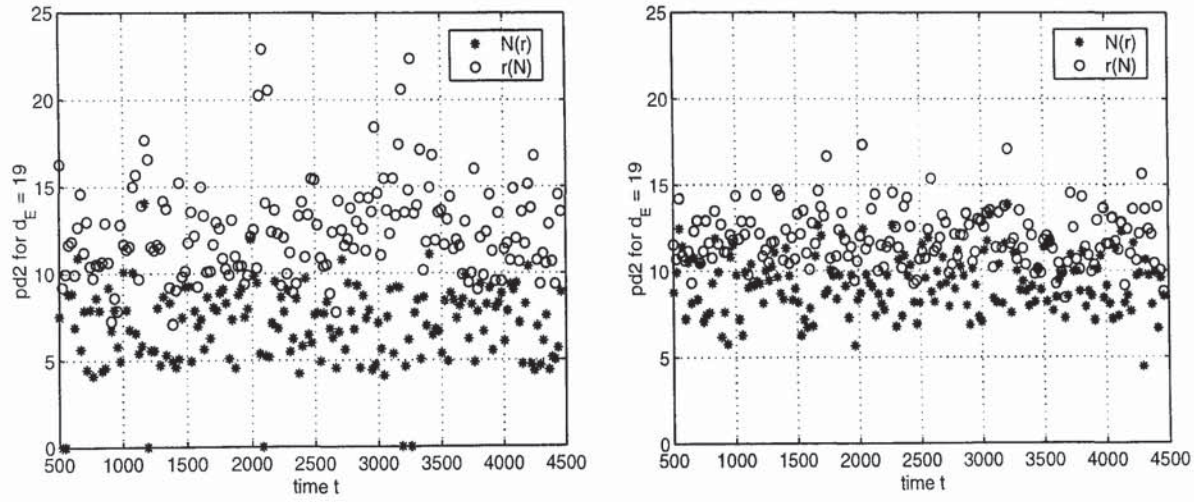

Figure 3.10: Estimates for the pointwise correlation dimension for set 1 of IBM returns using the original returns (left) and the scrambled version (right) using both the correlation integral approach ($N(r)$) and the nearest neighbour approach ($r(N)$)

This behaviour has been confirmed with the other financial data tested. This difference in the results on the original and scrambled data is not large enough to confirm the presence of nonlinear structure in the data. Furthermore, the variation in the statistics seems to be random and not temporally emphasised. For this behaviour two explanations are possible. Either there is no nonlinearity in the data at all or the nonlinear relationships are changing so fast that they cannot be traced using daily data only.

3.4 Discussion

This chapter summarised the concept of deterministic dynamical systems applied to time series analysis. Under the premise of the availability of noise-free data this approach is theo-

retically able to reveal a nonlinear systems dynamics. One important tool in this framework is the estimation of the numbers of degrees of freedom respectively the dimension of the data generator by the approach of the correlation and information dimensions.

The motivation to use this concept for the analysis of financial data stems from the observation that random looking systems can actually have very simple deterministic equations. The question was therefore if there is any nonlinear dynamics in financial prices which could cause observed peculiar behaviour, particularly in situations such as stock market crashes, for instance.

Using the techniques of dimension estimation we have demonstrated that there is no indication for a low-dimensional attractor for daily financial prices or returns. In the performed experiments either the dimension estimates did not converge or similar estimates were also obtained for randomised data.

Apart from the possibility of a simple lack of the assumed nonlinear dynamics one explanation for these findings might be the presence of noise masking the deterministic relations. Such an effect has been confirmed by tests on synthetic data. There it has been shown that observational noise and to a much stronger degree process noise raise the dimension estimate and finally lead to non-convergence for the estimate.

On a daily or longer time scale it seems therefore unlikely to identify chaos. Since intraday data were not tested here it could be speculated that nonlinear relationships could be present on a shorter time.

However, here another direction is taken using the same daily time scale. Since noise seems to be an important factor for the results found, it would be useful to take the noise explicitly into account and allow therefore a probabilistic dependency of the current time series value from the past. This paradigm will be used throughout the remainder of this thesis. First, we concentrate on the modelling of the noise without any assumptions about dependencies. Finally, linear and nonlinear dynamics will be allowed, too.

Chapter 4

Density modelling

In the second part of the thesis we accept the intrinsic stochastic nature of the process which generates financial returns. This means we allow noise to enter the generating process and therefore model it in a probabilistic way, hence the need to characterise the distribution of financial returns.

The aim for this chapter is therefore to confirm or reject common hypotheses about the nature of financial distributions and to find an efficient way of modelling those densities since this will be necessary later for more complex models. Beside marginal distributions we are also interested in joint and conditional distributions. Looking for a fewer number of factors which explain a huge number of simultaneous observations and to separate process from observational noise is the aim of static factor models. These will be explored in the following chapter. Finally, allowing a dynamics for the return generating process we will propose a nonlinear state space model for tracking the time-varying non-Gaussian return distribution.

Since density modelling is an essential tool for analysing stochastic time series and plays an important role in finance *e.g.*, for option pricing and risk analysis, we explore in this chapter techniques for estimating the density of financial returns. Distributions of price changes are modeled, for example, in order to analyse the return and risk of an investment. This translates into a prediction for the mean and the spread of the distribution of returns of investment. Furthermore, such returns are often assumed to be independent and identically distributed and follow, for instance, a Gaussian distribution. These assumptions are made in several models of capital markets as *e.g.*, the Efficient Market Hypothesis (Fama, 1965), Modern Portfolio Theory (Markowitz, 1952) and the Capital Asset Price Theory.

Using a set of representative financial time series we will demonstrate empirical evidence for non-iid behaviour and specifically, the non-Gaussianity of returns. These findings are the motivation for a nonlinear approach to model the evolution of the predictive distribution in a later chapter.

4.1 Introduction

A discrete-time *stochastic* or *random process* X is a collection $\{X_t : t = 0, \pm 1. \pm 2, \ldots\}$ of time-indexed random variables X_t. In the context of stochastic modelling we refer to a finite realisation $\mathcal{X} = \{x_t\}_{t=1}^{T}$ of such a random process as a *time series*. Thus a stochastic time series is only partially determined by past values and should therefore be modeled as a probability density function conditional on past values.

A stochastic process is called *strictly stationary* if the joint distribution of a finite-dimensional subset of all families remains constant. The weak form of stationarity requires only constant first and second order moments.

In this chapter we are interested in characterising a stochastic process by its underlying distribution. Beside a brief introduction to relevant elements of distribution theory, methods are investigated for modelling of unconditional and conditional distributions of financial returns.

4.1.1 Distribution, density and characteristic functions

The distribution of a continuous random variable X can be fully specified either by its cumulative distribution function or the characteristic function. The *cumulative distribution function* (c.d.f.) determines the probability P to obtain a value for X less than a specific value x: $F(x) = P(X \leq x) \in [0, 1]$. With that F is monotone and its derivative with respect to x defines the *probability density function* $p(x)$:

$$p(x) = \frac{dF(x)}{dx}. \tag{4.1}$$

Due to the properties of F this is always greater or equal to zero and integrates to one:

$$0 \leq p(x) \leq 1, \qquad \int_{-\infty}^{\infty} p(x)\, dx = 1 \tag{4.2}$$

The *characteristic function* $\Phi(t)$ of a distribution is the expectation of e^{itX} for $t \in \mathbb{R}$ under $p(x)$, the Fourier transform of the probability density function:

$$\Phi(t) = \int_{-\infty}^{\infty} e^{itx}\, p(x)\, dx. \tag{4.3}$$

Unlike the density the characteristic function is always guaranteed to exist[1]. In the case that the p.d.f. does exist as well, it can be expressed by the corresponding inverse transform

$$p(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-itx}\, \Phi(t)\, dt. \tag{4.4}$$

Since the $p(x)$ has to be real, $\Phi(t)$ and $\Phi(-t)$ are complex conjugate[2] to each other. It follows furthermore, that $\Phi(0) = 1$. This means $\Phi(t)$ needs to be considered only for $t > 0$ which

[1] For the class of stable Paretian distributions one cannot write in general a closed form for the probability density function (Cp. with Section (4.3.1)).

[2] A complex conjugate pair $\Phi(t)$ and $\Phi(-t)$ differs just in the sign of the imaginary part therefore $\Re\{\Phi(t)\} = \Re\{\Phi(-t)\}$ and $\Im\{\Phi(t)\} = -\Im\{\Phi(-t)\}$.

allows a further simplification in order to get real values for the p.d.f. after the numerical transform of the characteristic function:

$$p(x) = \frac{1}{\pi} \int_0^\infty \cos(tx)\Re\{\Phi(t)\} + \sin(tx)\Im\{\Phi(t)\}\, dt. \tag{4.5}$$

The probability density, the cumulative distribution and the characteristic function are unique representations of a distribution. In the following some summaries about distributions are introduced which can often be used in case the full p.d.f. is, for instance, to difficult to derive.

4.1.2 Measures, moments and cumulants

For distributions with their probability mass concentrated in one particular area a useful description can be made in terms of the location of this concentration. One important measure for this is, for instance, the *mean* μ which is the expectation of X under its distribution:

$$\mu = \int_{-\infty}^{\infty} x\, p(x)\, dx \tag{4.6}$$

A more robust feature is the *median* x_m which divides the cumulative probability into two equal parts:

$$F(x_m) = 1 - F(x_m) = \frac{1}{2}. \tag{4.7}$$

A third important quantity is the *mode* $x_{mode} = \arg\max_x p(x)$ as the x with a corresponding maximum in $p(x)$. If beside this global maximum there are no further local maxima the distribution is unimodal. In case that $p(x)$ has several local maxima the distribution is multi-modal with each local mode defined via $\dot{p}(x_\mathrm{mode}) = 0$ and $\ddot{p}(x_\mathrm{mode}) < 0$.

Beside these measures of location unimodal distributions can also be described in terms of dispersion about its location. For instance, the average absolute deviation from the median x_m defines the *mean deviation* ν as

$$\nu = \int_{-\infty}^{\infty} |x - x_m|\, p(x)\, dx \tag{4.8}$$

while the *variance* or squared *standard deviation* σ^2 represents the mean squared distance from the mean μ:

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \mu)^2\, p(x)\, dx. \tag{4.9}$$

Generalising this concept of measures of location and dispersion leads to the moments of the distribution. The statistical *moment* M_n *about zero* of order $n \in \mathbb{N}_0$ is the expectation of X^n under its distribution:

$$M_n = \int_{-\infty}^{\infty} x^n\, p(x)\, dx. \tag{4.10}$$

This determines $M_0 = 1$ and $M_1 = \mu$ which in turn is used to define the *central moments* m_n about the mean of order $n \in \mathbb{N}$:

$$m_n = \int_{-\infty}^{\infty} (x - \mu)^n\, p(x)\, dx. \tag{4.11}$$

With that it follows $m_1 = 0$ and $m_2 = \sigma^2$. Apart from this descriptive purpose, the moments can be used to approximate the characteristic function since they appear as coefficients in a power series expansion of $\Phi(t)$ using the Taylor series expansion of e^{itx}:

$$\Phi(t) = \sum_{n=0}^{\infty} \frac{(it)^n}{n!} \int_{-\infty}^{\infty} x^n \, p(x) \, dx = \sum_{n=0}^{\infty} \frac{(it)^n}{n!} M_n. \tag{4.12}$$

Naturally, the moments can be derived from the characteristic function via

$$M_n = (-i)^n \frac{d^n \Phi(t)}{dt^n}\bigg|_{t=0}. \tag{4.13}$$

Beside the moments another set of descriptive figures exists which can be derived from the log of the characteristic function:

$$\kappa_n = (-i)^n \frac{d^n \log \Phi(t)}{dt^n}\bigg|_{t=0}. \tag{4.14}$$

Thus, the cumulants are therefore the coefficients in a power series expansion of $\log \Phi(t)$:

$$\log \Phi(t) = \sum_{n=1}^{\infty} \frac{(it)^n}{n!} \kappa_n. \tag{4.15}$$

Combining now Equation (4.12) and (4.14) the first four cumulants can be expressed via the moments as

$$\kappa_1 = M_1 = \mu, \quad \kappa_2 = m_2 = \sigma^2, \quad \kappa_3 = m_3, \quad \kappa_4 = m_4 - 3m_2^2, \tag{4.16}$$

which will be used in this thesis. For notational convenience we also write γ and κ to denote the third and fourth cumulant respectively. Furthermore, higher-order cumulants are often normalised by the corresponding power of the standard deviation in order to get a dimensionless quantity:

$$c_n = \kappa_n / \sigma^n, \tag{4.17}$$

a form we will adopt here as well. This is equivalent of normalising the data to zero mean and unit variance before estimating the cumulants.

Using cumulants instead of moments is motivated by the properties of cumulants. All of them are additive compared to just first two moments. However, although cumulants represent characteristic features of a distribution, they as well as the moments do not determine a distribution completely, since two different distributions can have the same set of moments (Stuart and Ord, 1994).

Furthermore, moments and cumulants do not always exist since the integrals in Equation (4.10) do not necessarily converge for every distribution[3]. If they do exist they are a set of constants describing the distribution in a useful way. Beside the mean and the variance two higher-order cumulants are often used for descriptive purposes.

[3]For example, the Cauchy distribution $p(x) = \pi^{-1}(1 + x^2)^{-1}$ has no moments at all.

The third cumulant, the *skewness*, measures the asymmetry of the distribution around its mean. A positive skewness indicates a heavier tail (including more probability mass) towards positive values and *vice versa*. Moreover, the fourth cumulant, the *kurtosis*, represents the relative peakedness or flatness of the distribution compared to the normal one. The bigger the kurtosis the more probability is concentrated around the centre and in the extreme tails of the distribution compared to the normal distribution.

In the financial context these cumulants are of interest since distributions of asset returns are very often characterised by the mean as the expected return and the variance accounting for the risk of an investment. Furthermore, taking skewness and kurtosis into account for instance in risk analysis, will achieve more accurate results for non-normal data than using the variance alone.

In that way cumulants can be used to express non-Gaussianity since for the normal distribution all but the first two cumulants are zero. Thus, the deviation from zero for higher-order cumulants can be taken as a measure of the non-Gaussian character of the data.

For multi-variate data cumulants can also be used to quantify dependencies between the variables since for independent X_{i_j} it follows that $\mathbb{E}[X_{i_1} \dots X_{i_n}] = 0$ apart from $i_1 = \cdots = i_n$. This can be derived by expressing the characteristic function in terms of cumulants for a multi-dimensional *random vector* $\boldsymbol{X} = (X_1, \dots, X_m)$. In that case the characteristic function is defined for $\boldsymbol{t} = (t_1, \dots, t_m) \in \mathbb{R}^m$ as follows

$$\Phi(\boldsymbol{t}) = \int_{-\infty}^{\infty} e^{i(\boldsymbol{x}'\boldsymbol{t})}\, p(\boldsymbol{x})\, d\boldsymbol{x} = \exp\left\{ \sum_{n=1}^{\infty} \frac{i^n}{n!} \sum_{j_1,\dots,j_n=1}^{m} \kappa_{j_1 \dots j_n}\, t_{j_1} \dots t_{j_n} \right\} \tag{4.18}$$

using the kth-order joint cumulant tensor, $\kappa_{i_1,\dots,i_m}$ defined as (Tong, 1990)

$$\kappa_{i_1,\dots,i_m} = \sum_{p=1}^{n} (-1)^{p-1} (p-1)!\, \mathbb{E}\left[\prod_{j \in v_1} X_{i_j} \right] \cdots \mathbb{E}\left[\prod_{j \in v_p} X_{i_j} \right] \tag{4.19}$$

where the summation extends over all partitions $(v_1, v_2, \dots, v_p)$ of $(1, 2, \dots, n)$. Assuming a zero first-order cumulant $\kappa_i = \mathbb{E}[X_i] = 0$ for each X_i, the second, third and fourth cumulant tensors are given as

$$\kappa_{ij} = \mathbb{E}[X_i X_j] \tag{4.20a}$$

$$\kappa_{ijk} = \mathbb{E}[X_i X_j X_k] \tag{4.20b}$$

$$\kappa_{ijkl} = \mathbb{E}[X_i X_j X_k X_l] - \kappa_{ij}\kappa_{kl} - \kappa_{ik}\kappa_{lj} - \kappa_{il}\kappa_{jk} \tag{4.20c}$$

from which the expressions for marginal cumulants in Equation (4.16) can be recovered using equal sub-indices.

One important property of cumulants is that they can be estimated from the data without any prior knowledge about the data. This non-parametric approach has a complexity which

grows with the number of data points and is in contrast to parametric models which assume a specific structure of the data. Then their parameters have to be estimated from the observed data. Due to their implied structure for the data as well as their data-independent complexity parametric methods are less flexible but more robust against overfitting compared to non-parametric techniques. For this reason we will also use beside non-parametric techniques for the estimation of cumulants and density functions some parametric distribution models which have been suggested for financial returns. Furthermore, we will look at mixture models as a semi-parametric approach for density estimation as a way to combine the advantages of each estimation paradigm.

4.2 Non-parametric estimation techniques

Here, cumulants will be used in several ways. Next we will propose a reliable technique to estimate sample cumulants and investigate their characterisation abilities for the set of selected financial time series. Afterwards we will explore the use of cumulants to approximate the characteristic function. Later we will summarise a cumulant-based algorithm to test the hypothesis of independence in successive financial returns and finally apply with independent component analysis an algorithm which diagonalises the cumulant tensor of fourth order to achieve statistically independent sources.

4.2.1 Sample cumulants

For practical purposes *sample cumulants* are estimated in the usual way by approximating the expected values as time averages over the whole dataset. Thereby, care has to be taken in determining the cumulants directly from the data since they are sensitive to extreme values; the higher the cumulant's order the more sensitive it is due to the power involved. This limits their use in the case of relatively short time series.

To avoid this problem we propose using the Bootstrap approach, introduced in Section 2.2, for a reliable estimate of the cumulants. For each of 1000 Bootstrap runs with the sample size of the original dataset the first four cumulants are calculated for the financial price returns. Assuming normal distributed cumulant estimates its mean approximates the true cumulant. Additionally, the standard deviation of the sample cumulants provides an error bar on the mean estimate.

Figure 4.1 shows the Bootstrap estimates for the first and second sample cumulant for each dataset. These results are also listed in Appendix D in Table D.1. It can be observed that the *mean* is close to zero for all investigated time series. Furthermore, for stock prices and indices there is a tendency for a positive mean. This can be interpreted as the average economic growth or price inflation of a financial asset. The only one exception is set 2 for the DJIA which covers the time of the Great Depression from 1929 until 1933.

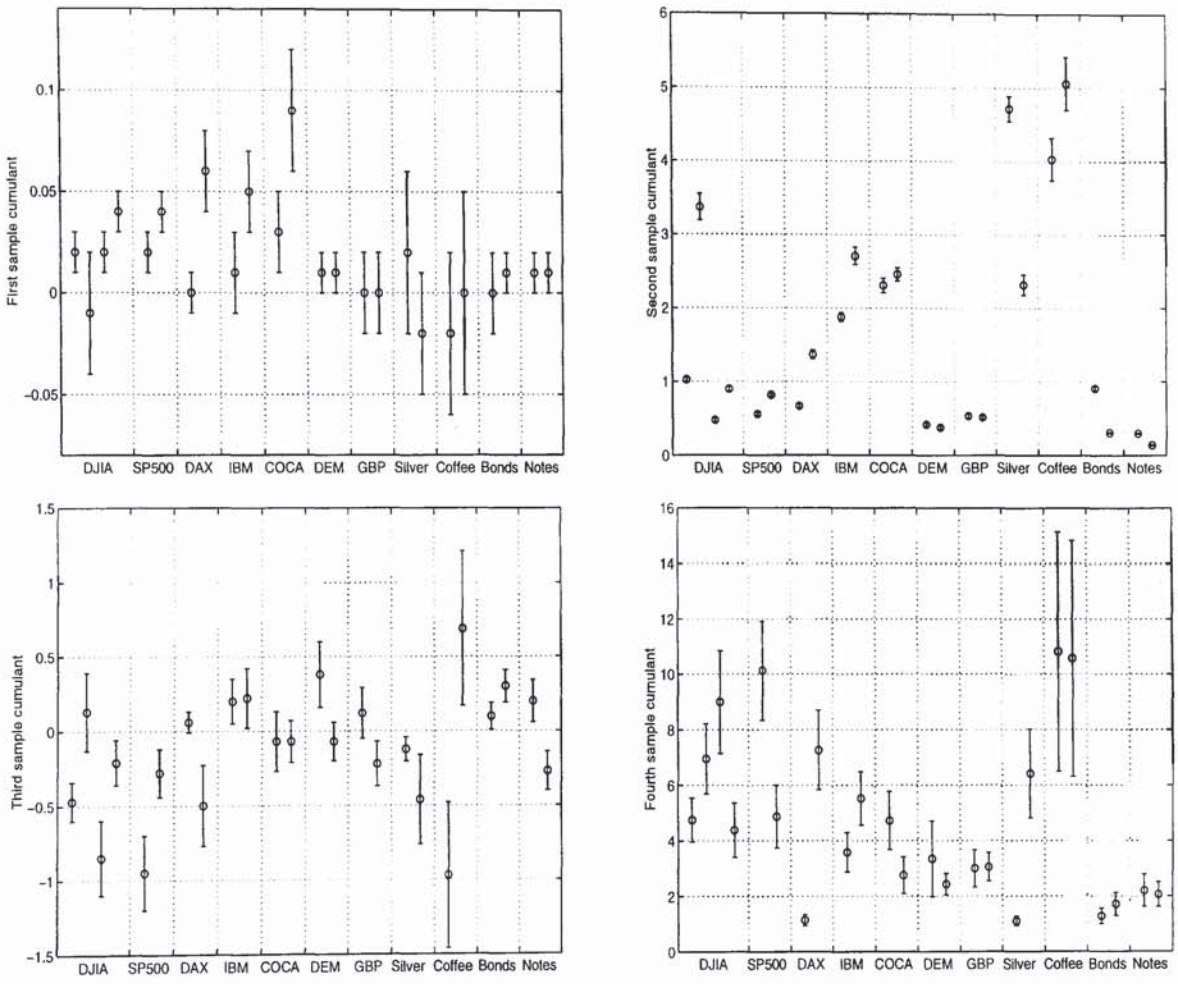

Figure 4.1: First four sample cumulants for all datasets

Concerning the *variance* it is interesting to note strong variations between the different datasets as well as within some of them. This emphasises the importance of a careful segmentation of the data in subsets of second-order quasi-stationarity during the preprocessing (cp. Section 2.1). Otherwise periods with different variances are averaged which will lead to questionable results. In summary, the prior of higher variance for commodities and single stocks compared to aggregated indices, currencies and bonds can be confirmed by the investigated examples.

For the *skewness*, it seems to be fair to conclude that all financial series tend to be slightly skewed though with no dominant direction apart from stock indices which on average seem to be slightly negatively skewed.

Regarding the *kurtosis* values significantly greater than zero can be found for all time series, with especially high values for the stock indices and the commodities. This confirms the notion of leptocurtic behaviour for financial returns. Summarising the results for all cumulants, it can be stated that the distribution of Bond returns is close to a Gaussian while all others investigated financial time series seems to deviate from this distribution significantly.

Another interesting, though not surprising, feature is the strong correlation in the results for set 3 and 4 of DJIA and set 1 and 2 of SP500, bearing in mind these sets cover exactly the same trading period. Since both indices cover the U.S. stock market some correlation can be expected. Nevertheless, it is remarkable that despite the differences in breadth and the way these two indices are calculated (cf. Appendix A), the statistical properties seem to be not significantly affected.

4.2.2 Probability density function

In the following we discuss first kernel-based methods for a non-parametric estimation of the probability density function and of the characteristic function. Finally, we will investigate the effect of using a finite number of the cumulants to approximate the characteristic function.

A naive non-parametric estimator for the probability density function $p(x)$ places a Dirac delta function[4] at each point x_t in the data set:

$$\hat{p}(x) = \frac{1}{T} \sum_{t=1}^{T} \delta(x - x_t). \tag{4.21}$$

For a practical estimator the delta function is replaced by a kernel $K(x) \geq 0$ which takes points in the neighborhood of x into account:

$$\hat{p}(x) = \frac{1}{T} \sum_{t=1}^{T} \frac{1}{h} K\left(\frac{x - x_t}{h}\right). \tag{4.22}$$

[4]The Dirac delta function δ is defined via its integral property $\int_{-\infty}^{\infty} f(x)\,\delta(x - x_0)\,dx = f(x_0)$.

with h as the kernel width. If the kernel K is a simple threshold function like

$$K(u) = \begin{cases} 1 & |u| < 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{4.23}$$

then Equation (4.22) estimates the *histogram* which we use here for comparative purposes. However, this is still a discontinuous representation of the true probability density. A smoothed estimate can be obtained by using kernels without strict membership functions, such as the Gaussian kernel

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2}. \tag{4.24}$$

One important parameter in this kernel approach is the kernel width h which has to be chosen carefully. If it is too small the density estimator will tend to overfit in low-density regions, while an h too big might smooth out some important characteristics of the density function.

4.2.3 Characteristic function

Using the representation of the probability density function in Equation (4.21) the delta estimate of the *empirical characteristic function* $\hat{\Phi}(t)$ can be obtained as the Fourier transform of $\hat{p}(x)$:

$$\hat{\Phi}_\delta(t) = \int_{-\infty}^{\infty} e^{itx} \left\{ \frac{1}{N} \sum_{t=1}^{T} \delta(x - x_t) \right\} dx = \frac{1}{T} \sum_{t=1}^{T} e^{itx_t}. \tag{4.25}$$

In analogy to using the kernel to approximate the p.d.f. the same can be done here by replacing the delta function in equation (4.25) with the kernel function $K(u)$ leading to

$$\hat{\Phi}(t) = \frac{1}{Th} \sum_{t=1}^{T} \int e^{itx} K\left(\frac{x - x_t}{h}\right) dx \tag{4.26}$$

which becomes for the Gaussian kernel from Equation (4.24)

$$\hat{\Phi}(t) = \frac{1}{Th\sqrt{2\pi}} \sum_{t=1}^{T} \int e^{itx} e^{-\frac{(x-x_t)^2}{2h^2}} dx = e^{-\frac{1}{2}t^2h^2} \frac{1}{T} \sum_{t=1}^{T} e^{itx_t} = \Phi_h(t)\Phi_\delta(t). \tag{4.27}$$

This is the convolution of the kernel function $K(\frac{x-x_t}{h})$ with the delta estimate $\Phi_\delta(t)$ for $\Phi(t)$ according to Equation (4.25). This makes it clear that computing the characteristic function from the data is not more efficient or accurate than the probability density function.

Another method for approximating the characteristic function is to calculate the sample cumulants and to reconstruct with them the power series defined in Equation (4.15). Since every non-Gaussian distribution has an infinite number of non-zero coefficients in this power series, such an approximation suffers from truncating higher-order cumulants which cannot be reliably determined. This leads to oscillations outside a certain interval around the origin. Increasing the order of the approximation is unfortunately not of great help since error in higher-order cumulants brings another imprecision.

ignored. For instance, the occurrence of too many extreme values was attributed to a different mechanism generating the returns. Consequently, these values were usually excluded from the analysis.

In the early 1960s Mandelbrot proposed using stable Paretian distributions as a wider class of distributions for modelling financial returns (Mandelbrot, 1963). This includes as special cases the Gaussian and Cauchy distribution. Recently, another class of stable distributions has been suggested which includes, for instance, the Weibull and Laplace distribution (Mittnik and Rachev, 1993). In contrast to the Gaussian both classes of stable distributions have the ability to model skewed and leptokurtic behaviour in financial returns.

The drawback of such more complex distribution models is the increased effort to fit their parameters. Here we use the maximum likelihood approach to estimate the model parameters, which is straightforward for the Gaussian and Laplace distributions since the parameter solutions can be written down explicitly. For Cauchy and Weibull distributions a quasi-Newton nonlinear optimisation technique has to be applied in order to maximise the likelihood.

For the general case of the stable Paretian distribution this needs furthermore a numerical approximation of the p.d.f. using the inverse Fourier transform of the characteristic function given in Equation (4.4) to compute the likelihood function and of its derivative with respect to the model parameters. Here a quasi-Newton optimisation is employed again to obtain optimal parameter estimates.

Since the maximum likelihood parameters are derived from samples only, the influence of the sampling error needs to be assessed. Therefore the Bootstrap approach introduced in Section 2.2.2 is applied to provide error bars for each parameter estimate.

4.3.1 Deterministic summation stable distributions

Two reasons should be pointed out for using stable distributions for modelling asset returns. The first is their already mentioned ability to model rich behaviour of distributions including the fat tails and asymmetry which have been observed for financial returns. The second is that the Generalised Central Limit Theorem points to stable distributions as the only possible non-trivial limit of normalised sums of independent identically distributed terms (Nolan, 1999). This means that a linear combination of copies X_i of a stable distributed variable X remains stable up to scaling and shift:

$$X \stackrel{\circ}{=} a_n(X_1 \otimes X_2 \otimes \cdots \otimes X_n) + b_n \tag{4.28}$$

with $a_n > 0$, $b_n \in \mathbb{R}$ and $\stackrel{\circ}{=}$ denoting distributional equivalence. Using summation as the operation $\otimes$ and a deterministic n, the class of stable Paretian distributions appears (sometimes also referred to as Lévy distributions). From this class we look at two special cases, the Gaussian and the Cauchy distribution.

PAGE

NUMBERING

AS ORIGINAL

Stable Paretian distributions

For the class of *stable Paretian distributions* one cannot write in general a closed form for the probability density function. Nevertheless, the log of its characteristic function is given as

$$\log \Phi(t) = \begin{cases} i\delta t - |\gamma t|^{\alpha} \left[1 - i\beta \operatorname{sgn}(t) \tan \frac{\pi\alpha}{2}\right] & \alpha \neq 1 \\ i\delta t - |\gamma t|^{\alpha} \left[1 + i\beta \operatorname{sgn}(t) \frac{2}{\pi} \log |t|\right] & \alpha = 1 \end{cases} \tag{4.29}$$

with scaling parameter $\gamma > 0$, mode $\delta \in \mathbb{R}$ and the sign function sgn[5]. The skewness $\beta \in [-1, 1]$ controls the symmetry of the distribution; hence for $\beta > 0$ the distribution is skewed right and has therefore a longer tail on the right then on the left, and for $\beta = 0$ we obtain symmetric stable distributions. Note that the skewness represented by β is not identical with the third-order cumulant, although both give usually the same qualitative result.

The *stability index* $\alpha \in (0, 2]$ determines the shape of the distribution: α is the slope of the tails in a log-log plot and determines the total probability contained in the tails of the distribution; the smaller α is, the more probability is contained. For $\alpha > 1$ the mode δ is equal to the mean μ otherwise the mean does not exist. All other moments exists only for $\alpha = 2$ which produces the normal distribution.

Fama (1965) lists three, unfortunately problematic, methods to estimate the stability index. First, α could be determined as the slope of $p(x)$ against x in a double-log plot since the tails of stable Paretian distribution follow the Pareto law: $\lim_{x\to\infty} P(X > x) \propto x^{-\alpha}$. This requires a number of observations in the tails not feasible for daily data. As a second method, range analysis is suggested, which determines the scaling behaviour of the interfractile range of accumulated returns. However, these statistics are biased for dependencies in the returns. The third method is sequential analysis, which looks at the scaling of the sample variance for an increasing sample size. Unfortunately, Fama concluded that this approach gives rather unreliable results.

Therefore, we estimate the four parameters of the stable distribution with a maximum likelihood nonlinear optimisation scheme. Chobanov *et al.* (1996) have applied the maximum likelihood approach to determine the stable parameters for currency exchange rates. Nolan (1997) describes in detail the approach and its properties. Thereby the characteristic function is numerically transformed from the Fourier space into the p.d.f. domain. This allows to calculate the likelihood for the whole data set. Then the partial derivatives with respect to the model parameters can be computed numerically (details can be found in Appendix C.1).

In order to estimate also error bars and those parameters the Bootstrap approach is applied here again. Therefore the maximum likelihood solution is computed for 100 Bootstrap sample sets drawn from the original returns dataset. For the nonlinear optimisation part the

[5]The sign function is defined as $\operatorname{sgn}(x) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\sin(tx)}{t} dt = -1$ for $x < 0$, 0 for $x = 0$ and 1 for $x > 0$.

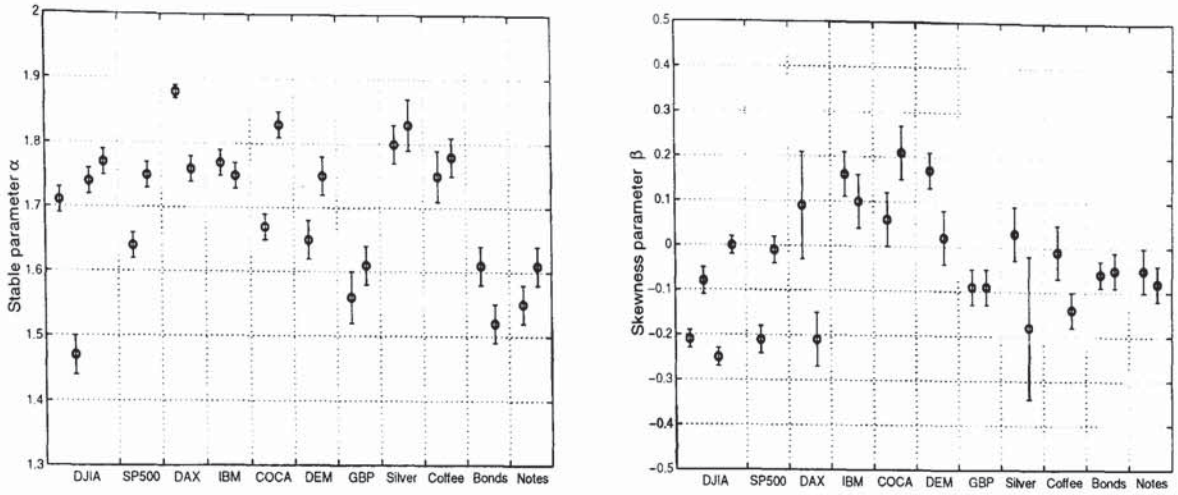

Figure 4.3: Stable parameters α and β for all datasets

quasi-Newton method is employed. The whole parameter estimation algorithm runs until convergence of the parameters α, β and γ. The mode δ can be computed as the sample mean $\hat{\mu}$. Assuming a normally distributed parameter statistics we report in Table D.5 the mean with one standard deviation of the 100 Bootstrap estimates for all datasets. Since one Bootstrap sample has to be reasonably big (at least 10^3 data points) and 100 Bootstrap runs should be performed as a minimum this approach is computationally very intensive. However, it achieves good results as long as the initial conditions are set properly. Here we chose to initialise $\alpha = 1.8$, $\beta = 0$ and $\gamma = 0.5$.

Figure 4.3 summarises graphically the results for the stable parameters α and the skewness parameter β. Regarding the stability it can be noticed that a majority of the values lies around the interval $[1.7, 1.8]$. The exceptions are set 2 for the DJIA (Great Depression period), the British Pound and the bonds time series which have a slightly lower stability.

For the skewness we see a similar outcome as for the third sample cumulants in section 4.2.1. There is a tendency for U. S. stock indices to be negatively skewed while the single stocks investigated here show positiv skewness. For the bonds and commodities a small negative bias for the skewness seems to be present. However, here the results differ slightly from those for the third sample cumulant. This is especially the case for the commodities were also relatively large error bars have been obtained.

The Gaussian distribution

The *Gaussian* distribution is the standard distribution for white noise processes due to its good approximation abilities for many real-world phenomena and its analytical properties (*e.g.*, being stable under linear transformations, being the limit distribution for the sum of i.i.d. random variables with finite variance). The Gaussian appears as a stable distribution for $\alpha = 2$ with mean $\mu = \delta$ and variance $\sigma^2 = 2\gamma^2$. Its characteristic function is $\Phi(t) = e^{i\mu t - \frac{1}{2}\sigma^2 t^2}$

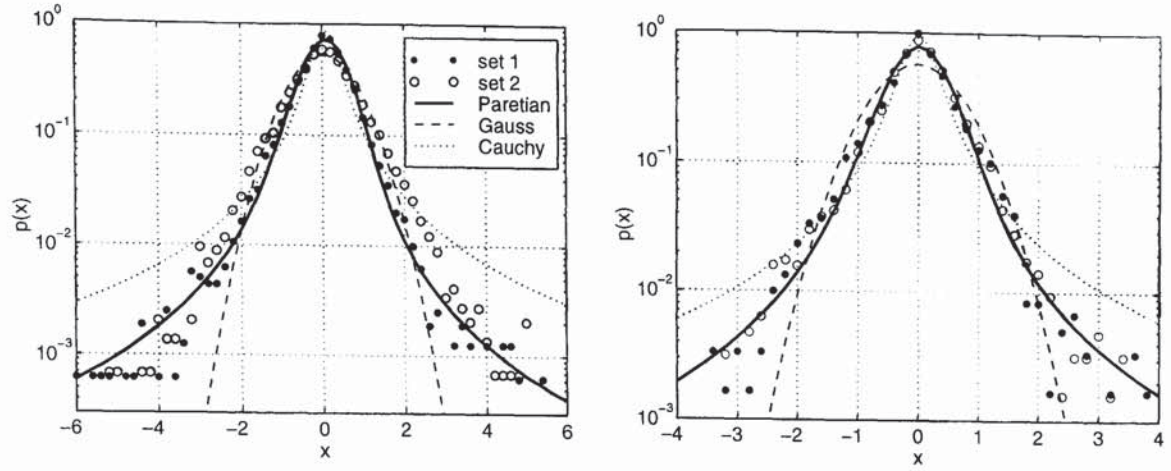

Figure 4.4: Stable distribution estimate for SP500 (left) and GBPUSD (right): Histogram for set 1 (training) and set 2 (test) together with the numerical approximation of the p.d.f. for the stable, the Cauchy and the Gaussian distribution fitted on the training set

and the probability density function is given by

$$p(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{4.30}$$

The Cauchy distribution

The *Cauchy* or *Lorentzian* distribution is a symmetric stable distribution with $\alpha = 1$ and $\beta = 0$. Its c.f. is given by $\Phi(t) = e^{i\delta t - |\gamma t|}$. From that the probability density is obtained as

$$p(x;\delta,\gamma) = \frac{\gamma}{\pi\left(\gamma^2 + (x-\delta)^2\right)} \tag{4.31}$$

Note that δ is the mode respectively the median of the distribution and that γ and δ are not related to the variance or the mean of the distribution since these do not exist.

The general fitting capabilities of the Gaussian, Cauchy and general stable Paretian distribution for financial returns is demonstrated for two examples of SP500 and GBPUSD in Figure 4.4. There neither the Cauchy nor the Gaussian seem to achieve a good modelling result while the stable Paretian distribution fits the density remarkably well.

The results for the Gaussian, the Cauchy and the stable Paretian distribution for all datasets in terms of estimated parameters and corresponding negative log-likelihood are included in Table D.3, D.7 and D.5. Note that the parameters for the Gaussian are identical with the first two cumulants in Table D.1. Therefore they are not reported here again.

4.3.2 Random summation stable distributions

If the number of variables n in Equation (4.28) is a random variable itself rather than deterministic the class of *random summation stable distributions* appears. Chobanov *et al.* (1996) motivate this random summation scheme by the assumption that financial markets may change their probabilistic structure randomly in time. Therefore the price changes should

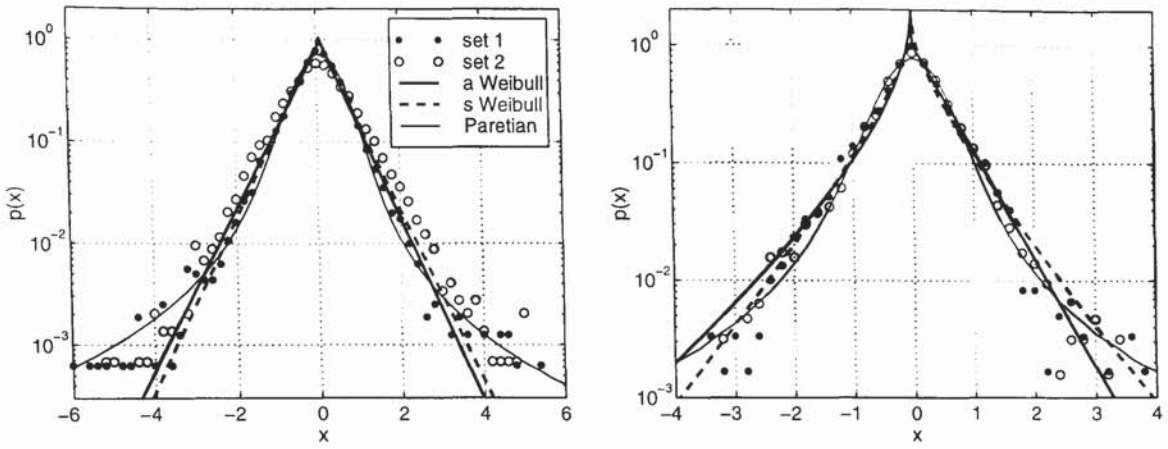

Figure 4.5: Random stable distribution estimate for SP500 (left) and GBPUSD (right): Histogram for set 1 (training) and set 2 (test) together with the pdf estimate for the asymmetric (a) and symmetric (s) Weibull distribution fitted on the training set

only be accumulated until the time horizon n for which the structure is intact. Modelling now the time horizon n as a random variable with a geometric distribution leads to the random summation scheme in contrast to the deterministic summation for the stable Paretian distribution family.

The Weibull distribution

One representative of the class of random summation stable distributions is the *symmetric (double-sided) Weibull* distribution given by its density function

$$p(x;\alpha,\lambda) = \frac{1}{2}\lambda\,\alpha\,|x-\mu|^{\alpha-1}e^{-\lambda|x-\mu|^{\alpha}} \tag{4.32}$$

with scale $\lambda > 0$, shape parameter $\alpha > 0$ and shift $\mu \in \mathbb{R}$. In case of $\alpha < 1$ it is defined only for $x \neq \mu$. The symmetric form assumes that negative and positive returns (after the shift) have the same distribution. Mittnik and Rachev (1993) report good modelling properties for the Weibull distribution applied to the S&P 500 stock index compared to several other parametric distributions including the stable Paretian one.

The Laplace distribution

For $\alpha = 1$ the *symmetric Laplace* or *double exponential* distribution appears from the Weibull distribution with its probability density function

$$p(x;\mu,\lambda) = \frac{\lambda}{2}\,e^{-\lambda|x-\mu|}. \tag{4.33}$$

Figure 4.5 shows the fit for the Weibull distribution on the previously used two datasets SP500 and GBPUSD. Since the Laplace fit is virtually identical with the Weibull fit (cf. with the near unity value for α) only the Weibull fit is plotted in its symmetric and asymmetric form.

4.4 Mixture models

The limitations of parametric and non-parametric density estimation techniques in terms of requirements for data or prior knowledge motivate the use of a combination of elements of both approaches. In contrast to kernel-based techniques, in a mixture model only a small number of basis functions is used to represent the density. This reduces the computational cost of computing the p.d.f. and restricts the model complexity. On the other hand, an arbitrarily large number of kernels still allows a greater flexibility than most parametric models especially for multi-modal distributions.

Here we are interested in mixture models as an efficient density estimation technique since this will be needed later in an automated procedure for density representation. Therefore the standard Gaussian mixture model will be described briefly, first. Then we will present a mixture model composed of several Gaussian and one Laplace component for which the EM algorithm will be sketched. Finally, a modified version of the EM algorithm is proposed in order to allow a distribution estimate with mixture models for weighted samples.

In a mixture model the probability density $p(x)$ is approximated as a sum of M parametrised density functions $p(x \mid j)$ weighted by their corresponding prior probabilities $P(j)$:

$$p(x) = \sum_{j=1}^{M} p(x \mid j)\, P(j). \tag{4.34}$$

The model parameters specifying the component densities and prior probabilities can be estimated in a maximum likelihood framework. Using θ to denote the vector of all model parameters, the vector θ has to be found which maximises the likelihood $\mathcal{L}(\theta)$ of the model given the data. Assuming that the dataset $\mathcal{X}$ consists of i.i.d. samples x_t, this likelihood is equivalent to the joint probability of all the samples x_t:

$$\mathcal{L}(\theta) \equiv p(\mathcal{X} \mid \theta) = \prod_{t=1}^{T} p(x_t \mid \theta). \tag{4.35}$$

This is equivalent to minimising the error function E defined as the negative log-likelihood for which the expectation-maximisation (EM) algorithm was proposed by Dempster *et al.* (1977). Further details of the EM algorithm for Gaussian mixtures can be found in Appendix C.6

The fit of a typical Gaussian mixture model is shown in Figure 4.6 for the SP500 and GBPUSD returns. It can be seen that the model has achieved a smooth representation of the data with just three Gaussian components. Nevertheless, nonstationarity, for instance in the form of larger variance of the SP500 test set compared to the training set, limits the model's ability to generalise. In contrast, for the GBPUSD time series the generalisation seems to be successfull. Mixture models share this problem with all other models which assume stationarity, achieving therefore only suboptimal results.

One practical question related to generalisation is, of course, how many components to use in the mixture and if all of these should be actually Gaussians. Earlier we have shown

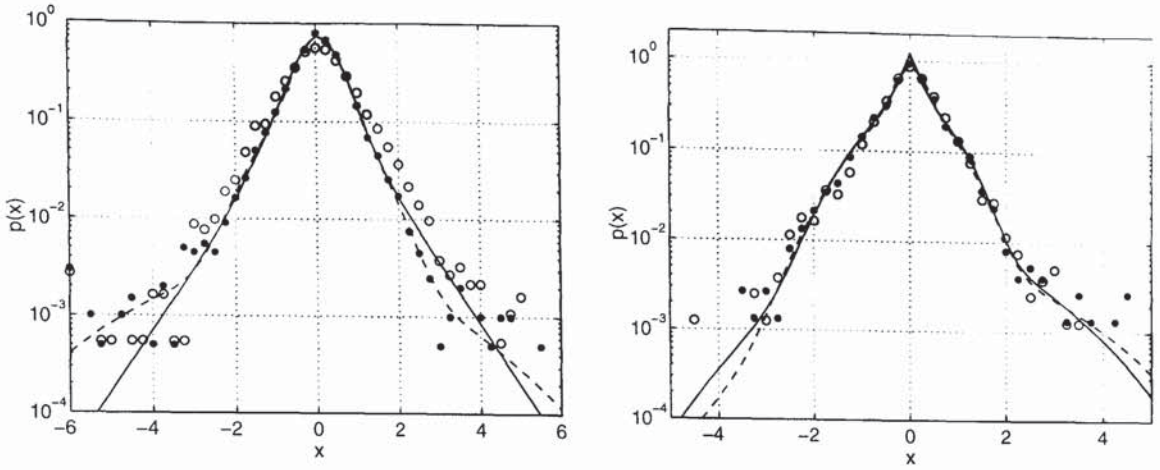

Figure 4.6: Mixture model density estimates for SP500 (left) and GBPUSD (right): Estimates for the pdf via a GMM (dashed) with three components and a GLMM (solid) with two Gaussians and one Laplace component for set 1 (training) compared to the histogram for set 1 (dot) and set 2 (circle).

that financial returns typically possess a significant leptocurtic distribution. For those we have furthermore noticed that mixtures with few Gaussian basis functions have difficulties modeling the peaked shape around the centre of the distribution and the non-Gaussian tails.

Since the Laplace distribution introduced in Equation (4.33) achieves good modelling results especially in the tails and the centre of the distribution we suggest to use one Laplace component in a combined mixture model. This will save the use of several Gaussian components in order to approximate leptocurtic behaviour. The result is a less complex model since fewer parameters have to be determined. The modification of the EM algorithm in order to update the Laplace component is straight forward, the details are given in Appendix C.7.

Figure 4.6 shows, beside the Gaussian mixture approximation, also the Gaussian-Laplace combined mixture fit. It turns out that while both approaches model the centre of the distribution almost identically, they differ in the tails. While the pure Gaussian mixture still models the log probability in the tails eventually as declining quadratically, the combined mixture achieves a linear fit.

In order to evaluate in more detail the quality of these two model classes Figure 4.7 shows the modelling error as a function of the number of components in the mixture. Since mixture models are sensitive to initial conditions we report here the mean of the log likelihood with errorbars for 100 runs of each model. It becomes clear that a single Laplace provides a better fit than a single Gaussian. Furthermore, using a two component model the Gauss-Laplace mixture is still superior to the pure Gaussian one. However, using three or more Gaussians in each model achieves similar results and also does not seem to change the log likelihood significantly.

Furthermore, this figure confirms the results for the fit of the probability density in Figure 4.6. The likelihood results are quite different for the training and test set of SP500 returns

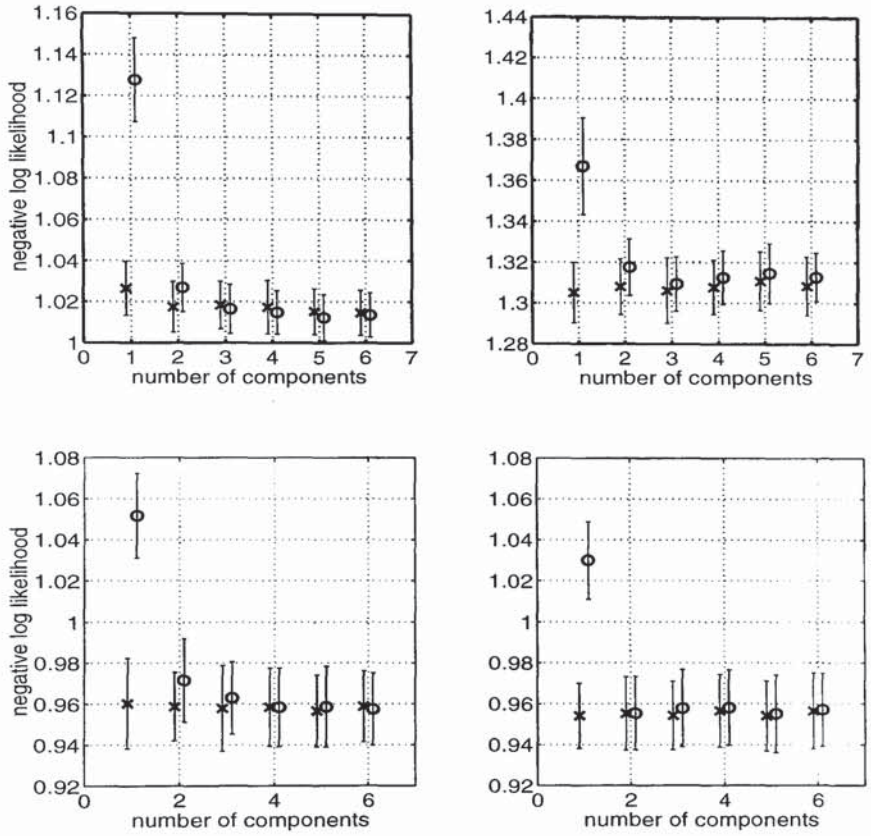

Figure 4.7: Mixture model likelihoods for SP500 (top) and GBPUSD (bottom): mean of the negative log likelihood and errorbars as one standard deviation for 100 runs of a GMM (circle) and a GLMM (cross) for set 1 as training data (left) and set 2 as test data (right)

compared to the GBPUSD data. There an almost identical likelihood has been achieved on the test set. This could be the effect of a present nonstationarity in the stock market. Such a possibility will be addressed in a later chapter.

4.5 Conditional density estimation

In theory, all of the non-parametric and semi-parametric techniques for density estimation discussed so far can be extended to multi-dimensional cases. A practical problem occurs here since the number of data points required for a reliable density estimate grows exponentially with the dimension of the data. For this reason the estimation of conditional probabilities is practical for low dimensions only. Furthermore, we have previously demonstrated the difficulties within the kernel density approach to determine optimal kernel widths. For detecting dependency in financial returns we will therefore consider here only the usage of Gaussian mixture models and multivariate cumulants.

For the purpose of demonstrating short-term dependencies in financial time series we additionally restrict ourselves to one independent and one dependent variable. The aim is therefore to determine the conditional probability $p(y \mid x)$ of obtaining y given x. Assuming the joint density $p(x, y)$ has been already determined, the conditional density $p(y \mid x)$ can be derived via Bayes' theorem

$$p(y \mid x) = \frac{p(x, y)}{\int p(x, y)\, dy}. \tag{4.36}$$

For unimodal or highly peaked conditional distribution where a Gaussian approximation is feasible a prediction for the most likely value can be made as the expectation of obtaining y while having observed the value x:

$$\hat{y} = \mathbb{E}[y \mid x] = \int_{-\infty}^{\infty} y\, p(y \mid x)\, dy. \tag{4.37}$$

In a similar way the conditional variance $\mathbb{E}\left[(y - \hat{y})^2 \mid x\right]$ can be obtained via

$$\hat{\sigma}_y^2 = \mathbb{E}\left[(y - \hat{y})^2 \mid x\right] = \int_{-\infty}^{\infty} (y - \hat{y})^2\, p(y \mid x)\, dy. \tag{4.38}$$

In order to apply this approach for time series, x will be identified with the current value of the time series x_t and y represents the next value x_{t+1} thus we are interested in $p(x_{t+1} \mid x_t)$. Then the joint distribution of consecutive values $p(x_t, x_{t+1})$ and the marginal distribution $p(x_t)$ are estimated, finally the conditional distribution $p(x_{t+1} \mid x_t)$ can be determined via Equation 4.36.

4.5.1 Multi-dimensional Gaussian mixture models

In order to illustrate the capability of Gaussian mixture models for conditional probability density estimation two-dimensional mixture models have been trained on the first dataset of

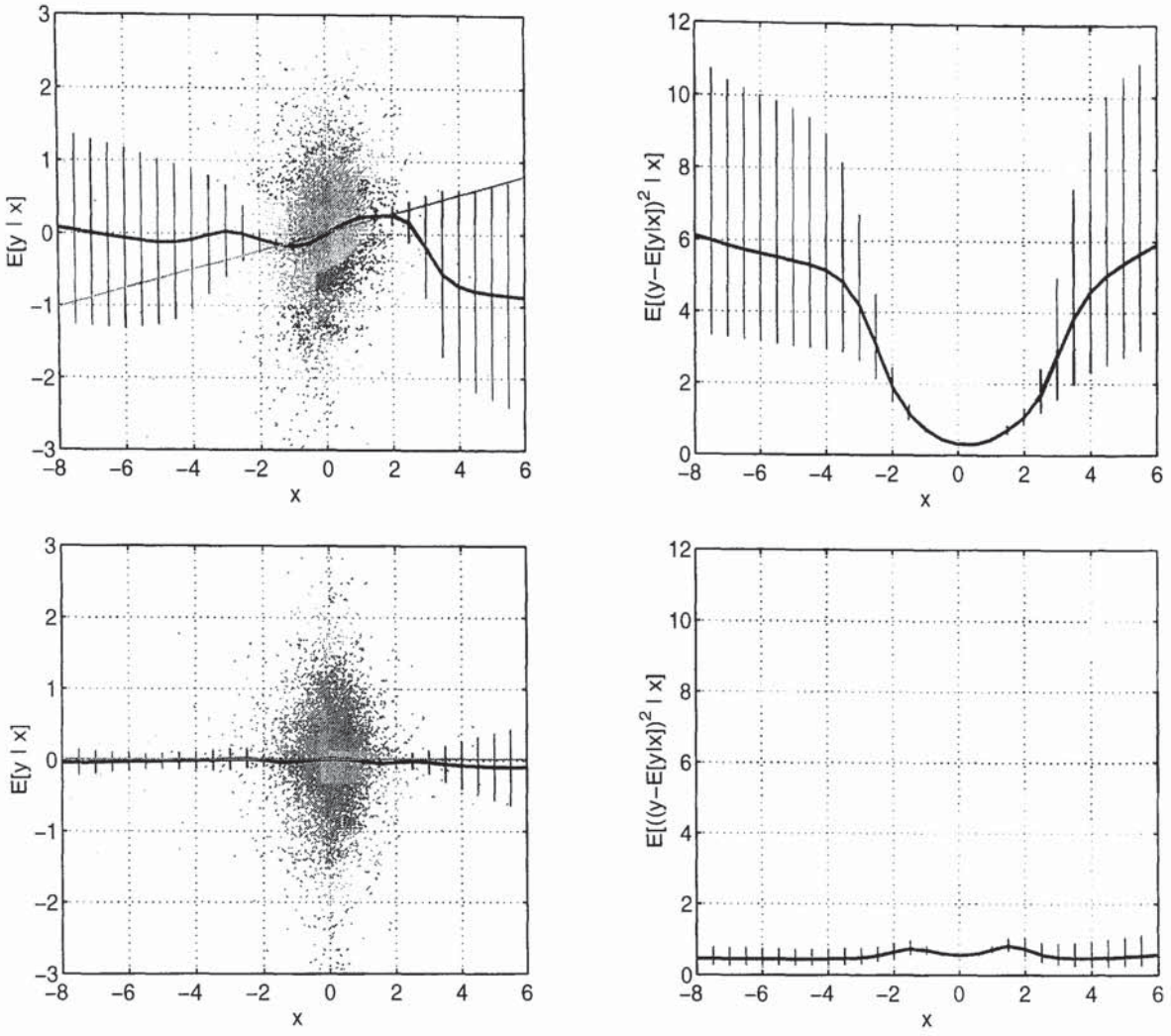

Figure 4.8: Expectations for the mean $\hat{y} = \mathbb{E}[y \mid x]$ and the variance $\mathbb{E}\left[(y - \hat{y})^2 \mid x\right]$ for the conditional probability density for the first set of SP500 returns (top) and the surrogate version (bottom) estimated by 100 runs of a 2d Gaussian mixture model with 7 components, trained each for 25 iterations

S&P500 returns and a surrogate version (sampling with replacement). Figure 4.8 shows the average of the mean expectation over 100 runs with the one standard deviation errorbars.

There it becomes clear that there is some positive correlation between today's and tomorrow's return which can be reliably estimated in the range $[-2, 2]$ and is additionally confirmed by a linear regression. Furthermore, this correlation also tends to be slightly nonlinear in the tails suggesting that an extreme daily return is usually compensated the next day by a return with the opposite sign.

In contrast to these findings, a hypothesis of independent x_t and x_{t+1} implies that the conditional distribution $p(x_{t+1}|x_t)$ should be represented by an approximately constant distribution equal to the marginal distribution $p(x_{t+1})$. Therefore the distribution's mean and variance should be constant, too. However, the volatility or conditional variance also varies and is strongly positively correlated with the amplitude of todays return. This corresponds

Performance criterion	Approach	S&P500	Surrogate
Normalised mean squared error	linear	0.988	1.000
	nonlinear	0.993	1.798
Annualised excessive profit in %	linear	14	0
	nonlinear	11	0
Correct sign prediction in %	linear	54	52
	nonlinear	54	52

Table 4.1: Performance results for linear and nonlinear regression of 1-day-ahead returns using a Gaussian mixture model for the test set of the S&P500 data compared to a surrogate version using sampling with replacement

to the common observation of persistence in volatility in financial time series.

The predictions made with this nonlinear model have been evaluated with three different error criteria. Table 4.1 summarises the results: The main error criterion, the normalised mean squared error is for both regressions just slightly smaller than one. This corresponds also to a just above chance number of correct sign predictions.

The annualised excessive profit has been calculated by accumulating each day's return multiplied by the sign of the prediction assuming zero transaction costs. The total return is then reduced by the gain of the underlying equity itself. In order to annualise the total excessive return this quantity is divided by the total number of trading days in the period considered and multiplied by the number of annual trading days (253). Remarkable is here the poor performance of the nonlinear approach compared to the linear one. One can attribute this result to either overfitting or a non-stationarity. However, the estimate for the conditional mean shown in Figure 4.8 is of a quite typical form which has been confirmed also for other time series. Therefore we are going to focus on the issue of nonstationarity next by using a test of the change of linear and nonlinear correlation over time.

4.5.2 Non-parametric independence test

Deco *et al.* (1997) proposed a non-parametric approach based on higher-order cumulants to test statistical dependency in univariate financial time series. This test is briefly introduced since we will propose here a Bootstrap estimation of the cumulants in order to obtain a reliable statistics about the evolution of correlation in financial time series.

In order to estimate cumulant tensors for a scalar time series, X_i is identified as the variable X lagged by i steps in time, which leads to the embedding approach introduced in Chapter 3. Hence, for a given time series $\mathcal{X}$ embedding vectors $\boldsymbol{x}_t$ of dimensionality m and with time lag Δt are constructed according to Equation (3.4). A generic $\boldsymbol{x}$ consists therefore of the single components $x_1, \ldots, x_m$. The null hypothesis is now defined as independence in successive values of the time series. Therefore, the joint probability density $p(\boldsymbol{x}) = p(x_1, x_2, \ldots, x_m)$ should be equal to the product of the single density $p(x_1)$ and the

remaining joint density $p(x_2, \ldots, x_m)$:

$$H_0 \equiv p(x_1, x_2, \ldots, x_m) = p(x_1)\, p(x_2, \ldots, x_m). \tag{4.39}$$

Testing this null hypothesis directly in the probability space means to estimate the density, which might be difficult for a high data dimension m due to the exponentially growing number of data points required for a reliable estimate.

Using the Fourier space instead, the independence condition can be expressed in terms of the characteristic functions involved. As for probabilities, the characteristic function of independent components is the product of the characteristic functions for each component (Stuart and Ord, 1994), and the null hypothesis in Equation (4.39) therefore becomes

$$\log \Phi(t_1, \ldots, t_m) = \log \Phi(t_1) + \log \Phi(t_2, \ldots, t_m). \tag{4.40}$$

Expanding this in multi-dimensional cumulant tensors $\kappa_{j_1 \ldots j_n}$ defined in Equation (4.18) and marginal cumulants in Equation (4.15) it becomes

$$\sum_{n=1}^{\infty} \frac{i^n}{n!} \sum_{j_1, \ldots, j_n = 1}^{m} \kappa_{j_1 \ldots j_n}\, t_{j_1} \ldots t_{j_n} = \sum_{n=1}^{\infty} \frac{i^n}{n!} \left\{ \kappa_1^{(n)}\, t_1^n + \sum_{j_1, \ldots, j_n = 2}^{m} \kappa_{j_1 \ldots j_n}\, t_{j_1} \ldots t_{j_n} \right\} \tag{4.41}$$

with the one-dimensional cumulant $\kappa_j^{(n)}$ of order n for the vector element j defined in Equation (4.14). Note that here we use now $\kappa_j^{(n)}$ corresponding to t_j instead of κ_n. Re-arranging this and writing for the scalar cumulant $\kappa_j^{(n)} = \kappa_{j_1 \ldots j_n}$ with $j_1 = \cdots = j_n = j$ we get

$$\sum_{n=1}^{\infty} \frac{i^n}{n!} \sum_{j_2, \ldots, j_n = 1}^{m} (1 - \delta_{1 j_1 \ldots j_n})\, \kappa_{1 j_2 \ldots j_n}\, t_1 t_{j_2} \ldots t_{j_n} = 0 \tag{4.42}$$

with $\delta_{j_1, \ldots, j_n}$ as Kroenecker's delta[6]. Since this has to be fulfilled for all possible $\boldsymbol{t}$, the coefficients $(1 - \delta_{1 j_2 \ldots j_n})\, \kappa_{1 j_2 \ldots j_n}$ have to be zero for all $j_2, \ldots, j_n = 1, \ldots, m$. This is equivalent to testing the deviation of the relevant cumulants from zero and building the following cost function

$$s = \sum_{n=1}^{\infty} \sum_{1 \le j_2 \le \cdots \le j_n = 2}^{m} \kappa_{1 j_2 \ldots j_n}^2. \tag{4.43}$$

Here only cumulants up to fourth order are considered since higher cumulants suffer from the estimation problem. Furthermore, we modify this approach slightly by calculating the cost functions s_2, s_3 and s_4 separately for each cumulant rather than summing up the deviations for each cumulant in Equation (4.43), as otherwise the higher-order cumulants dominate the lower-order ones. Therefore it seems to be more expressive to look at each cumulant individually.

Another variation is used here concerning the calculation of the significance of the cumulants' deviations. Deco *et al.* (1997) suggested to compute this significance by normalising

[6]Kroenecker's delta is defined as $\delta_{j_1 \ldots j_n} = 1$ for $j_1 = \cdots = j_n$ and 0 otherwise.

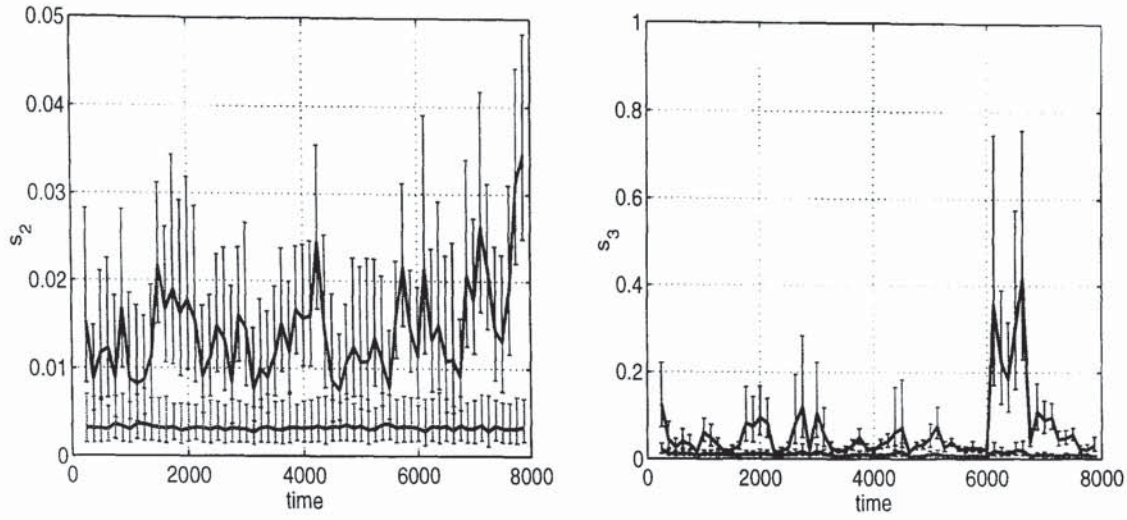

Figure 4.9: Second and third-order cumulants statistics for SP500 returns of set 1: Original statistics shows the mean with one standard deviation errorbars for 100 bootstrap runs of the original data. The surrogate statistics represents the mean and one standard deviation errorbar for 100 runs of randomised data (sampling with replacement). The mean for the original data for both cumulants can be recognized as being always above the one for the surrogate data and showing a bigger variation in its amplitude. Each statistic was computed for an embedding vector of six successive returns, a moving time window of 250 points which overlaps to 50%.

the original statistics with the mean and standard deviation of the results on a number of surrogate datasets. Since the statistics is rather log-normal distributed than normal we propose here to use the mean and standard deviation of the log of the statistics. Figure 4.9 shows the results for the second and third cumulant for the SP500 time series. There the scale is the original one, mean and standard deviation have been computed in the log space and transformed back to the original space via exponentiation.

It can be seen there that the correlation is varying significantly with time and in comparison, that the statistics for the randomised data are relatively stable for the second and third cumulant. The results for the fourth order cumulant are quite similar to those for the third order and are therefore not shown here. The results demonstrate that over relatively short periods of time there is a correlation not explainable by the independence assumption. This correlation shows a certain persistence[7] and therefore could be used for predictive purposes.

The idea by Deco *et al.* (1997) to select with this approach those periods which show significant correlation might be useful in order to train a model with such a subset instead of presenting all data. However, since only a fraction of the original time series would be selected we will pursue here a rather different approach in the next chapter. There a model will be discussed which assumes a certain temporary state corresponding to a specific correlation.

[7]Having a window overlap of 50% means there should not be any correlation between every other window.

4.6 Discussion

This chapter has provided a detailed discussion of analysing the distribution of daily financial returns and their properties. Under the assumption of independent and identically distributed returns different techniques for estimating the marginal probability density were introduced first in order to determine features of the return distribution such as shape, mode and symmetry. Finally, leaving this assumption we investigated the hypothesis of independence and identity of the distribution within a multivariate framework.

Concerning the properties of financial return it has been confirmed with the non-parametric technique of cumulants that their mean is near zero and that the variance strongly varies for different types of the underlying asset. Furthermore, a significantly positive kurtosis can be found for almost all financial time series indicating heavier tails and higher peaks than the normal distribution (leptocurtic).

These findings have been confirmed also by the kernel-based density estimation techniques. There problems have been noticed regarding the determination of hyper-parameters such as the kernel width, for example, and the quality of the estimate for low density regions attributed to the lack of samples and a model for the data. Therefore parametric density models were fitted to the data as well with superior results for the stable Paretian and the Laplace distribution compared to the Gaussian and the Cauchy distribution.

The experiments with mixture models have shown that they are in general capable of modelling heavy-tailed distributions arbitrarily well with a few basis functions only. This represents therefore a more efficient technique than kernel density estimators. A combined Laplace-Gaussian mixtures is capable of modelling leptocurtic distributions with fewer component than a pure Gaussian mixture. However, the difference in terms of the likelihood vanishes for a higher number of components. Therefore the standard Gaussian mixture model will be used during the remainder of this thesis.

Testing the hypothesis of independence in successive daily returns we found that they are slightly positively correlated, especially around the mean of the distribution. Even stronger is the correlation in their magnitude. However, comparing a linear with a nonlinear fit, represented by a two-dimensional Gaussian mixture model, the nonlinear model does not seem to be superior.

Chapter 5

Static factor models

Stock prices in a specific market segment are often significantly correlated due to, for instance, similarity in their business profile. One can then ask for the cause of these correlations and speculate that those are induced by a common factor which is hidden. The task at hand would then, for instance, to extract from the observed stock prices those common hidden factors.

This chapter discusses the application of principal component analysis and independent component analysis for univariate financial time series. In order to perform single-channel versions of these techniques, we work within the embedding framework, using delay coordinate vectors to obtain a multidimensional representation of the system dynamics at each time instance. The main objective is to investigate if these techniques are able to perform feature extraction, signal-noise-decomposition and dimensionality reduction since that would enable a further inside look into the behaviour and mechanics of financial markets.

5.1 Introduction

Three main problems have been identified for restricting the progress in the analysis of financial time series: the existence of nonlinear behaviour between financial variables, the nonstationarity of relationships among the relevant variables, and a low signal-to-noise ratio.

The limited results achieved so far by basic neural networks architectures for forecasting prices of financial equities based upon their past can be attributed to these three characteristics: noise limits the amount of information which can be extracted at each time instance, nonstationarity restricts the number of data points in time used to filter out the noise in order to disclose the deterministic components in the data, and nonlinearity couples the degrees of freedom in the system preventing model simplifications by divide-and-conquer strategies.

These considerations motivate the use of unsupervised feature extraction methods to transform the problem from the time domain to an alternative space capable of revealing 'interesting' structure in the data. The extracted features can then be used for forecasting

purposes. The role of the feature space is to provide an easier signal-noise-decomposition than in the time domain and with that a dimensionality reduction. Furthermore, it could allow an explicit modelling of nonstationarity when time is taken into account as an independent variable.

Here two methods for feature extraction are investigated: principal component analysis and independent component analysis. These techniques will be used to decompose financial time series into 'interesting' components which could then be connected to *e.g.*, political, economical or psychological factors influencing financial markets.

Previous attempts in this domain have been restricted to looking at ensemble methods using multiple time series from specific markets (Back and Weigend, 1997). In contrast, here univariate data are used and therefore single-channel versions of these two algorithms are performed. In order to do this, the embedding framework, introduced in Chapter 3 is again employed by using delay coordinate vectors. Thus, both methods are applied on a $d \times n$ embedding matrix $\boldsymbol{Y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$ consisting of embedding vectors $\boldsymbol{y}_t \in \mathbb{R}^d$ defined in Equation (3.4).

5.2 Factor analysis

Factor analysis is at the core of several generative models from which specialised cases such as principal and independent component analysis can be developed. The key assumption here is the complete temporal independence of observations and their underlying factors. The factors are hidden, uncorrelated variables $\boldsymbol{x}_t$ which are modeled for simplicity as a multivariate Gaussian random variable with zero mean and unit covariance:

$$\boldsymbol{x}_t = \boldsymbol{\epsilon}_t, \tag{5.1}$$

$$\boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \tag{5.2}$$

The observations $\boldsymbol{y}_t$ are functions of the factors contaminated with zero mean Gaussian noise with covariance matrix $\boldsymbol{\Sigma}$:

$$\boldsymbol{y}_t = \boldsymbol{G}\boldsymbol{x}_t + \boldsymbol{\eta}_t, \tag{5.3}$$

$$\boldsymbol{\eta} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), \tag{5.4}$$

using a linear observation function $\boldsymbol{G}$. The observations will therefore be noisy and due to $\boldsymbol{G}$ possibly intra-correlated. The task at hand is to determine the hidden variables $\boldsymbol{x}_t$ given the observations $\boldsymbol{y}_t$. In order to do that the structure of the covariance matrix $\boldsymbol{\Sigma}$ has to be restricted in some way. Only with a proper constraint is the model forced to distinguish between signal and noise. Otherwise the model would simply take the empirical covariance of the observations as the estimate of $\boldsymbol{\Sigma}$ and set the observation function $\boldsymbol{G}$ to the identity.

There are several possible restrictions of the covariance matrix. The standard for the common factor analysis is to use a diagonal matrix $\boldsymbol{\Sigma} = \text{diag}(\sigma_1, \ldots, \sigma_d)$. Going a step further to assume equal variance in each dimension of the data leads to a scalar covariance matrix $\boldsymbol{\Sigma} = \sigma \boldsymbol{I}$ which corresponds to the *sensible* or *probabilistic principal component analysis* (Roweis and Ghahramani, 1999; Tipping and Bishop, 1997). For the limiting case $\boldsymbol{\Sigma} = \lim_{\sigma \to 0} \sigma \boldsymbol{I}$ it can be shown that the solutions are the eigenvectors of the empirical covariance matrix, in other words, the principal components.

The latter approach will be discussed next in more detail. Afterwards one extension is outlined for including the non-Gaussian case, independent component analysis. A following chapter will also investigate models, such as the Kalman filter, which allow temporal correlation between the observations.

5.3 Principal component analysis

Principal components represent those orthonormal axes of $\boldsymbol{Y}$ onto which the retained variance under projection is maximal (Jolliffe, 1986). Assuming a zero-mean $\boldsymbol{Y}$ the principal components can be obtained as the d eigenvectors $\boldsymbol{u}_i$ of the covariance matrix $\boldsymbol{C}$ given by

$$\boldsymbol{C} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{y}_i \boldsymbol{y}_i'. \tag{5.5}$$

The eigenvectors form as column vectors the eigenmatrix $\boldsymbol{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_d) \in \mathbb{R}^{d \times d}$ whose inverse $\boldsymbol{U}^{-1} = \boldsymbol{U}'$ maps the dataset $\boldsymbol{Y}$ into the feature or hidden factor space. The result is a source matrix $\boldsymbol{X}$ of (up to second order) decorrelated source vectors $\boldsymbol{x}_i$ given as

$$\boldsymbol{X} = \boldsymbol{U}' \boldsymbol{Y}. \tag{5.6}$$

The corresponding eigenvalue λ_i is the variance of the i^{th} of d rows in $\boldsymbol{X}$ and therefore expresses the relevance of the eigenvector for this projection. Performing now a projection of an input vector $\boldsymbol{y}_i$ with only the first $q < d$ dominant eigenvectors $\boldsymbol{u}_i$ a mapping is defined from the d-dimensional $\boldsymbol{y}_i$ onto a q-dimensional vector $\boldsymbol{x}_i$ defining a subspace of the original input space

$$\boldsymbol{x}_i = \boldsymbol{U}_q' \boldsymbol{y}_i. \tag{5.7}$$

In this way, PCA can be viewed as a linear mapping from $\mathbb{R}^d$ to $\mathbb{R}^q$ and performs therefore a linear dimensionality reduction. Such a projection can then be further analysed instead of the original one. By using the PCA approach a linear reduction in the dimensionality of the dataset is achieved though the most relevant information was kept. That makes it easier to analyse the data in terms of reduced computational resources and model complexity. Therefore very often this technique is used for preprocessing purposes.

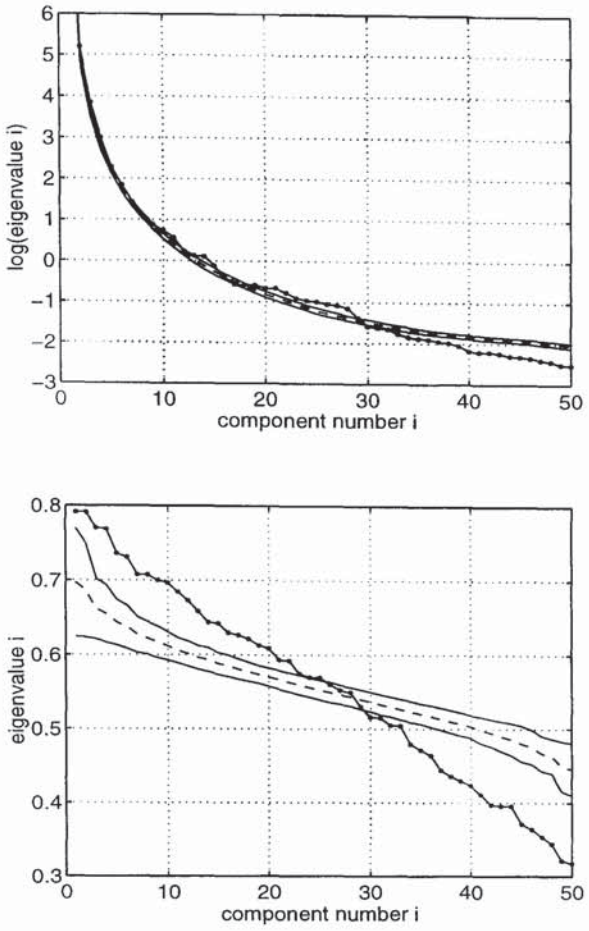

Figure 5.1: Eigenspectrum for set 1 (left) and set 2 (right) of the SP500 data as the log of the eigenvalues for the prices (top) and as the eigenvalues for the returns (left) of the original dataset (dotted) compared to the statistics for 100 runs of resampled returns indicated by the mean and two standard deviations (thin)

Following (Broomhead and King, 1986) in this feature space the signal is represented by the most 'important' components, while the noise is accounted for in the least ones. In PCA importance is defined as the size of the eigenvalue, since it represents the proportion of variance explained by the corresponding principal component. The plot of the sorted eigenvalues against their number is called the *eigenspectrum* which can be used to perform a signal-noise-decomposition (Cattell, 1966). For a stochastic system a smooth (exponential) decline of the eigenvalues is expected. Any deviation from that in form of *e.g.*, a sharp, discontinuous decline, is an indication of deterministic structure in the data.

Figure 5.1 shows the eigenspectrum for the prices and returns of the two SP500 datasets and compares them to the ones expected for a truly random process. To simulate such a process randomly resampled returns were used. Here we used both log prices and returns and created embedding vectors of dimension 50 with a one day lag. Minor deviations can be observed here for the prices, for example around component number 10 for set 2 and around component number 25 for set 1. Additionally, the last eigenvalues have all a less than

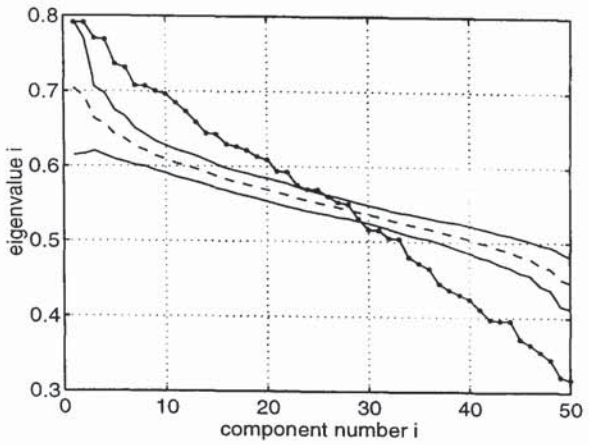

Figure 5.2: Original eigenspectrum for set 1 (left) and set 2 (right) of the SP500 data (dotted) compared to the statistics for 100 runs on randomly flipped returns indicated by the mean and two standard deviations (thin)

expected size.

In contrast, for the returns the deviation from the expected spectrum are very clear. This indicates the presence of some deterministic structure in the data. For set 2 even a structural break is evident at component number 9 indicating the begin of the noise floor.

One explanation could be the phenomenon of persistent volatility in financial returns. In order to test this the eigenspectrum was calculated for 100 samples of randomly flipped returns for both datasets. In Figure 5.2 it can be observed that there are still significant although slightly smaller differences between the spectra for the original data and the expected one for a time series. Furthermore, the error bar on the estimate for the 100 Bootstrap runs is slightly larger that for the scrambled version. This could indeed be interpreted as an indication for the presence of volatility persistence.

In Figure 5.3 the nine most important principal components for the prices and returns are shown. It can be seen that the components for the prices are represented by sine functions with increasing frequency. This is easily verified via a spectral decomposition. However, similar results were obtained for other financial as well as random time series. This means that the principal components of embedded prices account first of all for the Random walk structure in the prices.

In contrast, principal components for the returns seem to be less well behaved. The eigenvectors differ quite significantly for the two datasets. However, this could be simply due to a different sorting order.

In order to test this hypothesis further experiments were performed in which local eigenvalues were computed for a fixed set of eigenvectors. A local eigenvalue is here defined as the moving variance of the projection onto the corresponding principal component.

$$\lambda_t^{(m)} = \mathrm{Var}(\{\boldsymbol{U}'\boldsymbol{x}_i\}_{i=t-m}^{m}) \tag{5.8}$$

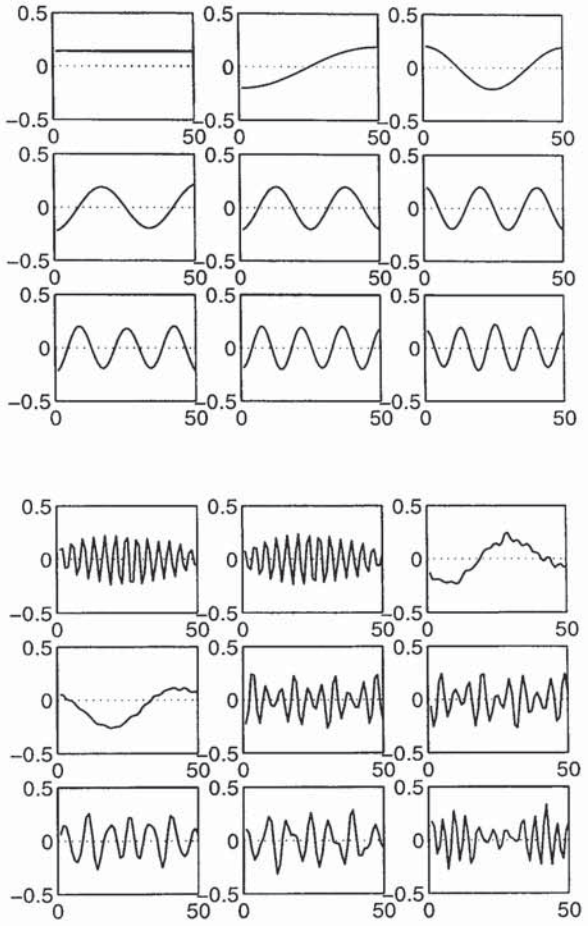

Figure 5.3: The nine most important principal components (eigenvectors) for the SP500 prices (top) and the returns (bottom), to be read from left to right and top to bottom for the set 1 (left) and set 2 (right)

Figure 5.4 shows the results for a segment of 4000 points of the first SP500 dataset. For the prices the top six local eigenvalues can clearly be separated. They move slowly and it turns out the order of the first three eigenvectors is hardly ever changing. In contrast, the ranking of components of higher order can change from time to time. Furthermore, the relative importance compared to each other is changing significantly as well. For example. at one time the first component accounts for almost all variance (around point 250) while at other times many more components are needed (around point 1600).

For the returns just the first, third and fifth local eigenvalue are shown for visual clarity since the remaining are strongly correlated to those displayed. For them it is characteristic to change more abruptly. Furthermore, no clear dominance of one eigenvector over all others can be observed.

5.4 Independent component analysis

Since PCA is linear and based entirely on second-order statistics the question arises naturally, if the problem at hand is linear or nonlinear. In the latter case higher-order statistics should be able to achieve better results. One extension to PCA pursued here is therefore independent component analysis.

In ICA the equivalence to the eigenmatrix $\boldsymbol{U}$ in PCA is a mixing matrix $\boldsymbol{A}$ with its columns as independent components. Its inverse $\boldsymbol{W}$ separates linearly the embedding matrix $\boldsymbol{X}$ into *statistically* independent sources $\boldsymbol{S} = \boldsymbol{W}\boldsymbol{X}$. In contrast to PCA the demixing matrix diagonalises not only the covariance matrix but also higher-order cumulant tensors.

For the estimation of the demixing matrix $\boldsymbol{W}$ several algorithms have been proposed. Here we use the FastICA[1] approach (Hyvärinen and Oja, 1997) for which the following assumptions are made: The sources are statistically independent, there is at most one source with a Gaussian distribution and the signals are stationary. Furthermore, it is assumed that there are as many signals as sources and that the mixing occurs instantaneously.

However, one problem remains for ICA: the ranking of the independent components and sources. With PCA, a ranking is defined according to the size of the eigenvalues. In ICA, those 'eigenvalues' are normalised to one. Therefore, Cardoso and Souloumiac (1993) suggested to order the columns in the mixing matrix $\boldsymbol{A}$, the ICs, according to their Euclidean norm L_2, since the sources with the most energy appear then first in the source matrix $\boldsymbol{S}$. Using the L_2 norm on the ICs is one reliable approach since this ranking is equivalent to a sorting according to the reconstruction performance of each single independent source $\boldsymbol{S}_i$ and this corresponds as well to the ranking of the principal sources and components.

Figure 5.5 shows this norm of the independent components respectively sources for the prices and returns of the SP500 data. It can be observed that for the prices one component

[1]The software for the FastICA is available at `http://www.cis.hut.fi/projects/ica/fastica/`.

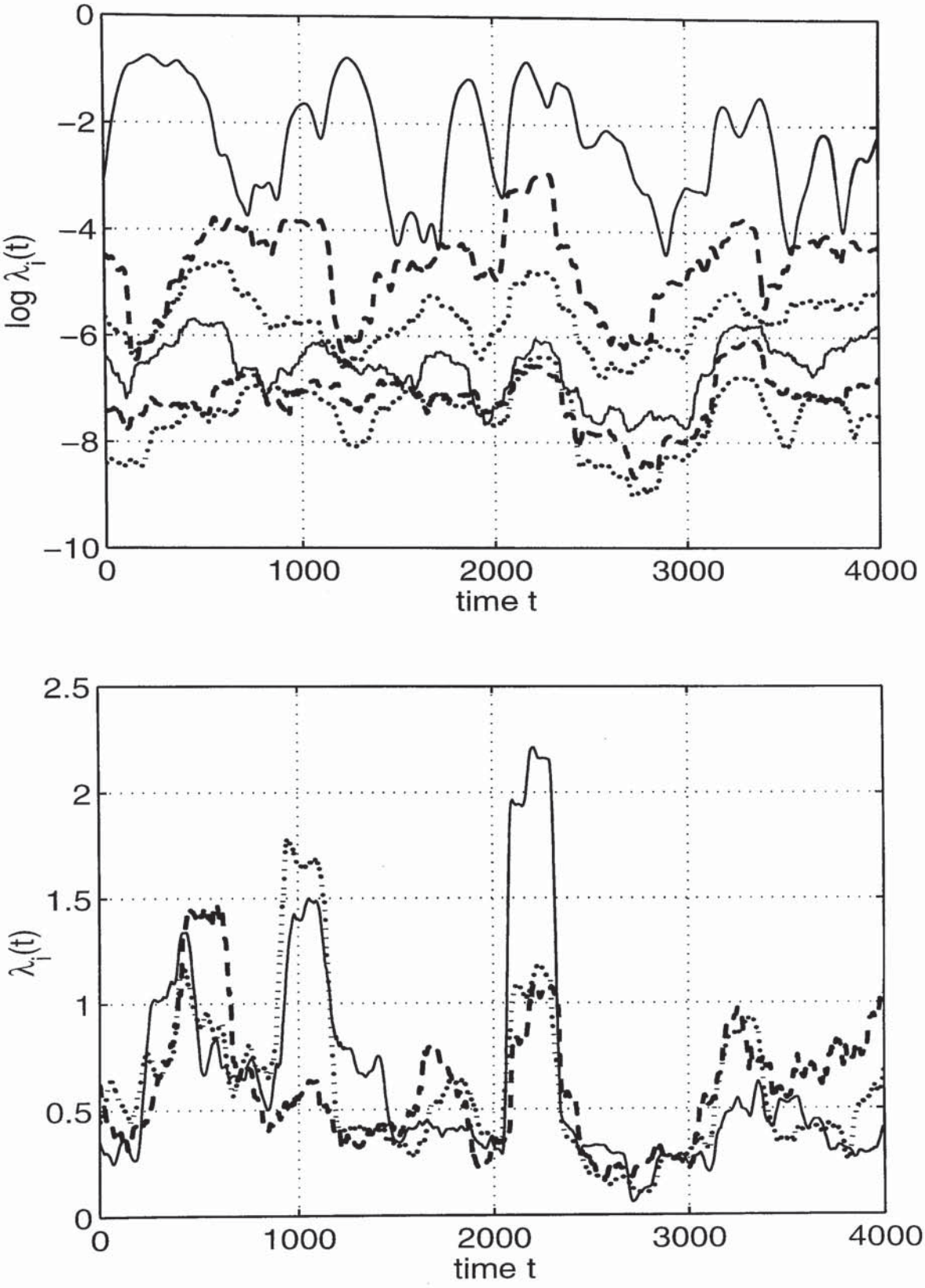

Figure 5.4: The top six moving eigenvalues for the prices (top) and the first, third and fifth moving eigenvalues for returns (bottom) of the last 4000 points of set 1 of the SP500 data

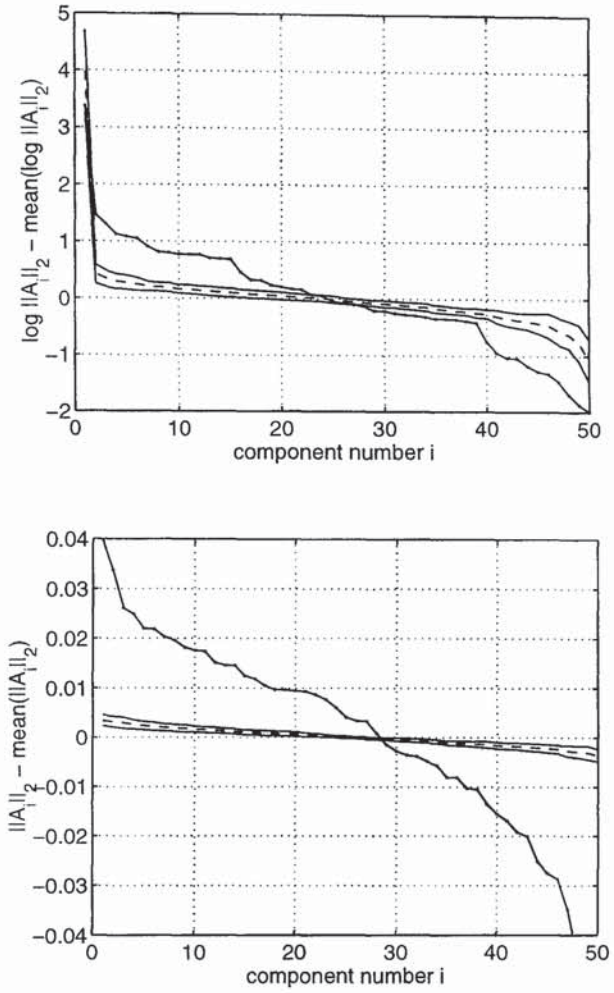

Figure 5.5: The mean normalised L_2 norm of the independent components $\boldsymbol{A}_i$ for sp500 prices (top) and returns (bottom) for set 1 (left) and set 2 (right) compared to the mean and one standard deviation for 10 runs of randomised data (sampling with replacement)

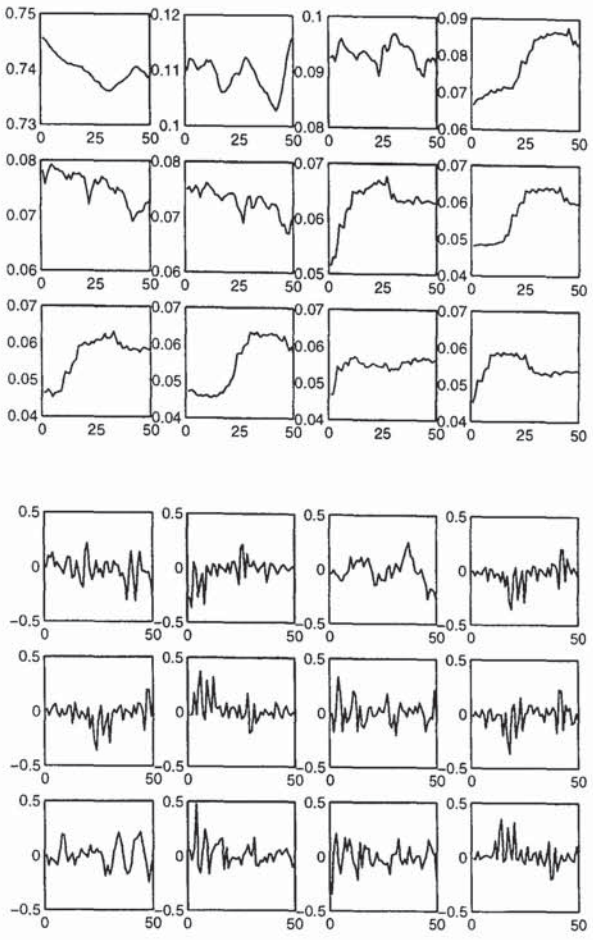

Figure 5.6: The twelve most important independent components $\boldsymbol{A}_i$ for SP500 prices (top) and the returns (bottom) from left to right and top to bottom for set 1 (left) and set 2 (right)

accounts for most of the information like in the PCA, and that the importance of the following ICs declines smoothly. For the returns the result is a partially discontinuous decline in the first third of the spectrum, like in PCA.

According to the chosen ranking, the twelve most important independent components for the prices and returns are presented in Figure 5.6. Here a completely different behaviour compared to the principal components can be observed: For the prices a straight-forward representation of the independent components as sine functions is not possible. Furthermore, for both prices and returns the ICs seem to be similar to each other, either due to having just the opposite sign or being shifted by a certain lag. That could be the result of using delayed vectors in the embedding process.

As last we want to show in Figure 5.7 the reconstruction quality of ICA by mixing just the most important components and comparing them with the original data. For the SP500 prices in dataset 2 just the first component was chosen since it had the most importance and dominated clearly all others. For the returns the first 13 components were selected according to the norm of the mixing vectors (cp. Figure 5.5). It can be seen that the reconstruction is

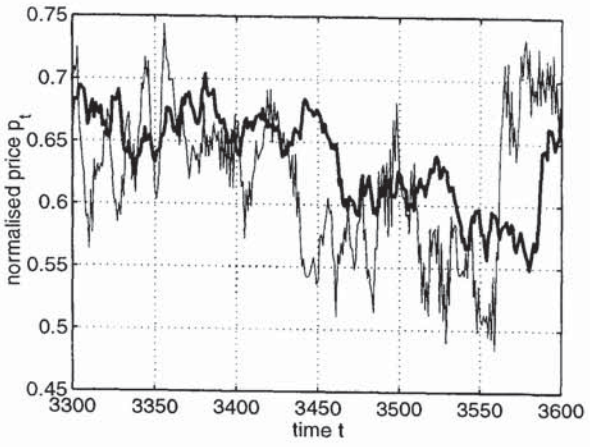

Figure 5.7: The reconstruction results for the SP500 prices (left) and returns (right). The original data (thick) is superimposed by the corresponding reconstracted time series (thin). For the prices just the most important component was taken while for the returns the first 13 components were used for the reconstruction.

quite poor for the prices and the returns. This also does not change significantly when using a few more components. The noisy behaviour is characteristic for the reconstruction. An expected denoising ability for the ICA can not be found. In contrast, the returns and prices seem to be even more noisy than the original data.

5.5 Discussion

This chapter has demonstrated the application of principal and independent component analysis as examples of factor models for single-channel financial time series. The motivation was to allow independent hidden factors without any dynamics to produce correlated observations via a linear and nonlinear transformation without any additional noise.

Since the analysis was performed on univariate data only the embedding approach with delayed coordinate vectors was used. As a null hypothesis randomised data were created via sampling with replacement. Both methods achieved significantly different results in terms of the eigenspectra for the original and randomised data. However, for a different version of random data, where the magnitude is kept intact and just the sign of the returns are randomly flipped the same results have been obtained. This means the univariate application of principal and independent component analysis extracts structure which represents the dependency of the current return on the magnitude of the previous one. In other words, it accounts for the persistence of the volatility in financial returns.

Furthermore, we have shown that the importance of the principal and independent components can change over time. Importance is here defined as the size of the eigenvalue, resulting in the eigenspectrum, and the Euclidian norm of the independent component. Concerning the form of the estimated components, the results are quite different. While the principal

components can be represented as orthogonal sine functions, the independent components are much closer in their morphology to the signals but do not have an obvious analytical representation.

Furthermore, the experiments have revealed evidence of clustering in the independent components. This could lead to a dimensionality reduction by deriving prototypical components of the correlated groups achieving a sparse representation of the signals. A further step for both methods would be also to use temporal information beside the spatial one in order to avoid the shift effect in components respectively source. Allowing such a temporal structure in the model and additionally observational noise leads to the next chapter which introduces state space models. These models can be understood as a factor model with a hidden dynamics.

Chapter 6

State space models

The main objective of this chapter is to propose a nonlinear state space model and to test if the non-Gaussian filter and smoother techniques involved, such as the particle filter, are able to perform better than the linear Kalman filter on financial time series. The general idea for using state space models is to allow for nonstationarity by tracking the distribution of financial returns conditional on previous data and to learn the evolution of this distribution from one time step to the next. Since the previous chapters have shown nonlinearity and non-Gaussianity for financial returns the nonlinear model discussed should achieve an improved predictive distribution which can then be used in the context of investment decision problems instead of a representation of the distribution by only its mean and variance.

The structure of this chapter is as follows. It will briefly give the foundations of linear and nonlinear state space models and discuss possibilities of performing inference and learning simultaneously. Thereby we will introduce an extension of the particle filter, the Hybrid particle filter and apply the same methodology for smoothing purposes, too. Furthermore, we will describe the representation of the nonlinear model equations by RBF networks and discuss a maximum-likelihood learning scheme for all model parameters. Finally, we apply these concepts to an artificial time series and discuss the corresponding problems.

6.1 Introduction

In time series analysis one common aim is to obtain a forecast conditional on previous data. Naturally, one question arises about how many past values are sufficient in order to capture all the information necessary for an accurate prediction. Here the concept of a hidden variable or factor (formally introduced in Chapter 5) might be helpful. Such a variable of arbitrary dimension is introduced to represent all predictive information. Thus the original time series can be seen as a sequence of observations depending fully on the sequence of the hidden variable. This circumvents the problem of long-range dependencies in the observations and allows separate modelling of the dynamics and noise structure in the hidden variable and of

the observation process.

As one example, such hidden variables might explain the phenomenon of clustered volatility in financial time series. There, almost no significant correlation exist between consecutively observed returns, the relative price changes, but strong correlation persists for their amplitude. So it can be imagined that there exist an underlying volatility process, in the form of a time-varying variance. This process creates then random samples observed as the returns in financial times series (Timmer and Weigend, 1997; Engle, 1995).

If the hidden variables are assumed to be discrete then this model family is equivalent to the family of *(discrete) hidden Markov models*. The underlying dynamics can then be modeled via a transition matrix which represents the conditional probabilities to obtain a certain state next given the current state. Since we are here interested in continuous variables the conditional probabilities cannot be represented by such a transition matrix. Instead a nonlinear function is used which is defined on the range of the hidden state and is modeled by a radial basis function network. This continuous hidden Markov model is also known as state space model.

The goal of this chapter is to model the evolving distribution of financial prices and returns using hidden variables within this state space model framework. The predictive distribution for the observed returns can then be used to optimise trading strategies in terms of, for example, risk minimisation. One fundamental representative of this model class for tracking and forecasting densities, the Kalman filter, assumes Gaussian distributions at each time step and linear underlying and observation processes. With that assumption the evolution of the distributions can be easily derived.

However, in Chapter 4 we have shown some evidence that financial returns follow non-Gaussian distributions and, furthermore, have confirmed mildly nonlinear relationships. Naturally, under these conditions the Kalman filter assumptions do not hold anymore and the tracking and learning becomes more difficult. Therefore we extend the linear framework by allowing a nonlinearity in the underlying dynamics as well as in the observation process.

The remaining sections of this chapter are organised as follows. First, the model is introduced with its underlying assumptions. Then it will be discussed how to infer the hidden state sequence and how to estimate the model parameters. For this task one learning approach, the expectation-maximisation algorithm, will be outlined. After a short summary of the linear model, the nonlinear version of state space models as well as corresponding strategies for inference and learning will be discussed in detail. Finally, experiments and results will be reported and discussed on artificial data.

6.2 State space models

A state space model represents a system characterised by an underlying but unobservable and thus hidden state $\boldsymbol{x}_t \in \mathbb{R}^n$ and an observation $\boldsymbol{y}_t \in \mathbb{R}^m$ attributed to the system at a discrete time t. The current state $\boldsymbol{x}_t$ is assumed to be a first-order Markov process, depending only on the last state $\boldsymbol{x}_{t-1}$, an available input $\boldsymbol{u}_t \in \mathbb{R}^r$ and an additive i.i.d. *system* or *dynamical noise* $\boldsymbol{w}_t \in \mathbb{R}^n$ drawn from a zero-mean Gaussian distribution with an $n \times n$ covariance matrix $\boldsymbol{Q}$. This dependency can therefore be modeled by the *system transition function* $\boldsymbol{f}_t : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^n$ as

$$\boldsymbol{x}_t = \boldsymbol{f}_t(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t) + \boldsymbol{w}_t \tag{6.1}$$

and writing the state dependency in a probabilistic way the *system transition density* can be obtained as a Gaussian with a nonlinear mean:

$$p(\boldsymbol{x}_t \,|\, \boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{f}_t(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t), \boldsymbol{Q}). \tag{6.2}$$

Since the state cannot be observed, the only available information about the system and its underlying process is the observation $\boldsymbol{y}_t$ obtained by the *observation function* $\boldsymbol{g}_t : \mathbb{R}^n \to \mathbb{R}^m$ relating it to the hidden state $\boldsymbol{x}_t$ via

$$\boldsymbol{y}_t = \boldsymbol{g}_t(\boldsymbol{x}_t) + \boldsymbol{v}_t, \tag{6.3}$$

with additive[1] i.i.d. zero-mean Gaussian *observation noise* $\boldsymbol{v}_t \in \mathbb{R}^m$ parametrised by an $m \times m$ covariance matrix $\boldsymbol{R}$. In analogy to the system transition density, this can be represented probabilistically as the *observation density*

$$p(\boldsymbol{y}_t \,|\, \boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{g}_t(\boldsymbol{x}_t), \boldsymbol{R}) \tag{6.4}$$

The advantage of such state space models becomes clear: instead of modelling the dependency of the current observation $\boldsymbol{y}_t$ on all previous observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{t-1}$ directly, the current observation depends just on the current hidden state $\boldsymbol{x}_t$, itself following a first-order Markov process.

Beside the system transition and observation density, it is necessary to specify the initial state density $p(\boldsymbol{x}_1)$, modelled for simplicity as a Gaussian with a mean $\boldsymbol{\pi} \in \mathbb{R}^n$, and an $n \times n$ covariance matrix $\boldsymbol{V}$:

$$p(\boldsymbol{x}_1) = \mathcal{N}(\boldsymbol{\pi}, \boldsymbol{V}). \tag{6.5}$$

Finally, the form of the system and observation function $\boldsymbol{f}_t$ and $\boldsymbol{g}_t$ has to be specified. In the linear case these functions are represented by matrices. In the nonlinear context here,

[1] Using additive Gaussian noise for the system and observation equation is less restrictive than in the linear case, since non-Gaussian distributions can be emulated to a certain degree by the allowed nonlinearities in the state and observation process (Ghahramani and Roweis, 1999).

we model them by radial basis function (RBF) networks (Broomhead and Lowe, 1988)[2]. Such RBF networks have been used by Roweis and Ghahramani (1999) in the context of the Extended Kalman filter. Furthermore, we assume the functions to be time-invariant and drop therefore the time index t from now on.

In the following, two sections will consider separately the two basic tasks for state space models: inference and learning. After that we will discuss one specific algorithm to actually perform these tasks.

6.3 Inference

Inference in the context of state space models means to discover the sequence of hidden states given the observations and the model parameters for either *descriptive* or *predictive* time series analysis. In a descriptive sense this hidden state sequence is used to explain the corresponding observations. For example, in radar tracking of an airplane only noisy measurements can be observed from a set of sensors about the position of the plane. The aim is here then to infer its true position represented by the hidden states.

In other cases, where the meaning of the hidden states is not clear in advance, a lower-dimensional representation of the observations by these states might aid an interpretation and explanation. In speech recognition, for instance, where a compact description of the observations given in form of time-varying frequency spectra is required, the hidden states have been found to represent phonemes forming words on a higher level of inference (Rabiner, 1989).

Applying this concept in a financial context, all stock prices in a particular market can be thought as noisy observations about the hidden market 'state'. It can be speculated that such a state could reflect *e.g.*, macro-economic fundamentals and important political events. Thus, once the hidden states have been inferred from the observations they can be further analysed in order to find support for various market hypotheses. This raises another issue which will be discussed comprehensively in the next section: when there is no model available in advance then it has to be estimated along the hidden states.

Another reason for using state space models is the ability to predict future observations based on estimated current hidden states. One example is the anticipation of the trajectories of two aircrafts close to each other in the sky in order to avoid their collision. A financial example is the forecast for a single stock based on the current market state.

After motivating the inference for state space model we are going now to outline how to achieve its two objectives. First we discuss how to *predict* future observations and then focus on the *retrieval* of the hidden states.

[2]Another network architecture, multilayer perceptrons (MLP), has been employed by (Briegel and Tresp, 1998) for nonlinear state space models.

Since the observations might follow a non-Gaussian distribution, a point prediction of the mean and variance is not entirely adequate. Therefore we need to consider the full *predictive* distribution $p(\boldsymbol{y}_{t+1} \mid \mathcal{Y}_t, \boldsymbol{\theta})$ given the time series $\mathcal{Y}_t = \{\boldsymbol{y}_\tau\}_{\tau=1}^t$ of all t previous observations *and* the model parameters $\boldsymbol{\theta}$. Those parameters remain fixed during the inference, therefore we drop $\boldsymbol{\theta}$ as a conditional term in the distributions throughout the remainder of this section.

The predictive distribution $p(\boldsymbol{y}_{t+1} \mid \mathcal{Y}_t)$ is the product of the *predictive state* distribution $p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)$ and the observation density defined in Equation (6.4) integrated over the hidden states $\boldsymbol{x}_{t+1}$:

$$p(\boldsymbol{y}_{t+1} \mid \mathcal{Y}_t) = \int p(\boldsymbol{y}_{t+1} \mid \boldsymbol{x}_{t+1})\, p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)\, d\boldsymbol{x}_{t+1} \tag{6.6}$$

The predictive state distribution in turn is obtained by propagating forward the current *posterior state* distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_t)$ via the system density according to Equation (6.2):

$$p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t) = \int p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_t \mid \mathcal{Y}_t)\, d\boldsymbol{x}_t \tag{6.7}$$

and the posterior state distribution is finally computed using Bayes' theorem

$$p(\boldsymbol{x}_t \mid \mathcal{Y}_t) = \frac{p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_t \mid \mathcal{Y}_{t-1})}{p(\boldsymbol{y}_t \mid \mathcal{Y}_{t-1})}. \tag{6.8}$$

Note that the evidence $p(\boldsymbol{y}_t \mid \mathcal{Y}_{t-1})$ is here the prediction made for the current observation at the previous time step $t-1$. This suggests the following two-step iterative procedure for retrieving the hidden state and predicting the observation: Given a posterior estimate for the current state the next predictive state distribution is calculated in the *prediction* step via the system density. In the following *update* step, the now available observation is taken into account to correct this prediction via Bayes' theorem resulting in the posterior distribution. The procedure starts with the initial state density $p(\boldsymbol{x}_1)$ given in Equation (6.5) as the predictive distribution $p(\boldsymbol{x}_1 \mid \mathcal{Y}_0)$ and iterated through the whole observation sequence.

According to which observations $\mathcal{Y}_\tau$ are used to retrieve the hidden state $\boldsymbol{x}_t$ at time t inference is distinguished into *prediction*, based only on previous observations ($\tau < t$) and *filtering*, where previous and current observations are used ($\tau \leq t$). The process which takes also future values into account ($\tau \leq T$) is called *smoothing* and its objective is to obtain the conditional distribution

$$p(\boldsymbol{x}_t \mid \mathcal{Y}_T) = p(\boldsymbol{x}_t \mid \mathcal{Y}_t) \int \frac{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_T)}{p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)}\, d\boldsymbol{x}_{t+1} \tag{6.9}$$

given all observations $\mathcal{Y}_T$. Naturally, this smoothing distribution cannot be utilised for predictions, since future observations are used for its estimation. However, smoothing allows us to obtain less ambiguous and less noisy estimates for the hidden states which facilitates their interpretation. Furthermore, in case the model parameters are not known in advance, the learning of the underlying dynamics becomes easier and more robust. This issue will be discussed in detail in the next section.

Finally, it needs to be emphasised that the integrals for prediction, filtering and smoothing are analytically solvable only for a small class of parametric distributions, such as the Gaussian, for instance. The linear Kalman filter and smoother will therefore be analysed briefly in section 6.5. In the case of arbitrarily smooth nonlinearities, the system and observation equations can be linearised locally via Taylor expansion. The distributions are then approximated by Gaussians and the linear Kalman filter equations can be applied, leading to the Extended Kalman filter (Anderson and Moore, 1979) which will not be considered here further.

A more general approach to deal with nonlinearity in the state transition and observation function is the particle filter we will examine here in detail. Such a filter uses samples from the distribution instead of its parametric form. In section 6.6 we discuss strategies to obtain and use these samples to approximate the distributions of interest.

6.4 Learning

In many real-world time series problems there is only limited or even no knowledge available about the underlying model of the time series. In such cases the model parameters have to be estimated along with the hidden state sequence based on the available observations only. We will therefore discuss in this section the Bayesian treatment of this problem first and then derive an approach for inferring the most probable model parameters.

During inference one computes the posterior distribution $p(\mathcal{X}_T \mid \mathcal{Y}_T, \boldsymbol{\theta})$ of the hidden state given the observations *and* the model parameters $\boldsymbol{\theta}$. Now the hidden states and parameters need to be estimated simultaneously. Therefore, we are interested in the joint posterior distribution of the hidden states and model parameters given the observations, which can be obtained via Bayes' theorem as

$$p(\mathcal{X}_T, \boldsymbol{\theta} \mid \mathcal{Y}_T) = \frac{p(\mathcal{X}_T \mid \mathcal{Y}_T, \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid \mathcal{Y}_T)}{p(\mathcal{Y}_T)} \tag{6.10}$$

with the posterior parameter distribution $p(\boldsymbol{\theta} \mid \mathcal{Y}_T)$ representing the uncertainty in their estimate.

Here we adopt the *maximum likelihood* (ML) approach for estimating the model parameters $\boldsymbol{\theta}$ and write the likelihood $\mathcal{L}$ as a function of those parameters

$$\mathcal{L}(\boldsymbol{\theta}) = p(\mathcal{Y}_T \mid \boldsymbol{\theta}) = \int p(\mathcal{X}_T, \mathcal{Y}_T \mid \boldsymbol{\theta})\, d\mathcal{X}_T. \tag{6.11}$$

Now the joint distribution $p(\mathcal{X}_T, \mathcal{Y}_T \mid \boldsymbol{\theta})$ of all states and observations can be factorised under the Markov assumption into a product of system and observation density terms:

$$p(\mathcal{X}_T, \mathcal{Y}_T \mid \boldsymbol{\theta}) = p(\boldsymbol{x}_1) \prod_{t=2}^{T} p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) \prod_{t=1}^{T} p(\boldsymbol{y}_t \mid \boldsymbol{x}_t). \tag{6.12}$$

However, as already mentioned earlier, instead of maximising $\mathcal{L}(\boldsymbol{\theta})$ directly, it is useful to maximise the *total log likelihood* of the observations:

$$\log \mathcal{L}(\boldsymbol{\theta}) = \int p(\mathcal{X}_T \,|\, \mathcal{Y}_T, \boldsymbol{\theta}) \, \log \frac{p(\mathcal{X}_T, \mathcal{Y}_T \,|\, \boldsymbol{\theta})}{p(\mathcal{X}_T \,|\, \mathcal{Y}_T, \boldsymbol{\theta})} \, d\mathcal{X}_T. \tag{6.13}$$

In order to maximise this log-likelihood function with respect to the model parameters, several gradient-based optimisation algorithms have been proposed as mentioned in Section 2.2. Here we use exclusively the expectation-maximisation (EM) algorithm for the combined inference and learning problem (Dempster *et al.*, 1977). Shumway and Stoffer (1982) proposed the EM algorithm to estimate the hidden state distributions and to learn the model parameters in an elegant and simple way.

In contrast to nonlinear optimisation methods which maximise the likelihood function directly, the EM algorithm divides the problem into two parts. Using the abbreviation $q(\mathcal{X}_T) \equiv p(\mathcal{X}_T \,|\, \mathcal{Y}_T, \boldsymbol{\theta})$ for the posterior state distribution, the log-likelihood in Equation (6.13) can be re-written in the following way:

$$\log \mathcal{L}(\boldsymbol{\theta}) = \int q(\mathcal{X}_T) \, \log p(\mathcal{X}_T, \mathcal{Y}_T \,|\, \boldsymbol{\theta}) \, d\mathcal{X}_T - \int q(\mathcal{X}_T) \, \log q(\mathcal{X}_T) \, d\mathcal{X}_T \tag{6.14}$$

$$= \mathcal{F}(q(\mathcal{X}_T), \boldsymbol{\theta}). \tag{6.15}$$

This decomposition allows a two-step iterative maximisation procedure which will be repeated until convergence. In each step the log-likelihood is maximised with respect to either the distribution $q(\mathcal{X}_T)$ or the model parameters $\boldsymbol{\theta}$ while the other quantity remains fixed:

$$\begin{aligned} \text{E step:} \quad & q_{i+1} \Leftarrow \arg\max_{q} \mathcal{F}(q, \boldsymbol{\theta}_i) \\ \text{M step:} \quad & \boldsymbol{\theta}_{i+1} \Leftarrow \arg\max_{\boldsymbol{\theta}} \mathcal{F}(q_{i+1}, \boldsymbol{\theta}) \end{aligned}$$

At first, inference is performed in the *expectation step*: the state distribution $q(\mathcal{X}_T)$ is estimated conditional on all observations and current model parameters. Then learning takes place in the *maximisation step*: the model parameters $\boldsymbol{\theta}$ are trained using the new estimate q_{i+1} for the distribution $q(\mathcal{X}_T)$. This step can be simplified since the second term in Equation (6.14), the entropy of the state distribution is fixed at this point. Therefore only the first term, the *expected log likelihood*

$$\mathcal{Q}(\boldsymbol{\theta}) = \int p(\mathcal{X}_T \,|\, \mathcal{Y}_T, \boldsymbol{\theta}) \, \log p(\mathcal{X}_T, \mathcal{Y}_T \,|\, \boldsymbol{\theta}) \, d\mathcal{X}_T \tag{6.16}$$

needs to be maximised. This is done in the usual way by setting its derivative with respect to the model parameters to zero and then solving these equations to get the new parameter estimates.

In contrast to the direct optimisation techniques mentioned above, the EM algorithm always finds a mode of the likelihood function $\mathcal{L}(\boldsymbol{\theta})$. It is furthermore guaranteed to increase or at least to stay flat in every iteration. An additional feature is its simplicity in deriving the

equations for the expectation and maximisation step. Therefore EM is used in cases where the likelihood function is difficult to maximise with respect to the model parameters directly. Nevertheless, since second-order derivatives of the error function are not calculated, the EM algorithm does not provide error bars and suffers from slow convergence towards the end of the learning.

One final note has to be made: (Roweis and Ghahramani, 1999) showed that the covariance matrix $\boldsymbol{Q}$ for the system noise can be set to the unity matrix since the scale of the noise can be shifted to the system transition function. This reduces the number of equivalent solutions. Another ambiguity arises from the ordering of the components in the hidden state vector. Therefore an arbitrary ranking scheme could be employed. For the linear case an ordering was suggested based on the norm of the columns in the observation matrix $\boldsymbol{G}$. We will come back to those technicalities when we consider the learning of the model.

Depending on the functional form of the system and observation densities we will get different classes of solutions which we are going to describe in the following. For Gaussian densities a solution can be obtained via the Kalman filter and smoother in combination with the maximisation of the likelihood. That will be summarised next. After that we will outline an extension for the nonlinear, non-Gaussian case.

6.5 The linear case

In the linear Gaussian state space model the system transition and observation function $\boldsymbol{f}$ and $\boldsymbol{g}$ in Equation (6.1) and (6.3) are constrained to be linear transformations represented by an $n \times n$ system transition matrix $\boldsymbol{F}$, an $n \times r$ input transformation matrix $\boldsymbol{H}$ and an $m \times n$ output matrix $\boldsymbol{G}$ which results in

$$\boldsymbol{x}_t = \boldsymbol{F}\boldsymbol{x}_{t-1} + \boldsymbol{H}\boldsymbol{u}_t + \boldsymbol{w}_t \tag{6.17}$$

$$\boldsymbol{y}_t = \boldsymbol{G}\boldsymbol{x}_t + \boldsymbol{v}_t. \tag{6.18}$$

Based on these equations the conditional state and observation densities are given by Gaussians linear in their mean:

$$p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) = (2\pi)^{-n/2} |\boldsymbol{Q}|^{-1/2} e^{-\frac{1}{2}[\boldsymbol{x}_t - \boldsymbol{F}\boldsymbol{x}_{t-1} - \boldsymbol{H}\boldsymbol{u}_t]' \boldsymbol{Q}^{-1} [\boldsymbol{x}_t - \boldsymbol{F}\boldsymbol{x}_{t-1} - \boldsymbol{H}\boldsymbol{u}_t]} \tag{6.19}$$

$$p(\boldsymbol{y}_t \mid \boldsymbol{x}_t) = (2\pi)^{-m/2} |\boldsymbol{R}|^{-1/2} e^{-\frac{1}{2}[\boldsymbol{y}_t - \boldsymbol{G}\boldsymbol{x}_t]' \boldsymbol{R}^{-1} [\boldsymbol{y}_t - \boldsymbol{G}\boldsymbol{x}_t]} \tag{6.20}$$

$$p(\boldsymbol{x}_1) = (2\pi)^{-n/2} |\boldsymbol{V}|^{-1/2} e^{-\frac{1}{2}[\boldsymbol{x}_1 - \boldsymbol{\pi}]' \boldsymbol{V}^{-1} [\boldsymbol{x}_1 - \boldsymbol{\pi}]}. \tag{6.21}$$

For the E step of the EM algorithm the expected log-likelihood $\mathcal{Q}(\boldsymbol{\theta}) \equiv \mathbb{E}[\log p(\mathcal{X}_T, \mathcal{Y}_T \mid \boldsymbol{\theta}) \mid \mathcal{Y}_T]$ defined in Equation (6.16) has to be computed. For notational convenience we denote with $\boldsymbol{x}_{t|\tau}$ and $\boldsymbol{V}_{t|\tau}$ the expectations for the mean and the covariance of the hidden state $\boldsymbol{x}_t$ given

all observations up to time τ:

$$\boldsymbol{x}_{t|\tau} \equiv \mathbb{E}[\boldsymbol{x}_t \mid \mathcal{Y}_\tau] \tag{6.22}$$

$$\boldsymbol{V}_{t|\tau} \equiv \mathbb{E}\left[(\boldsymbol{x}_t - \boldsymbol{x}_{t|\tau})(\boldsymbol{x}_t - \boldsymbol{x}_{t|\tau})' \mid \mathcal{Y}_\tau\right]. \tag{6.23}$$

Next we will discuss how these expectations are computed in the E step. After that, the M step will be considered, which estimates the model parameter with a maximum likelihood approach.

6.5.1 The Kalman filter and smoother

With a Gaussian state and observation distribution defined in Equation (6.19) and (6.20), filtering necessarily results in Gaussians for the predictive distribution $p(\boldsymbol{y}_t \mid \mathcal{Y}_{t-1})$ of the observations, as well as for the predictive state distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_{t-1})$ and the posterior distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_t)$, with corresponding means $\boldsymbol{y}_{t|t-1}$, $\boldsymbol{x}_{t|t-1}$ and $\boldsymbol{x}_{t|t}$ and covariance matrices $\boldsymbol{\Sigma}_{t|t-1}$, $\boldsymbol{V}_{t|t-1}$ and $\boldsymbol{V}_{t|t}$, respectively:

$$p(\boldsymbol{y}_t \mid \mathcal{Y}_{t-1}) = \mathcal{N}(\boldsymbol{y}_{t|t-1}, \boldsymbol{\Sigma}_{t|t-1}) \tag{6.24}$$

$$p(\boldsymbol{x}_t \mid \mathcal{Y}_{t-1}) = \mathcal{N}(\boldsymbol{x}_{t|t-1}, \boldsymbol{V}_{t|t-1}) \tag{6.25}$$

$$p(\boldsymbol{x}_t \mid \mathcal{Y}_t) = \mathcal{N}(\boldsymbol{x}_{t|t}, \boldsymbol{V}_{t|t}) \tag{6.26}$$

In order to estimate the mean and covariance parameters the following Kalman filter (forward) recursions are performed by analogy with the Equations (6.6), (6.7) and (6.8) with the predictive state distribution $p(\boldsymbol{x}_1 \mid \mathcal{Y}_0)$ given by the initial state density $p(\boldsymbol{x}_1)$ and therefore $\boldsymbol{x}_{1|0} = \boldsymbol{\pi}$ and $\boldsymbol{V}_{1|0} = \boldsymbol{V}$

$$\boldsymbol{x}_{t|t-1} = \boldsymbol{F}\boldsymbol{x}_{t-1|t-1} + \boldsymbol{H}\boldsymbol{u}_t \tag{6.27}$$

$$\boldsymbol{V}_{t|t-1} = \boldsymbol{F}\boldsymbol{V}_{t-1|t-1}\boldsymbol{F}' + \boldsymbol{Q} \tag{6.28}$$

$$\boldsymbol{y}_{t|t-1} = \boldsymbol{G}\boldsymbol{x}_{t|t-1} \tag{6.29}$$

$$\boldsymbol{\Sigma}_{t|t-1} = \boldsymbol{G}\boldsymbol{V}_{t|t-1}\boldsymbol{G}' + \boldsymbol{R} \tag{6.30}$$

$$\boldsymbol{K}_t = \boldsymbol{V}_{t|t-1}\boldsymbol{G}'\boldsymbol{\Sigma}_{t|t-1}^{-1} \tag{6.31}$$

$$\boldsymbol{x}_{t|t} = \boldsymbol{x}_{t|t-1} + \boldsymbol{K}_t(\boldsymbol{y}_t - \boldsymbol{y}_{t|t-1}) \tag{6.32}$$

$$\boldsymbol{V}_{t|t} = \boldsymbol{V}_{t|t-1} - \boldsymbol{K}_t\boldsymbol{G}\boldsymbol{V}_{t|t-1} \tag{6.33}$$

Now the smoothed expectations $\boldsymbol{x}_{t|T}, \boldsymbol{V}_{t|T}$ and $\boldsymbol{V}_{t,t-1|T}$ are computed for the mean, the spatial and temporal covariance respectively, via a set of Kalman smoother (backward) recursions in analogy to Equation (6.9):

$$\boldsymbol{J}_{t-1} = \boldsymbol{V}_{t-1|t-1}\boldsymbol{F}'\boldsymbol{V}_{t|t-1}^{-1} \tag{6.34}$$

$$\boldsymbol{x}_{t-1|T} = \boldsymbol{x}_{t-1|t-1} + \boldsymbol{J}_{t-1}(\boldsymbol{x}_{t|T} - \boldsymbol{F}\boldsymbol{x}_{t-1|t-1} - \boldsymbol{H}\boldsymbol{u}_t) \tag{6.35}$$

$$\boldsymbol{V}_{t-1|T} = \boldsymbol{V}_{t-1|t-1} + \boldsymbol{J}_{t-1}(\boldsymbol{V}_{t|T} - \boldsymbol{V}_{t|t-1})\boldsymbol{J}_{t-1}' \tag{6.36}$$

$$\boldsymbol{V}_{t-1,t-2|T} = \boldsymbol{V}_{t-1|t-1}\boldsymbol{J}_{t-2}' + \boldsymbol{J}_{t-1}(\boldsymbol{V}_{t,t-1|T} - \boldsymbol{F}\boldsymbol{V}_{t-1|t-1})\boldsymbol{J}_{t-2}' \tag{6.37}$$

with initialisations for $\boldsymbol{x}_{T|T}$ and $\boldsymbol{V}_{T|T}$ given by the posterior estimates in Equation (6.32) and (6.33) from the forward recursions and $\boldsymbol{V}_{T,T-1|T} = (\boldsymbol{I} - \boldsymbol{K}_T\boldsymbol{G})\boldsymbol{F}\boldsymbol{V}_{T-1|T-1}$.

6.5.2 Learning the linear model

In order to estimate the model parameters $\boldsymbol{\theta} \equiv (\boldsymbol{F}, \boldsymbol{G}, \boldsymbol{H}, \boldsymbol{R}, \boldsymbol{Q}, \boldsymbol{\pi}, \boldsymbol{V})$ the expected log-likelihood $\mathcal{Q}(\boldsymbol{\theta})$ is maximised with respect to $\boldsymbol{\theta}$. Here abbreviations are used for the mean $\hat{\boldsymbol{x}}_t \equiv \boldsymbol{x}_{t|T}$, the spatial covariance $\boldsymbol{P}_t \equiv \boldsymbol{V}_{t|T} + \boldsymbol{x}_{t|T}\boldsymbol{x}_{t|T}'$ and the temporal covariance $\boldsymbol{P}_{t,t-1} \equiv \boldsymbol{V}_{t,t-1|T} + \boldsymbol{x}_{t|T}\boldsymbol{x}_{t-1|T}'$ for an efficient notation. The new parameter estimates, denoted by *, are obtained via setting the corresponding derivative of $\mathcal{Q}$ to zero (Shumway and Stoffer, 1982; Ghahramani and Hinton, 1996) and solving the equation. This results in the following new estimates:

$$\boldsymbol{F}^* = \left(\sum_{t=2}^{T}\boldsymbol{P}_{t,t-1}\right)\left(\sum_{t=2}^{T}\boldsymbol{P}_{t-1}\right)^{-1} \tag{6.38}$$

$$\boldsymbol{G}^* = \left(\sum_{t=1}^{T}\boldsymbol{y}_t\hat{\boldsymbol{x}}_t'\right)\left(\sum_{t=1}^{T}\boldsymbol{P}_t\right)^{-1} \tag{6.39}$$

$$\boldsymbol{H}^* = \left(\sum_{t=2}^{T}(\hat{\boldsymbol{x}}_t - \boldsymbol{F}\hat{\boldsymbol{x}}_{t-1})\,\boldsymbol{u}_t'\right)\cdot\left(\sum_{t=2}^{T}\boldsymbol{u}_t\boldsymbol{u}_t'\right)^{-1} \tag{6.40}$$

$$\boldsymbol{R}^* = \frac{1}{T}\sum_{t=1}^{T}(\boldsymbol{y}_t\boldsymbol{y}_t' - \boldsymbol{G}^*\hat{\boldsymbol{x}}_t\boldsymbol{y}_t') \tag{6.41}$$

$$\boldsymbol{Q}^* = \frac{1}{T-1}\left(\sum_{t=2}^{T}\boldsymbol{P}_t - \boldsymbol{F}^*\sum_{t=2}^{T}\boldsymbol{P}_{t-1,t}\right) \tag{6.42}$$

$$\boldsymbol{\pi}^* = \hat{\boldsymbol{x}}_1 \tag{6.43}$$

$$\boldsymbol{V}^* = \boldsymbol{P}_1 - \hat{\boldsymbol{x}}_1\hat{\boldsymbol{x}}_1'. \tag{6.44}$$

Regarding the quality of the algorithm, the total log-likelihood can be computed completely during the filter (forward) pass in the 'innovations' form (Gupta and Mehra, 1974). This uses the evidence $p(\boldsymbol{y}_t \mid \mathcal{Y}_{t-1})$ in the Bayesian update equation of the state posterior. In the linear context the evidence is a Gaussian, defined in Equation 6.24, which gives a straight forward

expression for the negative log-likelihood E of the full dataset:

$$\begin{aligned} E &= -\log \mathcal{L} = -\log p(\mathcal{Y}_T \,|\, \boldsymbol{\theta}) = -\sum_{t=1}^{T} \log p(\boldsymbol{y}_t \,|\, \mathcal{Y}_{t-1}, \boldsymbol{\theta}) \\ &= -\sum_{t=1}^{T} \log \left\{ (2\pi)^{-m/2} \, |\boldsymbol{\Sigma}|^{-1/2} \, e^{-\frac{1}{2}(\boldsymbol{y}_t - \boldsymbol{y}_{t|t-1})' \boldsymbol{\Sigma}^{-1} (\boldsymbol{y}_t - \boldsymbol{y}_{t|t-1})} \right\} \\ &= \frac{mT}{2} \log(2\pi) + \frac{1}{2} \sum_{t=1}^{T} \Big\{ \log |\boldsymbol{\Sigma}_{t|t-1}| + (\boldsymbol{y}_t - \boldsymbol{y}_{t|t-1})' \boldsymbol{\Sigma}_{t|t-1}^{-1} (\boldsymbol{y}_t - \boldsymbol{y}_{t|t-1}) \Big\}. \end{aligned} \tag{6.45}$$

It rather reflects the quality of the predictions than of the smoothed values which is more useful for fair comparison with other methods.

The results using this linear paradigm will be discussed after the nonlinear approach has been introduced in order to allow a comparison.

6.6 The nonlinear case

The advantage of the linear Kalman filter for time series analysis lies in its analytical simplicity: all integrals can be computed by deriving the mean and the covariance of the corresponding Gaussian distributions. The drawback is its lack of applicability for nonlinear and non-Gaussian problems. The natural desire is therefore to allow more flexibility in the model in the form of nonlinear dynamics and non-normal predictive distributions. Unfortunately, in general this has the consequence that the solutions for the distributions involved cannot be computed analytically any more.

For the purpose of allowing non-Gaussian distributions we will therefore follow here the approach of representing the distributions of interest by an arbitrary number of samples (Gordon, 1996; Kitagawa, 1987; Pitt and Shephard, 1997). These samples are then used to perform inference. For obtaining such samples efficiently we propose a combined strategy of using a rejection sampler and Gaussian mixture models. This inference stage with its individual components for prediction, filtering and smoothing will be considered next.

In order to permit nonlinear system and observation equations we propose afterwards using radial basis function networks which can be trained efficiently with the EM algorithm in the learning stage of the model. Finally, there also implementation issues will be discussed, such as initialisation, the choice of the hidden state dimension and convergence.

6.6.1 The particle filter

In contrast to the analytical approach a distribution $p(\boldsymbol{x})$ is represented in the particle filter by a (generic) set $\mathcal{X} = \{\boldsymbol{x}_n\}_{n=1}^{N}$ of N samples $\boldsymbol{x}_n$ from its distribution. This representation is sufficient since an arbitrarily large number of such samples can be used by Monte Carlo methods to approximate any integral of interest (Appendix B).

This approach can therefore be used in the prediction step of the particle filter in the following way (Algorithm 1): The predictive state distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_{t-1})$, defined in Equation (6.7), can be estimated based on a set $\mathcal{X}_{t-1|t-1} = \{\boldsymbol{x}_{t-1|t-1}^{(n)}\}_{n=1}^{N}$ of *previous posterior* samples $\boldsymbol{x}_{t-1|t-1} \sim p(\boldsymbol{x}_{t-1} \mid \mathcal{Y}_{t-1})$. Applying Monte Carlo the predictive distribution can be approximated as

$$p(\boldsymbol{x}_t \mid \mathcal{Y}_{t-1}) = \int p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})\, p(\boldsymbol{x}_{t-1} \mid \mathcal{Y}_{t-1})\, d\boldsymbol{x}_{t-1} \simeq \frac{1}{N} \sum_{n=1}^{N} p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1} = \boldsymbol{x}_{t-1}^{(n)}). \tag{6.46}$$

This means that samples representing the predictive distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_{t-1})$ can be created by propagating i.i.d. samples from the previous posterior through the system via the system equation given in Equation (6.1). Therefore, N^* samples are drawn uniformly with replacement from $\mathcal{X}_{t-1|t-1}$ and mapped to a set $\mathcal{X}_{t|t-1}^{*} = \{\boldsymbol{x}_{t|t-1}^{(n)}\}_{n=1}^{N^*}$ of prediction samples $\boldsymbol{x}_{t|t-1}^{*} = f(\boldsymbol{x}_{t-1|t-1}^{*}, \boldsymbol{u}_t, \boldsymbol{w}_t^{*})$.

Algorithm 1 The particle filter: The set $\mathcal{X}_{t-1|t-1} \equiv \{\boldsymbol{x}_{t-1|t-1}^{(n)}\}_{n=1}^{N}$ of previous posterior samples $\boldsymbol{x}_{t-1|t-1} \sim p(\boldsymbol{x}_{t-1} \mid \mathcal{Y}_{t-1})$ is transformed into a set $\mathcal{X}_{t|t} \equiv \{\boldsymbol{x}_{t|t}^{(n)}\}_{n=1}^{N}$ of current posterior samples $\boldsymbol{x}_{t|t} \sim p(\boldsymbol{x}_t \mid \mathcal{Y}_t)$ using the current system input $\boldsymbol{u}_t$, the observation $\boldsymbol{y}_t$, and the state space model (f, g, $\boldsymbol{Q}$, $\boldsymbol{R}$)

% Forward propagation of posterior state samples
for $n = 1$ to N^* **do**
  $\boldsymbol{w}_n \sim \mathcal{N}(0, \boldsymbol{Q})$
  $\boldsymbol{x}_n \sim \mathcal{X}_{t-1|t-1}$
  $\boldsymbol{x}_n^{*} \Leftarrow f(\boldsymbol{x}_n, \boldsymbol{u}_t, \boldsymbol{w}_n)$
  $q_n \Leftarrow p(\boldsymbol{y}_t \mid \boldsymbol{x}_n^{*})$
end for
% Normalisation of weighting factors
for $n = 1$ to N^* **do**
  $\pi_n \Leftarrow q_n / \sum_{k=1}^{N^*} q_k$
end for
% Re-sampling of prediction samples weighted by the likelihood
for $n = 1$ to N **do**
  $\boldsymbol{x}_n \sim \{(\boldsymbol{x}_j^{*}, \pi_j)\}_{j=1}^{N^*}$ such that $P(\boldsymbol{x}_n = \boldsymbol{x}_j^{*}) = \pi_j$
end for
return $\mathcal{X}_{t|t} \equiv \{\boldsymbol{x}_n\}_{n=1}^{N}$

In the update step the prediction samples in $\mathcal{X}_{t|t-1}^{*}$ are used to create, via Bayes' theorem in Equation (6.8), the sample set $\mathcal{X}_{t|t} \equiv \{\boldsymbol{x}_{t|t}^{(n)}\}_{n=1}^{N}$ representing the current posterior distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_t)$ for the filtered estimates of the state $\boldsymbol{x}_t$.

For the implementation of this step it was suggested to use, for instance. the *sampling/importance re-sampling* (SIR) method (Gordon, 1996; Pitt and Shephard, 1997). Its idea is to draw a sample $\boldsymbol{x}_{t|t-1}^{(n)}$ uniformly with replacement from $\mathcal{X}_{t|t-1}$ and then to accept it with probability $\pi^{(n)} = q^{(n)} / \sum_{k=1}^{N^*} q^{(k)}$ using the likelihood $q^{(n)} \equiv p(\boldsymbol{y}_t \mid \boldsymbol{x}_{t|t-1}^{(n)})$. Further details including an algorithm are given in Appendix B.2. Similar to this is the method of *rejection sampling* (RS) discussed in Appendix B.3 with the difference of normalising π_n by its largest value over all n.

In the context of the nonlinear state space model we experienced a more time-efficient performance by the rejection sampler compared to SIR. This stems from the different normalisation with the consequence of a higher acceptance rate for the rejection sampler. However, both methods suffer when the likelihood is very different from the prior. In that case the proposed samples are too often rejected leading to a dramatic increase in computation time. This is demonstrated in Figure 6.1 for a nonlinear and bimodal prediction problem[3]. Note that by using an acceptance ratio as the likelihood normalised by the maximum likelihood over the entire sample set a huge improvement for the computation is achieved.

As an alternative to this direct sampling method we propose therefore a hybrid method which uses a Gaussian mixture model similar to an idea in (Gordon, 1996; Gordon, 1997). The key is to recognise that the desired posterior distribution is represented by the discrete distribution over the supporting points $\{\boldsymbol{x}_{t|t-1}^{(n)}\}_{n=1}^{N^*}$ (which are the predictive samples) with their associated weighting coefficients π_n as the probability mass (representing the normalised likelihood). Gordon (1996) suggests an algorithm which places Gaussian kernels at every data point and then merges successively close kernels together building up a mixture model. Unfortunately, this approach seems to be rather subjective in terms of merging strategies and is furthermore computationally expensive for a large sampling set. Therefore, we propose a direct approximation of a Gaussian mixture model from the weighted data samples.

Using this approach the sampling part becomes more efficient for strongly differing prior and posterior distributions. Additional computational effort is only necessary for the mixture approximation which is independent of differences in prior and likelihood. The posterior is therefore modelled in the usual way by a mixture of Gaussian kernels

$$\hat{p}(\boldsymbol{x}_t \mid \mathcal{Y}_t) = \sum_{j=1}^{M} \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\, P_j \tag{6.47}$$

with mean $\boldsymbol{\mu}_j$, covariance matrix $\boldsymbol{\Sigma}_j$ and prior probability P_j for the jth of $M \ll N$ components. The difference to the usual update equations for the GMM in the maximisation step of the EM algorithm is that the values $\boldsymbol{x}_n$ are not equally probable but occur with probability π_n. In Appendix C.8 we derive a modified version of the EM algorithm for this problem. There it is shown that the posterior component probabilities $P(j \mid \boldsymbol{x}_n)$ are computed

[3]Here the Kitagawa example is used which is discussed in detail in Section 6.7.

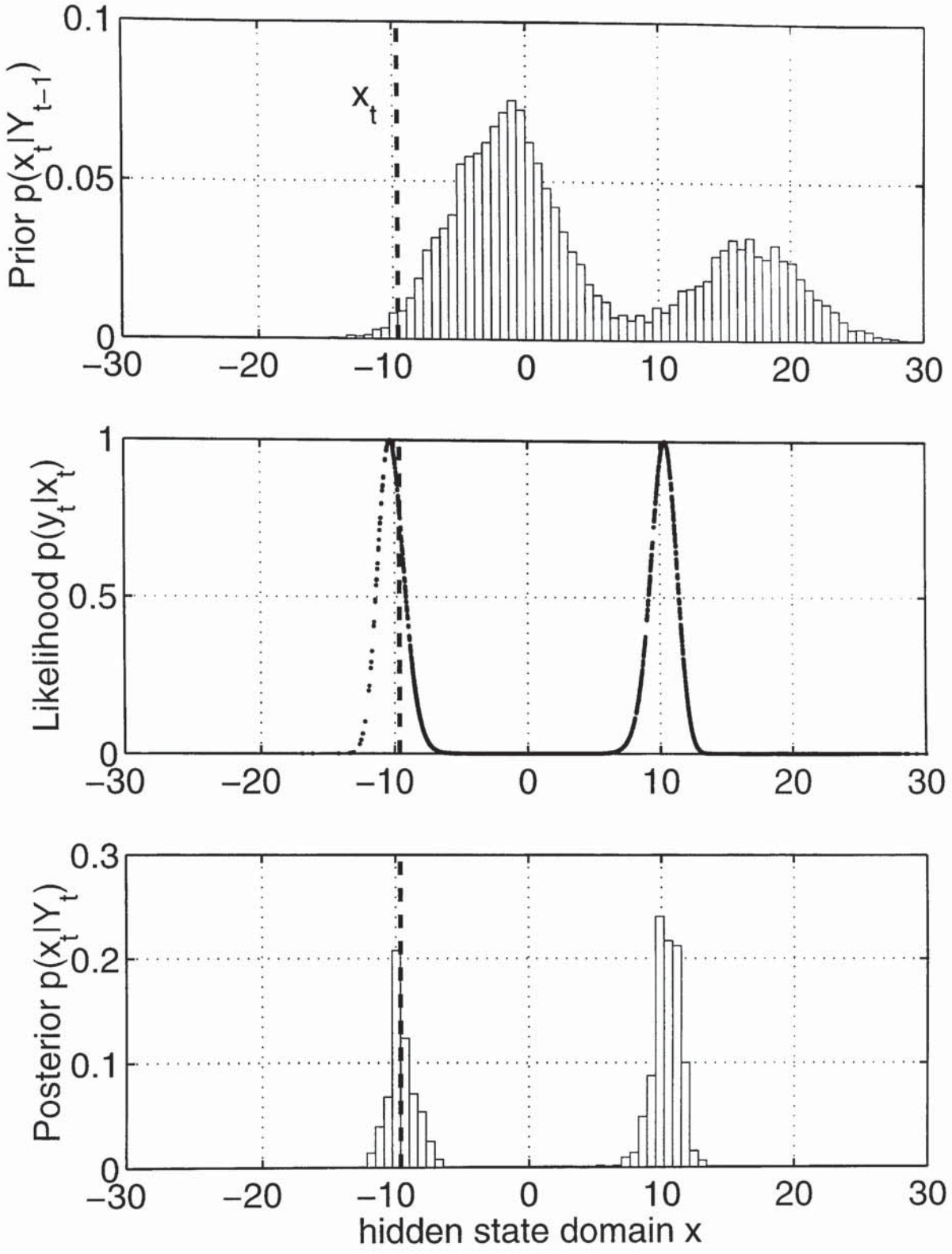

Figure 6.1: Prior, likelihood and Posterior for a bimodal prediction problem. The prior and posterior distributions are represented by a histogram and the current true hidden state x_t by a dashed line. The likelihood $p(y_t \mid x_t)$ for all prior samples x_t^- having observed y_t is plotted in the normalised form where each likelihood is divided by the maximum value over the whole sample set. Note that since the likelihood is here very different from the prior it takes a longer time to accept a proposed sample and with that to reach a reasonable size for the posterior sample set. Therefore the SIR algorithm suffers from a strong deviation of prior and likelihood

by simply weighting the usual EM update with the likelihood coefficients π_n.

$$P(j \mid \boldsymbol{x}_n)^* = P(j \mid \boldsymbol{x}_n)\, \pi_n = \frac{p(\boldsymbol{x}_n \mid j)\, P_j}{p(\boldsymbol{x}_n)}\, \pi_n.$$

We refer to this version as a Gaussian mixture model for weighted samples (WGMM). This is used from now on for estimating a distribution which can be written as the product of two other distributions.

6.6.2 The particle smoother

Beside the tendency for SIR and RS to degenerate the distributions involved in the NSSM such a purely sample-based approach is also impractical for non-trivial problems. Recalling the definition for the smoothing distribution from Equation (6.9) as

$$p(\boldsymbol{x}_t \mid \mathcal{Y}_T) = p(\boldsymbol{x}_t \mid \mathcal{Y}_t) \int \frac{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t)}{p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)}\, p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_T)\, d\boldsymbol{x}_{t+1} \tag{6.48}$$

it can be noticed that in order to estimate the smoothed distribution $p(\boldsymbol{x}_t \mid \mathcal{Y}_T)$ the current posterior $p(\boldsymbol{x}_t \mid \mathcal{Y}_t)$ is needed as well as the next predictive $p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)$ and smoothing distribution $p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_T)$. This means all these distributions, respectively their representative samples have to be stored which becomes computationally unfeasible quickly for real-world problems. Nevertheless, using now the mixture models estimated during the prediction and filtering step allows a compact representation of the distribution and thus an efficient sampling and evaluation.

In order to estimate the smoothed density $p(\boldsymbol{x}_t \mid \mathcal{Y}_T)$ the particle approach is applied in the same way as for filtering in the previous section. Now the posterior $p(\boldsymbol{x}_t \mid \mathcal{Y}_t)$ is chosen as the proposal density to get a sample set $\mathcal{X}_{t|t} \equiv \{\boldsymbol{x}_{t|t}^{(n)}\}_{n=1}^N$. This leaves estimating the weighting coefficient π_n as the approximation of the integral in Equation (6.48):

$$\pi_n = \int \frac{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t = \boldsymbol{x}_{t|t}^{(n)})}{p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)}\, p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_T)\, d\boldsymbol{x}_{t+1} \simeq \sum_{k=1}^{K} \frac{p(\boldsymbol{x}_{t+1} = \boldsymbol{x}_{t+1|t}^{(k)} \mid \mathcal{Y}_T)}{p(\boldsymbol{x}_{t+1} = \boldsymbol{x}_{t+1|t}^{(k)} \mid \mathcal{Y}_t)} \tag{6.49}$$

with propagated samples $\boldsymbol{x}_{t+1|t}^{(k)} = f(\boldsymbol{x}_{t|t}^{(n)}, \boldsymbol{u}_t, \boldsymbol{w}_t^{(k)})$ for $k = 1, \ldots, K$ and $K \gg 1$ using the representations for the posterior distribution $p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_t)$ and the smoothed distribution $p(\boldsymbol{x}_{t+1} \mid \mathcal{Y}_T)$ estimated in the previous smoothing step. Finally, the smoothed distribution can now be estimated via the weighted Gaussian mixture model approach (Appendix C.8) acting on the set $\{(\boldsymbol{x}_{t|t}^{(n)}, \pi_n)\}_{n=1}^N$ of samples $\boldsymbol{x}_{t|t}^{(n)}$ with their corresponding weighting coefficient π_n.

6.6.3 Learning the nonlinear model

Using the Gaussian distributions in Equations (6.2) and (6.4), representing the nonlinear state space model, the log of the joint probability distribution in Equation (6.12) becomes

$$\begin{aligned}\log p(\mathcal{X}_T, \mathcal{Y}_T) &= -\frac{T}{2}\log|\boldsymbol{R}| - \frac{1}{2}\sum_{t=1}^{T}[\boldsymbol{y}_t - \boldsymbol{g}(\boldsymbol{x}_t)]'\,\boldsymbol{R}^{-1}\,[\boldsymbol{y}_t - \boldsymbol{g}(\boldsymbol{x}_t)] \\ &\quad -\frac{T-1}{2}\log|\boldsymbol{Q}| - \frac{1}{2}\sum_{t=2}^{T}[\boldsymbol{x}_t - \boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)]'\,\boldsymbol{Q}^{-1}\,[\boldsymbol{x}_t - \boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)] \\ &\quad -\frac{1}{2}\log|\boldsymbol{V}| - \frac{1}{2}[\boldsymbol{x}_1 - \boldsymbol{\pi}]'\,\boldsymbol{V}^{-1}\,[\boldsymbol{x}_1 - \boldsymbol{\pi}] - \frac{T(n+m)}{2}\log 2\pi. \end{aligned} \tag{6.50}$$

Taking now the expectations $\langle . \rangle$ with respective to the state distribution $p(\mathcal{X}_T \,|\, \mathcal{Y}_T, \boldsymbol{\theta})$ we get finally the expected log likelihood as

$$\begin{aligned}\mathcal{Q}(\boldsymbol{\theta}) = &-\frac{1}{2}\sum_{t=1}^{T}\Big\{\boldsymbol{y}_t'\boldsymbol{R}^{-1}\boldsymbol{y}_t - 2\boldsymbol{y}_t'\boldsymbol{R}^{-1}\big\langle \boldsymbol{g}(\boldsymbol{x}_t)\big\rangle + \big\langle \boldsymbol{g}(\boldsymbol{x}_t)'\boldsymbol{R}^{-1}\boldsymbol{g}(\boldsymbol{x}_t)\big\rangle\Big\} \\ &-\frac{1}{2}\sum_{t=2}^{T}\Big\{\big\langle \boldsymbol{x}_t'\boldsymbol{Q}^{-1}\boldsymbol{x}_t\big\rangle - \big\langle \boldsymbol{x}_t'\boldsymbol{Q}^{-1}\boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\big\rangle \\ &\qquad - \big\langle \boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)'\boldsymbol{Q}^{-1}\boldsymbol{x}_t\big\rangle + \big\langle \boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)'\boldsymbol{Q}^{-1}\boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\big\rangle\Big\} \\ &-\frac{1}{2}\Big\{\big\langle \boldsymbol{x}_1'\boldsymbol{V}^{-1}\boldsymbol{x}_1\big\rangle - \big\langle \boldsymbol{x}_1'\big\rangle\boldsymbol{V}^{-1}\boldsymbol{\pi} - \boldsymbol{\pi}'\boldsymbol{V}^{-1}\big\langle \boldsymbol{x}_1\big\rangle + \boldsymbol{\pi}'\boldsymbol{V}^{-1}\boldsymbol{\pi} + \log|\boldsymbol{V}|\Big\} \\ &-\frac{T}{2}\log|\boldsymbol{R}| - \frac{T-1}{2}\log|\boldsymbol{Q}| - \frac{T(n+m)}{2}\log 2\pi. \end{aligned} \tag{6.51}$$

Filtering, smoothing and learning will be performed on the training set while 'true' prediction is only performed on the test set. In addition, for the learning we can fix the centres the RBF networks after a few iterations using factor analysis for initialisation purposes, then only the weights have to be adapted, which represents a linear optimisation problem. When a Gaussian kernel is used in the RBF networks also the widths can remain fixed after being determined by a Gaussian mixture model, for instance.

Furthermore, it seems useful to restrict the output function of the RBF networks to approach zero outside the range of available data by not providing a bias unit. It seems to be a valid prior to set a function value to zero for an unseen input. This prevents amplifications and oscillations for the network functions and the hidden state sequence during the learning process.

6.7 Comparison

This section focuses on the prediction ability of the nonlinear model compared to the linear model. As one artificial example the following system of equations has been chosen (Kitagawa,

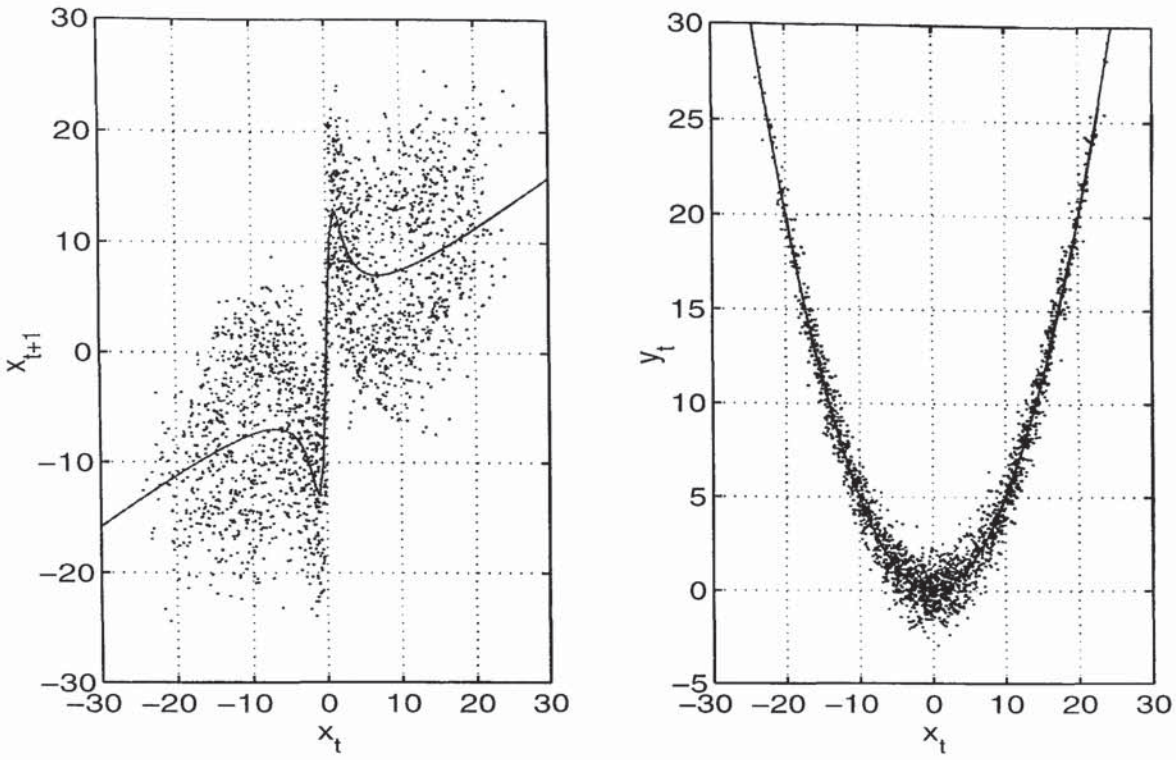

Figure 6.2: The training and test samples for the Kitagawa example: The next hidden state sample x_{t+1} (left) and the current observation sample y_t (right) as a function of the current hidden state sample x_t superimposed with the corresponding system and observation function.

1987):

$$x_t = \frac{x_{t-1}}{2} + \frac{25x_{t-1}}{1+x_{t-1}^2} + 8\cos 1.2t + v_t \tag{6.52}$$

$$y_t = \frac{x_t^2}{20} + w_t \tag{6.53}$$

with $x_0 \sim \mathcal{N}(0,5)$, $v_t \sim \mathcal{N}(0,10)$ and $w_t \sim \mathcal{N}(0,1)$.

A time series of 2000 data points has been generated of which the first 1600 serve as the training data and the remaining 400 points are used for test purposes. The training and test points for the hidden state and the observation are plotted in Figure 6.2 against the current state sample. There the nonlinear and nontrivial nature of the system and observation function becomes clear.

To model such a complex nonlinearity the system and observations function are represented each by a RBF network with ten hidden units and a thin plate spline kernel ($r^2 \log r$). As a sufficient and computationally feasible number of prior and posterior samples we choose 10000 and 1000 respectively. For a smaller size, such as 1000 and 100 for the two sample sets respectively, the problem of degenerating distributions in the Gaussian mixture fitting step occured very often for this particular example.

To test the suffiency we applied the filter and smoother for the given model and realized a very good estimation of the hidden state sequence. The NSSM was initialised with the state

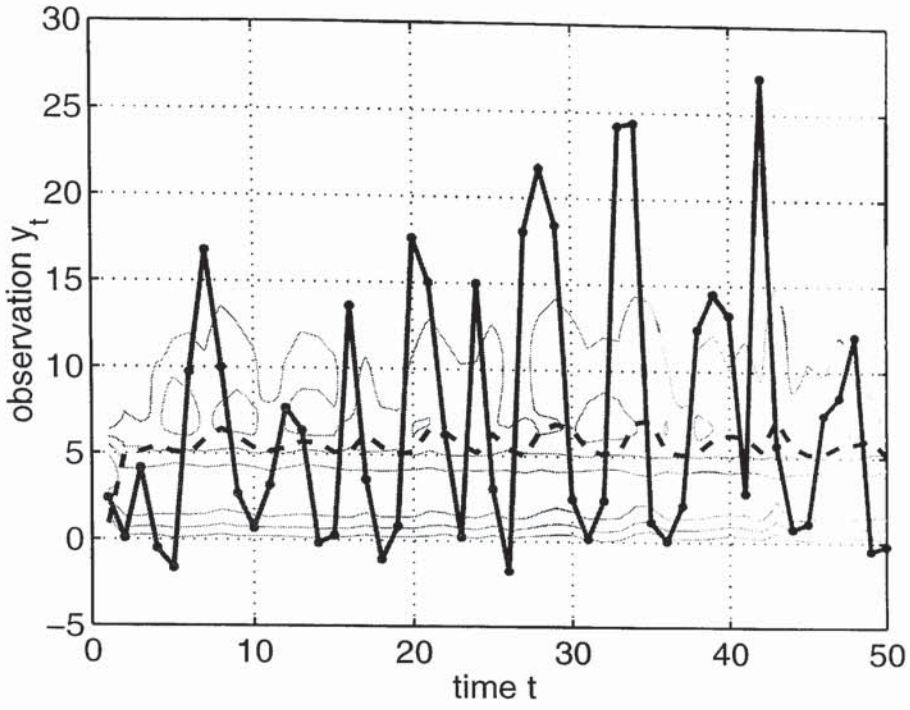

Figure 6.3: The first 50 observations y_t of the test set of the Kitagawa example (thick) superimposed by the mean of the Kalman prediction (dashed) and the predictive distribution of the NSSM represented as a contour plot of equal probabilities of 0.03, 0.06, 0.12, 0.24 and 0.48.

estimate of the Kalman smoother and the estimates for the noise covariances $\boldsymbol{R}$ and $\boldsymbol{Q}$ as well as the initial (Gaussian) state distribution represented by its mean $\boldsymbol{x}_0$ and its covariance $\boldsymbol{V}_0$. The latter parameter then remained fixed during the process while the system and the observation function were updated in the M step after each new E step. The running time is several days on a Pentium III workstation due to the chosen size of the sample sets. During a limited test period several trials (order of ten) were performed, however, the computational complexity prohibits the usually desired trial size (order of one hundred).

The first segment of 50 points in the test set is shown together with the linear and nonlinear prediction in Figure 6.3. For the Kalman estimate it is sufficient to report just the mean. For the nonlinear filter the mean would be misleading. As it can be seen the predictive distribution is very often positive skewed indicating a higher probability for larger positiv values. However, the results in terms of the achieved loglikelihood are not too different. In Figure 6.4 the loglikelihoods for the training and test set are shown for each iteration along with the values achieved for the Kalman filter. Despite exceeding the better value for the loglikelihood on both training and test set for the NSSM after just one iteration only a slightly superior value is achieved in saturation (after five iterations).

One possible explanation is that the initialisation by the linear solution obtained with the Kalman smoother is leading to a local minimum. From there the algorithm is not able to "escape" to a different minimum in the error function. In order to analyse this phenomenon

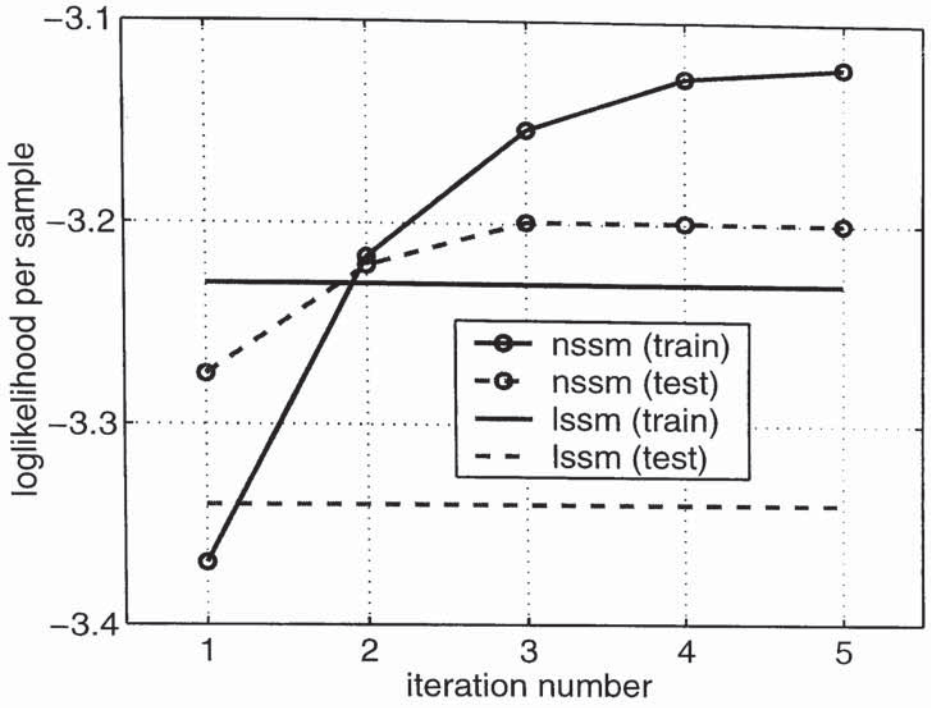

Figure 6.4: The log of the likelihood for the training (solid) and test (dashed) set of the Kitagawa example for the nonlinear state space model (circled line) and the linear state space model (straight line). For the LSSM only the last value is used for a better comparison.

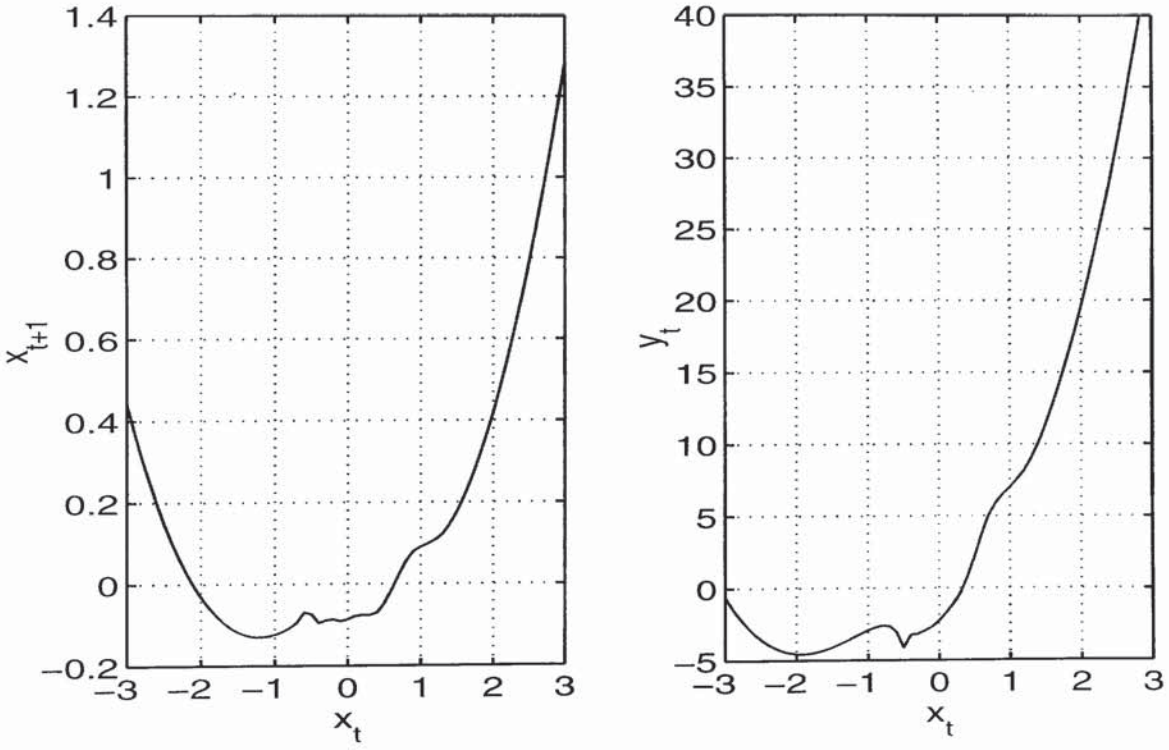

Figure 6.5: The learned nonlinear functions in the NSSM for the nonlinear state space model example represented by a RBF network for the Kitagawa data: The next hidden state x_{t+1} (left) and the current observation y_t (right) as a function of the current hidden state x_t.

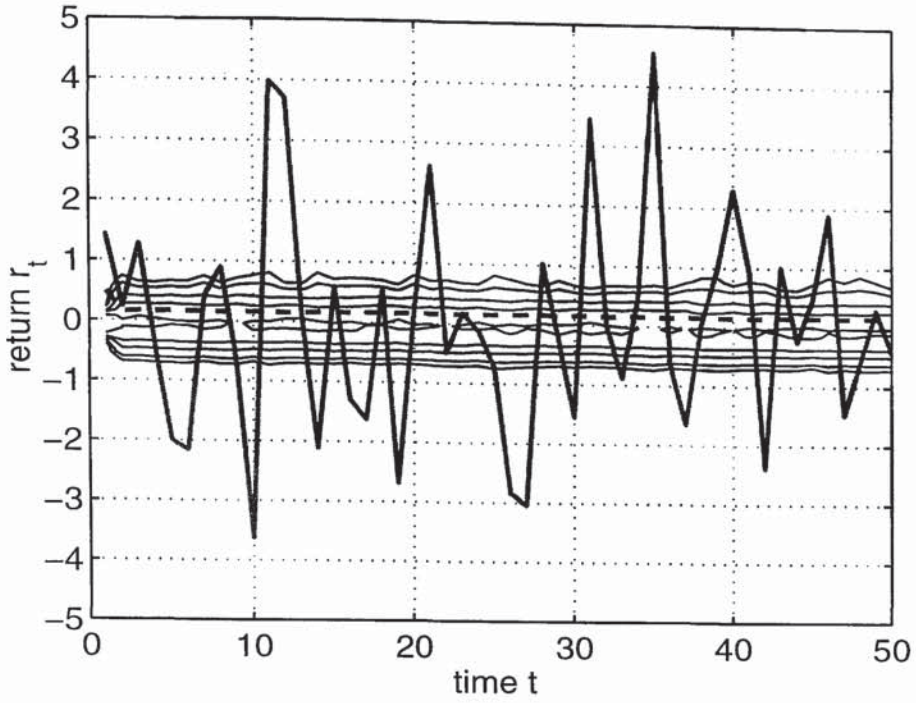

Figure 6.6: The first 50 observations y_t of the test set of the IBM example (thick) superimposed by the mean of the Kalman prediction (dashed) and the predictive distribution of the NSSM represented as a contour plot of equal probability densities of 0.03, 0.06, 0.12, 0.24 and 0.48.

Figure 6.5 shows both the system and the observation function depending on the current hidden state. The reconstructed (smoothed) hidden states range from around -1 to 3. This means that the system function maps preferably into a smaller positive interval such that the state values would collapse in the absence of noise. Within the mentioned interval the system function can furthermore roughly approximated by a semi-linear function which could explains the just slightly better performance compared to the linear Kalman filter.

It can therefore be concluded that by combining the two stages in this way with the initialisation by the linear solution is not sufficient to guarantee a significantly superior results than what can be obtained via the linear approach.

It is furthermore difficult to use an different initialisation scheme automatically. For instance, a random initialisation was tried instead of using the linear solution by the Kalman filter. This was even less efficient than the linear initialisation and furthermore lead to numerical problems such as collapsing Gaussian kernels in the Gaussian mixture estimation of the involved densities, the collapse of the sample distribution due to a highly peaked prior and numerical overflow due to an amplifying system function.

Despite these unexpected results we studied the capability of such a model to predict the one-day ahead value of IBM returns. For computational reasons we choose the last 1500 point of the set 4 of the IBM data and allocated 1000 points for the training and 500 for the test. The model structure remained the same as in the experiment before, 10 hidden units for

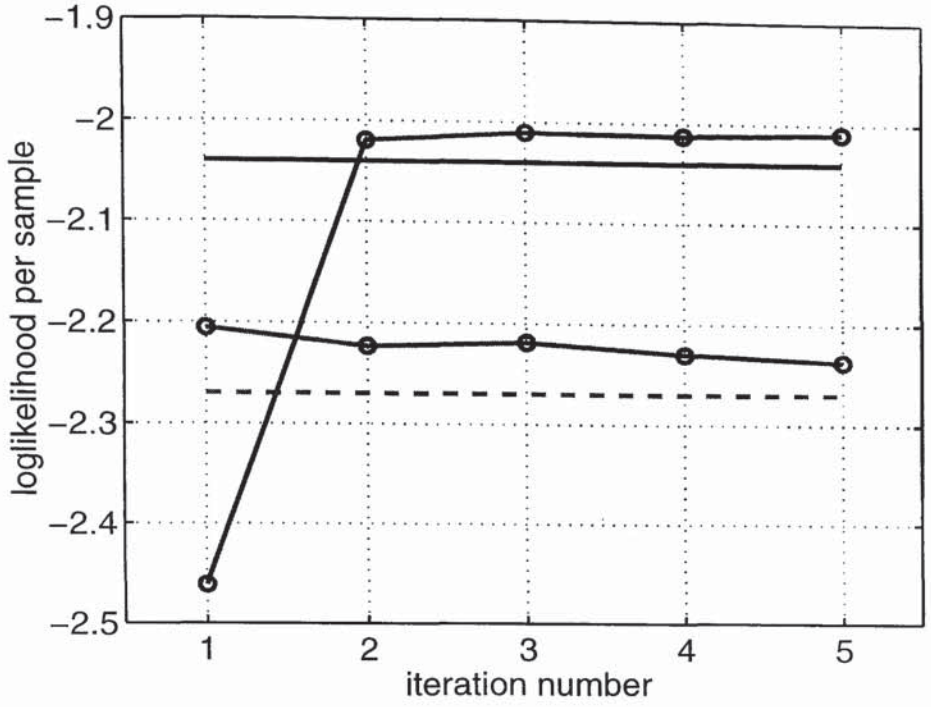

Figure 6.7: The log of the likelihood for the training (solid) and test (dashed) set of the IBM example for the nonlinear state space model (circled line) and the linear state space model (straight line). For the LSSM only the last value is used for a better comparison.

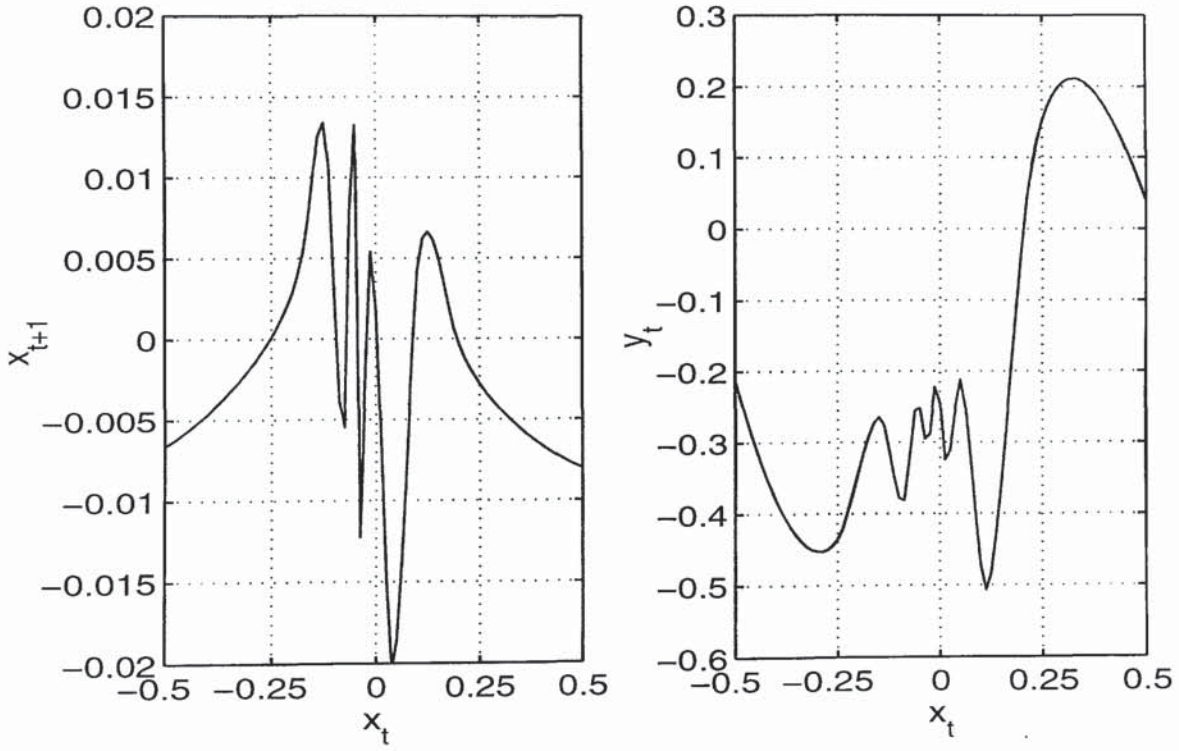

Figure 6.8: The learned nonlinear functions in the NSSM for the nonlinear state space model example represented by a RBF network for the IBM returns: The next hidden state x_{t+1} (left) and the current observation y_t (right) as a function of the current hidden state x_t.

the RBF networks with a thin plate spline activation function. The initialisation was done again with the solution by a linear Kalman filter.

Figure 6.6 shows the first 50 points of the test set together with the mean of the Kalman prediction and a contour plot to represent the predictive probability density of the nonlinear model. It can be seen that the linear solution is equivalent with the mean of the training set. The mode of the predictive distribution of the nonlinear model is much closer to zero, the distribution itself is slightly positively skewed. However, in terms of likelihood the difference is almost negligible. In Figure 6.7 for the nonlinear solution is better than for the linear one, however, not to a large amount.

It was therefore interesting to see the form of the approximated systems and observation function. Figure 6.8 shows that these functions are modelled as being highly nonlinear. It seems also that overfitting occurs in the center of the data distribution. Furthermore, since the model does not achieve a significantly better performance than the linear model it could be concluded that the solution presents a local minimum.

6.8 Discussion

This chapter has provided a framework for a nonlinear state space model as an extension to the static factor models introduced earlier. In comparison to other approaches for state space modeling the proposed model is neither restricted in its internal dynamics nor in the observation function.. The system and observation function are both modelled by individual RBF networks allowing a rich class of nonlinear functions. Furthermore, the use of Gaussian mixture models makes it possible to have non-Gaussian posterior and predictive distributions such as multimodal and asymmetric distributions.

However, the richness in this model has so far prevented effective inference and learning. The main problems have been attributed to an improper initialisation either with the linear solution from which it seems to be difficult to avert or a random one which might far from the solution or even lead to a divergence in the estimation process.

One possibility to handle these problems could be to start with several random initialisations and to stop updating those which show numerical problems. However, since this algorithm has a high time and space complexity due to the sampling procedures involved a multiple run with different initial configurations seems to be computationally too demanding at the moment.

A more promising approach would be the usage of a further restricted system and observation function, for instance, in form of a moderate and smooth nonlinearity, such that the estimation is less likely to show the same numerical difficulties. This would furthermore avoid overfitting. However, we realised earlier that the used network configuration was necessary to approximate the nonlinear system function in the Kitagawa example. Restricting the

network's complexity could then mean to underfit the data and therefore to be even closer to the linear solution. This could mean that a fixed network configuration is less favorable here. It remains to be tested if a model with a growing complexity achieves a better generalisation performance. Additionally, different kernel functions could be tested for the RBF networks. Here only thin plate spline functions were used since they showed the best approximation capabilities on the Kitagawa example and furthermore do not require a width parameter such as the variance for the Gaussian kernel. However, this and other kernels could be tested provided a robust update scheme for the width exists.

A similar approach could be tried for the Gaussian mixture models. Here a maximum number of modes could be determined in advance as well as a minimum covariance of each mixture component in order to avoid the collapse of a component.

Concluding the experiments it must be said that with the currently available learning scheme the theoretical generalisation capabilities of such a model are not exhausted. However, improvements in terms of the initialisation and the ability to escape local minima as well as the continuously increasing availability of computational ressources should show the practical relevance of such a model, soon.

Chapter 7

Conclusion

The goal of this thesis was to analyse financial returns from two angles: the deterministic and the stochastic paradigm. Historically, these two branches have focused on different aspects in financial data under opposit assumptions. However, both approaches also have not been able to fully explain all empirical phenomena indicating useful and missing elements in both of them.

In order to investigate how valid those assumptions are and how a model could be derived which incorporates useful elements from each side, a hierarchy of hypotheses about the nature of financial returns was established and tested. Starting from both angles with the most basic assumptions the test results at each level were then used to obtain a more complex model leading finally to a fusion of the elements into one model.

At first, a specific approach from dynamical systems theory has been employed in order to test if the apparent 'random' behaviour in financial returns is caused by a nonlinear determinism. Here we used several algorithms for estimating the fractal dimension of embedded returns. Unfortunately, either the assumption of a low fractal dimension could not have been confirmed or the difference in the dimension estimate compared to the one of randomised data is not significant. This leads to the conclusion that for the tested financial data no low-dimensional deterministic structure has been found. Since it is infeasible to detect a high-dimensional determinism in daily financial returns due to the lack of sufficient data the only practical choice is to assume that either noise has corrupted the deterministic data part or that the return itself is noise. Unfortunately, the method employed for dimension estimation relies on the assumption that the available data are noise-free.

The next step was therefore to look at the other end of the spectrum of analytical tools for methods which can deal with noise. For that purpose several density estimation techniques were explored. Using parametric distributions we have shown that the hypothesis of a Gaussian random walk for financial time series cannot be confirmed. Instead, stable distributions like the Paretian or Laplace fit the data better. Furthermore, a kernel density estimator was employed for approximating the probability density and the characteristic function. Since

such non-parametric methods are very costly and also unreliable for low-density regions a semiparametric approach such as the Gaussian mixture model was investigated. There the advantage of different basis functions was demonstrated. Furthermore, an extension for a mixture model estimate of weighted samples was introduced.

All the used density estimation techniques confirmed the leptokurtic (fat tails and highly peaked) and slightly skewed, and therefore non-Gaussian shape of financial return distributions. Since the Gaussian assumption is still broadly used in the financial industry, the risk for a portfolio drawdown, for instance, will therefore be under estimated.

Beside marginal also conditional distributions of the current return given the previous return were estimated. The results confirmed a slightly positive correlation in the mean and a variance growing with the amplitude of the current return.

In order to explore the idea of such a correlation being introduced due to a linear or nonlinear transformation of truly independent factors principal and independent component analysis was performed on embedded financial data. Both methods confirmed the significant correlation structure within the magnitude of returns. This correlation seems also to change slowly over time. However, a clear-cut way of separating noise from the signal and therefore a dimensionality reduction could not be achieved.

Nevertheless, all these methods are static approaches, ignoring explicitely time-dependencies although they can be applied for a moving time-window. However, they cannot explain why volatility persists over time. They can only detect this phenomenon. The idea was therefore to implement a temporal structure in the model in order to allow a certain dependency of the current from previous values. This lead to a (dynamical) state space model. In such a model an underlying dynamical volatility process, for instance, can result in returns correlated in their magnitude.

Such a temporal structure was implemented in the model by a linear and nonlinear system function. Those can be represented by a matrix transformation or a nonlinear neural network. For the nonlinear state space model it is furthermore necessary to allow non-Gaussian distributions as a result of a nonlinear transformation of a Gaussian distribution. Here Gaussian mixture models were used as efficient, flexible and compact density estimators. In order to perform inference in this model the particle filter approach was employed. The created particles in turn were used to estimate the mixture models. Despite the theoretical opportunities, the lack of an efficient initialisation scheme and numerical problems have prevented so far a practical use of this model.

References

Abraham, A. and D. Ikenberry (1994), June. The individual investor and the weekend effect. *Journal of Financial and Quantitative Analysis*.

Anderson, B. D. O. and J. B. Moore (1979). *Optimal Filtering*. Information and System Sciences Series. Englewood Cliffs, New Jersey: Prentice-Hall.

Back, A. D. and A. S. Weigend (1997). What drives stock returns? An independent component analysis. Technical Report IS-97-022, Leonard N. Stern School of Business, New York University. available via: `http://www.stern.nyu.edu/~aweigend/Research/Papers/ICA/Draft.ps`.

Bishop, C. M. (1995). *Neural Networks and Pattern Recognition*. Oxford University Press.

Briegel, T. and V. Tresp (1998). Fisher scoring and a mixture of modes approach for approximate inference and learning in nonlinear state space models. In M. I. Jordan, M. J. Kearns, and S. A. Solla (Eds.), *Advances in Neural Information Processing Systems*, Volume 10. Cambridge, MA: MIT Press.

Brock, W. A. (1991). Causality, chaos, explanation and prediction in economics and finance. Chapter 10, pp. 230–279. Boca Raton, Florida: CRC Press.

Brock, W. A., W. D. Dechert, and J. A. Scheinkmann (1987). A test for independence based on the correlation dimension. Technical report, Department of Economics University of Wisconsin, University of Houston and University of Chicago.

Broomhead, D. S. and G. P. King (1986). Extracting qualitative dynamics from experimental data. *Physica* **20D**, 217–236.

Broomhead, D. S. and D. Lowe (1988). Multi-variable functional interpolation and adaptive networks. *Complex Systems* **2**, 321–355.

Cardoso, J.-F. and A. Souloumiac (1993). Blind beamforming for non-gaussian signals. *IEE Proceedings F* **140** (6), 362–370.

Cattell, R. (1966). The scree test for the number of factors. *J. Multiv. Behav.* **1**, 245–276.

Chatfield, C. (1996). *The Analysis of Time Series* (5 ed.). London: Chapmann and Hall.

Chobanov, G., P. Mateev, S. Mittnik, and S. Rachev (1996). Modeling the distribution of highly volatile exchange-rate time series. Technical Report 90, Institute of Statistics and Econometrics, Christian Albrechts University at Kiel, Germany.

Cootner, P. (1964). *The Random Character of Stock Market Prices*. Cambridge: MIT.

Deco, G., R. Neuneier, and B. Schürmann (1997). Non-parametric data selection for neural learning in non-stationary time series. *Neural Networks* **10** (3), 401–407.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B* **39** (1), 1–38.

Diebold, F. X., T. A. Gunter, and A. S. Tay (1998). Evaluating density forecasts, with applications to financial risk management. *International Economic Review* **39**, 863–883.

Eckmann, J.-P. and D. Ruelle (1985). Ergodic theory of chaos and strange attractors. *Rev. Mod. Phys.* **57**, 617–656.

Engle, R. F. (Ed.) (1995). *ARCH: Selected Readings*. Advanced Texts in Econometrics. Oxford, UK: Oxford University Press.

Fama, E. F. (1965). The behaviour of stock market prices. *Journal of Business* **38**, 34–105.

Fama, E. F. (1970), March. Efficient capital markets: A review of theory and empirical work. *Journal of Finance* **25** (2), 383–417.

Fraser, A. M. and H. L. Swinney (1986), February. Independent coordinates for strange sttractors from mutual information. *Physical Review A* **33** (2), 1134–1139.

Gershenfeld, N. A. and A. S. Weigend (1994). The future of time series: Learning and understanding. In *Time Series Prediction: Forecasting the Future and Understanding the Past*, Volume 15 of *Santa Fe Institute Studies in the Sciences of Complexity*. New York: Addison Wesley.

Ghahramani, Z. and G. E. Hinton (1996), February. Parameter estimation for linear dynamical systems. Technical Report CRG-TR-96-2, Department of Computer Science, University of Toronto, Canada. available via: `ftp://ftp.cs.toronto.edu/pub/zoubin/tr-96-2.ps.gz`.

Ghahramani, Z. and S. T. Roweis (1999). Learning nonlinear dynamical systems using an EM algorithm. In M. S. Kearnsa, S. A. Solla, and D. A. Cohn (Eds.), *Advances in Neural Information Processing Systems*, Volume 11. Cambridge, MA: MIT Press. available via: `http://www.gatsby.ucl.ac.uk/~zoubin/papers/nlds-ftp.ps.gz`.

Gordon, N. (1996), September. On-line filtering for nonlinear/non-Gaussian state space models. Technical report, Defense Research Agency, Malvern, England. available via: `http://siwg.dra.hmg.gb/pattern/papers/postscript_files/njg_smie96.ps.gz`.

Gordon, N. (1997), January. A hybrid bootstrap filter for target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems* **1** (33), 353–358. available via: `http://siwg.dra.hmg.gb/pattern/papers/postscript_files/njg_ieeeaes.ps.gz`.

Grassberger, P. (1983). Generalized dimensions of strange attractors. *Phys. Lett. A* **97**, 227–230.

Gultekin, M. and M. Gultekin (1983), December. Stock market seasonality: International evidence. *Journal of Finance Economics*.

Gupta, N. K. and R. K. Mehra (1974), December. Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculations. *IEEE Transactions on Automatic Control* **AC-19** (6), 774–783.

Hall, P. (1992). *The Bootstrap and Edgeworth Expansion*. Springer Series in Statistics. New York: Springer-Verlag.

Haugen, R. A. and K. Lakonishok (1988). *The Incredible January Effect*. Irwin, IL: Dow Jones.

Hausdorff, F. (1919). Dimension und äußeres Maß. *Math. Annalen* **79**, 157.

Holzfuss, J. (1987). Zur Messung von fraktalen Dimensionen und Lyapunov-Spektren nichtlinearer Systeme am Beispiel akustisch erzeugter Kavitationsblasen. Thesis, Univ. Göttingen.

Hsieh, D. A. (1989). Testing for nonlinear dependence in daily foreign exchange rates. *Journal of Business* **62**, 339–369.

Hsieh, D. A. (1991), December. Chaos and nonlinear dynamics: Applications to financial markets. *Journal of Finance* **46** (5), 1839–1878.

Hyvärinen, A. and E. Oja (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation* **9** (7), 1483–1492. available via: `http://www.cis.hut.fi/~aapo/ps/gz/NC97.ps.gz`.

Jolliffe, I. T. (1986). *Principal Component Analysis*. Berlin: Springer-Verlag.

Kitagawa, G. (1987). Non-Gaussian state-space modeling of nonstationary time series. *Journal of the American Statistical Association* **82** (400), 1032–1063.

LeBaron, B. (1992). Some relations between volatility and serial correlation in stock market returns. *Journal of Business* **65** (2), 199–219.

LeBaron, B. (1993), December. The joint dynamics and stability of stock prices and volume. Technical report, Department of Economics, University of Winconsin - Madison.

LeBaron, B. and A. S. Weigend (1997). A bootstrap evaluation of the effect of data splitting on financial time series. Technical Report IS-97-13, Leonard N. Stern School of Business, New York University. available via: `http://www.stern.nyu.edu/~aweigend/Research/Papers/LeBaron.Weigend_IEEETNN97.ps`.

Loistl, O. and I. Betz (1994). *Chaostheorie*. München: Oldenbourg.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *J. Atmos. Sci.* **20**, 130.

Lowe, D. and N. Hazarika (1997). Complexity modelling and stability characterisation for long-term iterated time series prediction. In *5th IEE International Conference on Artificial Neural Networks*, pp. 53–58. The Institute of Electrical Engineers.

MacKay, D. J. C. (1995). Probable networks and plausible predictions – a review of practical Bayesian methods for supervised neural networks. *Network* **6** (3), 469–505.

Mandelbrot, B. B. (1963). The variation of certain speculative prices. *Journal of Business* **36**, 394–419.

Mandelbrot, B. B. (1982). *The Fractal Geometry of Nature*. San Francisco: Freeman.

Mané, R. (1981). On the dimension of the compact invariant sets of certain nonlinear maps. In D. A. Rand and L. S. Young (Eds.), *Dynamical systems and turbulence*, Volume 898, pp. 230–242. Berlin: Springer.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance* **7** (1), 77–91.

May, R. M. (1976). Simple mathematical models with very complicated dynamics. *Nature* **261**, 459.

Mittnik, S. and S. T. Rachev (1993). Modeling asset returns with alternative stable distributions. *Econometric Reviews* **12**, 261–330.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*. Springer. Lecture Notes in Statistics 118.

Nolan, J. P. (1997), May. Maximum likelihood estimation of stable parameters. Technical report, American University, Washington D.C., USA.

Nolan, J. P. (1999), April. Stable distributions. Draft of a book. available via: `http://www.cas.american.edu/~jpnolan/chap1.ps`.

Packard, N. H., J. P. Crutchfield, J. D. Farmer, and R. S. Shaw (1980). Geometry from a time series. *Phys. Rev. Lett.* **45**, 712–716.

Peters, E. E. (1991). *Chaos and Order in the Capital Markets*. New York: John Wiley & Sons.

Pitt, M. K. and N. Shephard (1997). Filtering via simulation: Auxiliary particle filter. available via: `http://www.nuff.ox.ac.uk/economics/papers/1997/w13/sir.zip`.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected application in speech recognition. In *Proceedings of the IEEE*, Volume 77, pp. 257–286.

Roweis, S. and Z. Ghahramani (1999). A unifying review of linear gaussian models. *Neural Computation* **11** (2), 305–345. available via: `http://www.gatsby.ucl.ac.uk/~zoubin/papers/lds.ps.gz`.

Scheinkman, J. A. and B. LeBaron (1989). Nonlinear dynamics and stock returns. *Journal of Business* **62**, 311–337.

Schuster, H. G. (1994). *Deterministisches Chaos*. Weinheim: VCH Verlagsgesellschaft mbH.

Sharp, W. F. (1970). *Portfolio Theory and Capital Markets*. New York: McGraw-Hill.

Shumway, R. H. and D. S. Stoffer (1982). An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis* **3** (4), 253–264.

Skinner, J. E., C. M. Pratt, and T. Vybiral (1993). Reduction in the correlation dimension of heartbeat intervals precedes imminent ventricular fibrillation in human subjects. *Am. Heart J.* **125**, 731–743.

Stuart, A. and K. Ord (1994). *Distribution Theory* (Sixth ed.). Kendall's Advanced Theory of Statistics volume 1. Edward Arnold.

Takens, F. (1981). Detecting strange attractors in turbulence. In D. A. Rand and L.-S. Young (Eds.), *Dynamical Systems and Turbulence (Warwick 1980)*, Volume 898 of *Lecture Notes in Mathematics*, pp. 366–381. Berlin: Springer.

Theiler, J., S. Eubank, A. Longtin, B. Galdrikian, and J. D. Farmer (1992). Testing for nonlinearity in time series: the method of surrogate data. *Physica D* **58**, 77–94.

Timmer, J. and A. S. Weigend (1997). Modeling volatility using state space models. *International Journal of Neural Systems* **8** (4), 385–398. available via: `http://www.stern.nyu.edu/~aweigend/Research/Papers/StateSpace`.

Tipping, M. E. and C. M. Bishop (1997). Mixtures of principal component analyzers. Technical Report NCRG/97/003, Neural Computing Research Group, Aston University, Birmingham, England. available via: `http://neural-server.aston.ac.uk/Papers/postscript/NCRG_97_003.ps.Z`.

Tong, H. (1990). *Nonlinear Time Series Analysis: A Dynamical Systems Approach*. Oxford: Oxford University Press.

Vaga, T. (1990). The coherent market hypothesis. *Financial Analysts Journal* **46** (6), 36–49.

Weigend, A. S. (1999). personal communication.

Weigend, A. S., M. Mangeas, and A. N. Srivastava (1995). Nonlinear gated experts for time series: discovering regimes and avoiding overfitting. *International Journal of Neural Systems* **6**, 373–399. available via: `ftp://ftp.cs.colorado.edu/pub/Time-Series/MyPapers/experts.ps.Z`.

Wolf, A., J. B. Swift, H. L. Swinney, and J. A. Vastano (1985). Determining Lyapunov exponents from time series. *Physica* **16D**, 285–317.

Yule, G. (1927). On a method of investigating periodicity in disturbed series with special reference to Wolfer's sunspot numbers. *Phil. Trans. Roy. So. London* **A 226**, 267–298.

Zoubir, A. M. and B. Boashash (1998), January. The bootstrap and its application in signal processing. *IEEE Signal Processing Magazine* (1), 56–76.

Appendix A

Datasets

This appendix introduces the datasets used in this thesis. These include daily time series of stock indices and prices, foreign currencies exchange rates, government bonds and commodity prices. Table A.1 summarises each time series in the form of the number of totally available points and the place on which the equity was traded to that price. Thereby the currencies' exchange rates are given as the value of the foreign currency in US Dollar. The futures price series are all continuous contracts.

Dataset	**Description**	**Points**	**Place**
Stock market indices			
DJIA	Dow Jones Industrial Average	28336	NYSE
SP500	Standard & Poors 500	18991	NYSE
DAX	German Stock Market Index	10000	Frankfurt
Single stock prices			
COCA	Coca Cola	7472	NYSE
IBM	International Business Machines Corp.	9457	NYSE
Foreign currencies exchange rates futures			
DEMUSD	Deutschmark/US-Dollar futures	6179	US
GBPUSD	British Pound/US-Dollar futures	6178	US
Government bond futures			
USNOTES	US Treasury 10 years Notes futures	4349	US
USBONDS	US Treasury 30 years Bonds futures	5532	US
Commodity futures			
COFFEE	Coffee futures	7503	US
SILVER	Silver futures	7424	US

Table A.1: Description of the financial datasets used with daily prices including the total number of prices and the trading place.

Regarding the stock data sets the following comments are necessary: The New York Stock Exchange (NYSE) closed August 1 until December 11, 1914 due to war. On December 12 the market opened again and finished that day with -33% compared to the last trading

day before the war. Therefore this return was removed from the set 1 of the DJIA series of one-day returns in order not to bias the estimate. Furthermore, the whole time series was divided into four datasets with approximately constant variance. For instance, set 2 covers the time around the global stock market crash in 1929, the following Great Depression and the recovery during which the variance is significantly higher during that time. In set 3 the three returns from October 19th to 21st 1987 accounting for the crash and a short recovery were removed. The same was done for the other US stock index S&P 500 and the stocks COCA and IBM. In total the following outliers have been removed:

- DJIA: 4 (12-Dec-1914, 19-Oct-1987 until 21-Oct-1987)
- SP500, COCA, IBM: 3 (19-Oct-1987 until 21-Oct-1987)
- DAX: 2 (29-May-1970, 16-Oct-1989)

Another important issue concerns the breadth of the Dow Jones Industrial average. Before August 1914 the index included 12 stocks, but it was expanded to 20 stocks when the exchange reopened. A further expansion took place on October 1, 1928 to 30 stocks. This number has been kept constant since, however, stocks are irregularly substituted in order to take into account the changing capitalisation of the corresponding companies. Another important difference between the DJIA and SP500 is that SP500 incorporates dividends and uses a weighting according to market capitalisation which is not done for the DJIA. There the prices of the 30 stocks are taken and simply averaged.

Table A.2 gives an overview of all the training and test sets for each time series. The choice of how to split a data set into different subsets is not trivial in case of nonstationary data (LeBaron and Weigend, 1997). Therefore we have selected segments of the time series which appear to be stationary, at least up to second order. Furthermore we checked for outliers which are outside ten standard deviations from the mean. This is mainly an issue for the stock prices and indices during the crash in October 1987. Figure 2.1 shows the price and return of the DJIA for 100 years with the separation in the four sets.

The data have been acquired from the following public sources:

- DJIA:
  1896-1999 `http://www.economagic.com/em-cgi/data.exe/djind/day-djiac`
  1901-1998 `ftp://ftp.quoteline.ch/dj1900d.exe`
  1900-1993 `ftp://wueconb.wustl.edu/econ-wp/data/papers/9603/9603001.tar.gz`
  1928-1999 `http://chart.yahoo.com/d?s=^DJI`
- SP500:
  `http://chart.yahoo.com/d?s=SPX`

Dataset	Type	Points	Begin	End	Outliers
DJIA	set 1	9500	26-May-1896	14-Jul-1928	1
	set 2	3499	16-Jul-1928	08-Apr-1940	0
	set 3	8000	09-Apr-1940	21-Apr-1970	0
	set 4	7332	22-Apr-1970	29-Apr-1999	3
SP500	set 1	8000	09-Apr-1940	21-Apr-1970	0
	set 2	7332	22-Apr-1970	29-Apr-1999	3
DAX	set 1	5000	09-Aug-1963	05-Nov-1982	1
	set 2	3998	08-Nov-1982	10-Jul-1998	1
IBM	set 1	5000	02-Jan-1962	08-Dec-1981	0
	set 2	4453	09-Dec-1981	28-Jul-1999	3
COCA	set 1	3736	01-Jan-1970	12-Oct-1984	0
	set 2	3732	15-Oct-1984	28-Jul-1999	3
DEMUSD	set 1	3000	14-Feb-1975	07-Jan-1987	0
	set 2	3178	08-Jan-1987	02-Aug-1999	0
GBPUSD	set 1	3000	14-Feb-1975	07-Jan-1987	0
	set 2	3177	08-Jan-1987	02-Aug-1999	0
USNOTES	set 1	2000	04-May-1982	30-Mar-1990	0
	set 2	2348	02-Apr-1990	02-Aug-1999	0
USBONDS	set 1	2500	16-Aug-1979	11-Jul-1989	0
	set 2	2531	12-Jul-1989	02-Aug-1999	0
SILVER	set 1	4000	06-Jan-1970	27-Dec-1985	0
	set 2	3423	30-Dec-1985	02-Aug-1999	0
COFFEE	set 1	3500	20-Aug-1973	26-Aug-1987	0
	set 2	3002	26-Aug-1987	02-Aug-1999	0

Table A.2: Description of the individual datasets used: the number of returns excluding the outliers, the begin and end date of each set and the number of excluded data points

- IBM:

  `http://chart.yahoo.com/d?s=IBM`

- GBPUSD, DEMUSD, USNOTES, USBONDS, SILVER, COFFEE:

  `http://www.chdwk.com/data/futures.html`

Appendix B

Monte Carlo methods

This appendix provides a brief summary of Monte Carlo methods for integrating non-normal distributions via the approach of averaging over a finite number of samples from that distribution. Specifically, two relevant methods are described: Sampling/importance resampling and rejection sampling.

B.1 Introduction

Monte Carlo methods are used to calculate the expectation $\mathbb{E}[F(\boldsymbol{X})]$ of an integrand of interest $F(\boldsymbol{X})$ under the distribution $p(\boldsymbol{x})$ where this cannot be solved analytically due to the non-Gaussianity of $p(\boldsymbol{x})$ or nonlinearity of F (Neal, 1996; MacKay, 1995):

$$\mathbb{E}[F(\boldsymbol{X})] = \int F(\boldsymbol{x})\, p(\boldsymbol{x})\, d\boldsymbol{x}. \tag{B.1}$$

Using a set $\mathcal{X} = \{\boldsymbol{x}_n\}_{n=1}^N$ of N samples $\boldsymbol{x}_n \sim p(\boldsymbol{x})$ allows us to represent $p(\boldsymbol{x})$ by the empirical density function $\hat{p}(\boldsymbol{x})$ as a sum of Dirac delta functions at the $\boldsymbol{x}_n$ according to Equation (4.21). Then the above integral can be approximated by the sum of the function values of these N samples:

$$\mathbb{E}[F(\boldsymbol{X})] \simeq \int F(\boldsymbol{x})\, \hat{p}(\boldsymbol{x})\, d\boldsymbol{x} = \frac{1}{N} \sum_{n=1}^{N} F(\boldsymbol{x}_n). \tag{B.2}$$

This means that a distribution can be represented to an arbitrary degree by its own samples. Such samples can be produced, for example, by sampling first uniformly with replacement from the sample set $\mathcal{X}$ and then applying the function F to each of these proposed samples. This approach is applied, for instance, in the prediction step of the inference for nonlinear state space models in Equation (6.46).

The remaining problem is to provide samples from $p(\boldsymbol{x})$ which is not always trivial. It is feasible in the case that $p(\boldsymbol{x})$ is proportional to an easy to sample from density $q(\boldsymbol{x})$ weighted by a likelihood term $\pi(\boldsymbol{x})$. Samples can then be obtained by proposing from $q(\boldsymbol{x})$ and accepting it with a probability proportional to $\pi(\boldsymbol{x})$. This method is employed, for

instance, when Bayes' theorem is applied in the update step of the inference part in order to compute the posterior distribution in Equation (6.8). The two strategies used here for using the likelihood term $\pi(\boldsymbol{x})$ in order to obtain samples from $p(\boldsymbol{x})$ are now summarised.

B.2 Sampling/importance resampling

Sampling/importance resampling has been frequently used in the context of nonlinear state space models (Gordon, 1996; Gordon, 1997; Pitt and Shephard, 1997). It requires that $\int_{-\infty}^{\infty} \pi(\boldsymbol{x})\, d\boldsymbol{x} = 1$ and $\pi(\boldsymbol{x}) > 0$. In this way the weight $\pi(\boldsymbol{x})$ itself represents the probability with which the proposed sample is accepted. Given a set $\{(\boldsymbol{x}_n^*, \pi_n)\}_{n=1}^{N^*}$ of samples $\boldsymbol{x}_n^* \sim p(\boldsymbol{x})$ and weighting coefficients $\pi_n \in (0,1)$ an algorithm can be derived easily for the sampling/importance resampling approach which is sketched in Algorithm B.2.

Algorithm 2 Sampling/Importance Resampling: Given a set $\{(\boldsymbol{x}_n^*, \pi_n)\}_{n=1}^{N^*}$ of samples $\boldsymbol{x}_n^* \sim p(\boldsymbol{x})$ and weighting coefficients $\pi_n \in (0,1)$ with $\sum_{n=1}^{N^*} \pi_n = 1$ representing the distribution $\pi(\boldsymbol{x})$ it produces a set $\{\boldsymbol{x}_k\}_{k=1}^{N}$ of samples $\boldsymbol{x}_k \sim p(\boldsymbol{x})\,\pi(\boldsymbol{x})$

for $k = 1$ to N **do**
  repeat
    $n \sim U(1, N^*)$
    $u \sim U(0,1)$
  until $u \leq \pi_n$
  $\boldsymbol{x}_k \Leftarrow \boldsymbol{x}_n^*$
end for
return $\{\boldsymbol{x}_k\}_{k=1}^{N}$

SIR will become very imprecise when π_n is very variable, which means it has a high variance. This happens when the likelihood is very peaked compared to the prior. SIR is furthermore vulnerable to *sample impoverishment*, the collapsing of the sample set to a single point. It also needs a large sample sizes in order to achieve a random sample set.

B.3 Rejection sampling

A slightly modified sampling method is rejection sampling (Pitt and Shephard, 1997). The variation is to normalise the weighting coefficients by the maximum coefficient:

$$\pi_n^* = \frac{\pi_n}{\max_i \pi_i}. \tag{B.3}$$

For a continuous case it might be difficult to determine $\max_i \pi_i$ exactly. However, for a discrete set of samples, like here, this is not an issue. This results therefore in a significant

speed-up and with that allows a larger sample size. Nevertheless, the problem of highly peaked priors compared to the likelihood remains.

Appendix C

ML estimation for parametric and mixture distributions

This appendix describes in detail the maximum likelihood (ML) approach of determining parametric and semiparametric density models. For all these models the likelihood of the data can be evaluated as a function of the model parameters. The ML approach simply determines then those parameters which give the highest likelihood to the data. In the following the ML procedure is discussed for the stable Paretian distribution, the Gaussian, Cauchy, Weibull and Laplace distributions. For all but the stable Paretian this is straightforward. There a nonlinear optimisation needs to be performed, which is considered in detail next. After, the ML approach is briefly reviewed for Gaussian mixture models and finally discussed thoroughly for the combined mixture model and the weighted Gaussian mixture model.

C.1 Stable Paretian distributions

Parameter estimation for stable distributions is performed via a nonlinear bootstrap. Since the parameters α, β and γ in Equation (4.29) are constrained to certain intervals a parameter transformation is necessary to be able to use a unconstrained nonlinear optimisation tool. Therefore the following re-parameterisations have been used with initial values α_0, β_0 and γ_0 to start the iterative estimation procedure:

- For the stability index $\alpha \in (1,2)$ we introduce $\tilde{\alpha} = -\log \frac{2-\alpha}{\alpha-1}$ and use the sigmoid function[1] in order to define $\alpha = \text{sigm}(\tilde{\alpha}) + 1$. The derivative is $\frac{d\alpha}{d\tilde{\alpha}} = (\alpha-1)(2-\alpha)$. As a prior for α_0 one could sample uniformly from $(1,2)$. However, we found empirically a much more efficient scheme is to sample from $(1.5, 1.9)$, a more likely range for this parameter.

[1] The sigmoid function is defined as $\text{sigm}(x) = (1+e^{-x})^{-1} \in (0,1)$.

- For the skewness $\beta \in (-1, 1)$ we take $\tilde{\beta} = -\log(\frac{2}{\beta+1} - 1)$ with inverse $\beta = 2\,\mathrm{sigm}(\beta) - 1$ and derivative $\frac{d\beta}{d\tilde{\beta}} = (\beta + 1)(1 - \beta)$ and set $\beta_0 = 0$.

- For the scaling parameter $\gamma > 0$ a simple log transformation is sufficient such as $\tilde{\gamma} = \log \gamma$ and inverse $\gamma = e^{\tilde{\gamma}}$ with derivative $\frac{\partial \gamma}{\partial \tilde{\gamma}} = e^{\tilde{\gamma}} = \gamma$. Its initial value is set to $\gamma_0 = \sqrt{\hat{\sigma}^2/2}$.

- The mode $\delta \in \mathbb{R}$ is unconstrained and can furthermore be set to $\delta = \hat{\mu}$ for $\alpha > 1$, the only case which we are going to consider here.

C.2 The Gaussian distribution

For the Gaussian distribution the sample negative log-likelihood E can be obtained from the probability density function in Equation (4.30):

$$E = \frac{1}{2}\log(2\pi) + \log\sigma + \frac{1}{2\sigma^2 T}\sum_{t=1}^{T}(x_t - \mu)^2. \tag{C.1}$$

The maximum likelihood solution for the parameters can then be derived explicitly as the sample mean $\hat{\mu}$ and the sample variance $\hat{\sigma}^2$:

$$\hat{\mu} = \frac{1}{T}\sum_{t=1}^{T} x_t, \qquad \hat{\sigma}^2 = \frac{1}{T}\sum_{t=1}^{T}(x_t - \hat{\mu})^2. \tag{C.2}$$

In order to get an unbiased estimate $\hat{\sigma}^2$ for the variance, T in the denominator is replaced by $T - 1$ (Bishop, 1995).

C.3 The Cauchy distribution

For the Cauchy distribution with its p.d.f. given in Equation (4.31) the negative sample log-likelihood E is given by

$$E = \log\pi - \log\gamma + \frac{1}{T}\sum_{t=1}^{T}\log\left\{\gamma^2 + (x_t - \delta)^2\right\} \tag{C.3}$$

In order to estimate the parameters E is differentiated with respect to δ and γ:

$$\frac{\partial E}{\partial \delta} = -\frac{2}{T}\sum_{t=1}^{T}\frac{x_t - \delta}{\gamma^2 + (x_t - \delta)^2}, \qquad \frac{\partial E}{\partial \gamma} = \frac{1}{\gamma} + \frac{2}{T}\sum_{t=1}^{T}\frac{\gamma}{\gamma^2 + (x_t - \delta)^2}. \tag{C.4}$$

These partial derivatives are then used by a quasi-Newton nonlinear optimisation tool to minimise E.

C.4 The Weibull distribution

The negative sample log-likelihood E is derived for the Weibull distribution from the density function in Equation (4.32) as

$$E = \log 2 - \log(\lambda\alpha) + \frac{\lambda}{T}\sum_{t=1}^{T}|x_t|^{\alpha} - \frac{\alpha-1}{T}\sum_{t=1}^{T}\log|x_t| \tag{C.5}$$

where all values $x_t = 0$ are excluded from the calculation. The corresponding derivatives are then

$$\frac{\partial E}{\partial \alpha} = -\frac{1}{\alpha} + \frac{\lambda}{T}\log\alpha\sum_{t=1}^{T}|x_t|^{\alpha} - \frac{1}{T}\sum_{t=1}^{T}\log|x_t|, \qquad \frac{\partial E}{\partial \lambda} = -\frac{1}{\lambda} + \frac{1}{T}\sum_{t=1}^{T}|x_t|^{\alpha}. \tag{C.6}$$

C.5 The Laplace distribution

For the Laplace distribution the maximum likelihood solution for the parameters can be calculated analytically from the negative sample log-likelihood E using the probability density function in Equation (4.33):

$$E = \log 2 - \log\lambda + \frac{\lambda}{T}\sum_{t=1}^{T}|x_t - \mu|. \tag{C.7}$$

Its partial derivatives with respect to the median μ and λ are given as:

$$\frac{\partial E}{\partial \mu} = -\frac{\lambda}{T}\sum_{t=1}^{T}\operatorname{sgn}(x_t - \mu), \qquad \frac{\partial E}{\partial \lambda} = -\frac{1}{\lambda} + \frac{1}{T}\sum_{t=1}^{T}|x_t - \mu|. \tag{C.8}$$

Setting these to zero yields $\hat{\mu} = \hat{x}_m$, the sample median according to Equation (4.7) and $\hat{\lambda}$ as the inverse of the sample mean absolute deviation corresponding to Equation (4.8):

$$\hat{\lambda}^{-1} = \hat{\nu} = \frac{1}{T}\sum_{t=1}^{T}|x_t - \hat{\mu}|. \tag{C.9}$$

C.6 Gaussian mixture models

The error function E is defined as the negative log-likelihood given in Equation (4.35):

$$E = -\log\mathcal{L}(\theta) = -\sum_{t=1}^{T}\log\left\{\sum_{j=1}^{M}p(x_t|j)\,P_j\right\}. \tag{C.10}$$

The Expectation step of the EM algorithm estimates the posterior probability $P(j|x)$ that a given data point x has been generated by the component j of the model. In the Maximisation step new parameter estimates are determined by minimising the likelihood based on the

calculated posterior in the E step. For a purely Gaussian mixture model with component densities $p(x \mid j)$ given by

$$p(x \mid j) = \mathcal{N}(x \mid \mu_j, \sigma_j^2) = \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left\{-\frac{(x-\mu_j)^2}{2\sigma_j^2}\right\}. \tag{C.11}$$

the parameter update equations have the following form:

$$\mu_j^* = \frac{\sum_{t=1}^T P(j|x_t)\, x_t}{\sum_{t=1}^T P(j \mid x_t)} \tag{C.12}$$

$$\sigma_j^{2^*} = \frac{\sum_{t=1}^T P(j \mid x_t)\,(x_t - \mu_j^*)^2}{\sum_{t=1}^T P(j \mid x_t)} \tag{C.13}$$

$$P_j^* = \frac{1}{T}\sum_{t=1}^T P(j \mid x_t) \tag{C.14}$$

Thereby the $*$ marks a new value after the update. The posterior probability $P(j \mid x)$ for component j is given via the Bayes' theorem as

$$P(j \mid x) = \frac{p(x \mid j)P_j}{p(x)} = \frac{p(x \mid j)P_j}{\sum_{i=1}^M p(x \mid i)P(i)}. \tag{C.15}$$

C.7 Gauss-Laplacian mixture models

The update equations for the combined Gaussian-Laplace mixture model distribution can easily be adapted by using the partial derivatives of the error function E with respect to the median μ and the scale λ in Equation (C.8):

$$\frac{\partial E}{\partial \mu} = -\lambda \sum_{t=1}^T P(L|x_t)\,\mathrm{sgn}(x_t - \mu), \qquad \frac{\partial E}{\partial \lambda} = -\sum_{t=1}^T P(L|x_t)\left(\frac{1}{\lambda} - |x_t - \mu|\right) \tag{C.16}$$

with the Laplace prior $P(L)$ and posterior $P(L|x_t) = \frac{p(x_t;\mu,\lambda)P(L)}{p(x_t)}$. Setting these to zero yields

$$\mu^* = \arg\min_{\mu} \left| \sum_{x_t > \mu} P(L|x_t) - \sum_{x_t < \mu} P(L|x_t) \right|, \quad \lambda^* = \frac{\sum_{t=1}^T P(L|x_t)}{\sum_{t=1}^T P(L|x_t)|x_t - \mu^*|} \tag{C.17}$$

as the new parameter estimates for the Laplace component within the mixture. The Gaussian components are updated the same way as before according to Equations (C.12), (C.13) and (C.14) with j running over all Gaussian components.

C.8 Gaussian mixture model for weighted data

Here we present a modification to the usual EM algorithm for Gaussian mixture models for the case of unequally weighted data samples. In this general case the task is to approximate

a continuous density function $p(x)$ with a mixture model given a set $\{(x_n, \pi_n)\}_{n=1}^N$ of samples $x_n \sim p(x)$ with weighting coefficients π_n with $\pi_n \geq 0$ and $\sum_{n=1}^N \pi_n = 1$ representing a probability measure for the discrete set of x_n

In the normal version of mixture models, as dealt with in the two previous sections, the observed data x_n are equally-probable samples, thus their corresponding weights $\pi_n = N^{-1}$. For unequal weights the EM algorithm has to be modified slightly. This more general EM version will be used later to determine the posterior distribution from a set of prior samples with corresponding likelihoods.

Using $\pi(x)$ to denote the probability function induced by the π_n we are interested in the product $p(x) = q(x)\,\pi(x)$ which will be modelled as a Gaussian mixture parameterised by $\boldsymbol{\theta} \equiv \{(P_j, \mu_j, \sigma_j^2)\}_{j=1}^M$:

$$\hat{p}(x \,|\, \boldsymbol{\theta}) = \sum_{j=1}^{M} p(x \,|\, j)\, P_j \tag{C.18}$$

with the component density functions $p(x \,|\, j)$ being normals given in Equation (C.11). The mixture model is estimated by minimising the Kullback-Leibler divergence between the 'true' distribution $p(x)$ and the approximating distribution $\hat{p}(x)$:

$$\mathrm{KL}(p, \hat{p}) = -\int p(x) \log \frac{p(x)}{\hat{p}(x)}\, dx \tag{C.19}$$

The goal is to find the parameter vector $\boldsymbol{\theta}^*$ which minimises this divergence:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \left\{ -\int p(x) \log \frac{p(x)}{\hat{p}(x)}\, dx \right\} \tag{C.20}$$

Since the entropy term $\int p(x) \log p(x)\, dx$ in Equation (C.19) does not depend on $\boldsymbol{\theta}$, it is sufficient to maximise just the second term, the *cross-entropy* between the distributions p and $\hat{p}$. This can be approximated for an arbitrarily large number of samples $x_n \sim p(x)$ by a finite sum:

$$E = \int p(x)\, \pi(x) \log \hat{p}(x)\, dx \approx \sum_{n=1}^{N} \pi_n \log \hat{p}(x_n) \tag{C.21}$$

which defines the error function E we are going to maximise. The derivative of E with respect to a distributional parameter θ is then given by

$$\frac{\partial E}{\partial \theta} = \sum_{n=1}^{N} \pi_n \frac{1}{\hat{p}(x_n)} \frac{\partial \hat{p}(x_n)}{\partial \theta} \tag{C.22}$$

and the derivatives of the mixture model density $\hat{p}(x_n)$ with respect to the model parameters become

$$\frac{\partial \hat{p}(x_n)}{\partial \mu_j} = p(x_n \,|\, j)\, P_j \left[\frac{x_n - \mu_j}{\sigma_j^2} \right] \tag{C.23}$$

$$\frac{\partial \hat{p}(x_n)}{\partial \sigma_j^2} = \frac{1}{2}\, p(x_n \,|\, j)\, P_j \frac{1}{\sigma_j^2} \left[\frac{(x_n - \mu_j)^2}{\sigma_j^2} - 1 \right] \tag{C.24}$$

$$\frac{\partial \hat{p}(x_n)}{\partial \gamma_j} = \sum_{i=1}^{M} p(x_n \,|\, i)\, [\delta_{ij} P_j - P(i) P_j] = P_j\, [p(x_n|j) - p(x_n)]\,. \tag{C.25}$$

Setting now the derivatives of E to zero for each model parameter and using Bayes' theorem for the posterior probability $P(j \mid x_n) = p(x_n \mid j)\, P_j / \hat{p}(x_n)$ we obtain

$$0 = \sum_{n=1}^{N} \pi_n \, P(j \mid x_n) \, (x_n - \mu_j) \tag{C.26}$$

$$0 = \sum_{n=1}^{N} \pi_n \, P(j \mid x_n) \left[\frac{(x_n - \mu_j)^2}{\sigma_j^2} - 1 \right] \tag{C.27}$$

$$0 = \sum_{n=1}^{N} \pi_n \, P_j \, \frac{p(x_n|j) - p(x_n)}{\hat{p}(x_n)}. \tag{C.28}$$

This results in the following solutions for the new parameter estimates (denoted by *):

$$\mu_j^* = \frac{\sum_{n=1}^{N} P(j \mid x_n) \, \pi_n \, x_n}{\sum_{n=1}^{N} P(j \mid x_n) \, \pi_n} \tag{C.29}$$

$$\sigma_j^{2^*} = \frac{\sum_{n=1}^{N} P(j \mid x_n) \, \pi_n \, (x_n - \mu_j^*)^2}{\sum_{n=1}^{N} P(j \mid x_n) \, \pi_n} \tag{C.30}$$

$$P_j^* = \sum_{n=1}^{N} P(j \mid x_n) \, \pi_n. \tag{C.31}$$

With that the only modification to the usual EM algorithm is to multiply the posterior probabilities $P(j \mid x_n)$ with the weighting coefficient π_n at the begin of each EM iteration:

$$P(j \mid x_n)^* = P(j \mid x_n) \, \pi_n, \tag{C.32}$$

then the new parameter estimates are obtained as usual via the Equations (C.29), (C.30) and (C.31).

Appendix D

Density estimation results

This appendix summarises the results obtained by parametric and semiparametric density estimation techniques. For parametric distributions the estimated parameters are reported along the corresponding log likelihood. However, first the estimated four sample cumulants are listed in Table D.1 for all datasets. Note that the first two cumulants coincide with the mean and variance parameters for a Gaussian distribution. Table D.2 provides then a summary of the achieved log likelihoods for all estimation techniques for the DJIA data using set 1 as training and set 2 as a test set and vice versa.

In the following Tables D.3, D.5, D.7, D.9 and D.11 the distributional parameters are reported with the corresponding log likelihoods for the Gaussian, the stable Paretian, the Cauchy, the Weibull and the Laplace distribution for each individual data set. Since for the DJIA four data sets were used the Tables D.4, D.6, D.8, D.10 and D.12 contain the obtained log likelihoods for the DJIA data using the parameters estimated for each set.

Dataset	Type	mean μ	variance σ^2	skewness γ	kurtosis κ
DJIA	set 1	$+0.02 \pm 0.01$	1.03 ± 0.03	-0.47 ± 0.13	4.75 ± 0.79
	set 2	-0.01 ± 0.03	3.37 ± 0.18	$+0.13 \pm 0.26$	6.95 ± 1.27
	set 3	$+0.02 \pm 0.01$	0.48 ± 0.02	-0.85 ± 0.25	9.00 ± 1.85
	set 4	$+0.04 \pm 0.01$	0.90 ± 0.03	-0.21 ± 0.15	4.38 ± 0.98
SP500	set 1	$+0.02 \pm 0.01$	0.56 ± 0.02	-0.95 ± 0.25	10.13 ± 1.79
	set 2	$+0.04 \pm 0.01$	0.82 ± 0.02	-0.28 ± 0.16	4.87 ± 1.13
DAX	set 1	$+0.00 \pm 0.01$	0.67 ± 0.02	$+0.06 \pm 0.07$	1.14 ± 0.20
	set 2	$+0.06 \pm 0.02$	1.37 ± 0.06	-0.50 ± 0.27	7.27 ± 1.44
IBM	set 1	$+0.01 \pm 0.02$	1.88 ± 0.06	$+0.20 \pm 0.15$	3.58 ± 0.71
	set 2	$+0.05 \pm 0.02$	2.71 ± 0.12	$+0.22 \pm 0.20$	5.53 ± 0.96
COCA	set 1	$+0.03 \pm 0.02$	2.31 ± 0.10	-0.07 ± 0.20	4.72 ± 1.05
	set 2	$+0.09 \pm 0.03$	2.46 ± 0.09	-0.07 ± 0.14	2.75 ± 0.66
DEMUSD	set 1	$+0.01 \pm 0.01$	0.42 ± 0.02	$+0.38 \pm 0.22$	3.35 ± 1.36
	set 2	$+0.01 \pm 0.01$	0.38 ± 0.02	-0.07 ± 0.13	2.43 ± 0.38
GBPUSD	set 1	-0.00 ± 0.02	0.54 ± 0.02	$+0.12 \pm 0.17$	3.00 ± 0.67
	set 2	-0.00 ± 0.02	0.52 ± 0.02	-0.22 ± 0.15	3.05 ± 0.51
SILVER	set 1	$+0.02 \pm 0.04$	4.72 ± 0.17	-0.12 ± 0.08	1.10 ± 0.17
	set 2	-0.02 ± 0.03	2.32 ± 0.14	-0.46 ± 0.30	6.42 ± 1.60
COFFEE	set 1	-0.02 ± 0.04	4.03 ± 0.29	-0.97 ± 0.49	10.82 ± 4.32
	set 2	$+0.00 \pm 0.05$	5.06 ± 0.36	$+0.69 \pm 0.52$	10.57 ± 4.26
USBONDS	set 1	$+0.00 \pm 0.02$	0.91 ± 0.03	$+0.10 \pm 0.09$	1.28 ± 0.28
	set 2	$+0.01 \pm 0.01$	0.31 ± 0.01	-0.30 ± 0.11	1.71 ± 0.42
USNOTES	set 1	$+0.01 \pm 0.01$	0.30 ± 0.01	$+0.20 \pm 0.14$	2.21 ± 0.58
	set 2	$+0.01 \pm 0.01$	0.14 ± 0.01	-0.27 ± 0.13	2.06 ± 0.45

Table D.1: First four sample cumulants with empirical error bars obtained as the standard deviation of 1000 bootstrap estimates for each dataset. Note that the third and fourth cumulants are normalised by the corresponding power of the second cumulant according to Equation 4.17 in order to make these cumulants comparable over different datasets.

Distribution	$E_{1\|1}$	$E_{2\|1}$	$E_{2\|2}$	$E_{1\|2}$
Gauss	1.522	1.389	1.367	1.548
Stable	1.360	1.279	1.267	1.372
Cauchy	1.423	1.374	1.373	1.424
Weibull	1.408	1.315	1.301	1.425
Laplace	1.368	1.281	1.277	1.372
GMM3	1.355	1.278	1.264	1.377
GLMM2	1.357	1.275	1.266	1.375
KDE	1.367	1.375	1.271	1.290

Table D.2: Negative Log-likelihood on set 1 and set 2 for the DJIA for all investigated distributions

Dataset	Type	E_{Gauss}	$E^*{}_{Gauss}$	E_{Stable}	$E^*{}_{Stable}$
DJIA	set 1	1.43 ± 0.01	–	1.37 ± 0.01	–
	set 2	2.03 ± 0.03	–	1.90 ± 0.02	–
	set 3	1.05 ± 0.02	–	0.96 ± 0.01	–
	set 4	1.36 ± 0.02	–	1.32 ± 0.01	–
SP500	set 1	1.13 ± 0.02	1.16 ± 0.01	1.02 ± 0.01	1.07 ± 0.01
	set 2	1.32 ± 0.02	1.37 ± 0.02	1.27 ± 0.01	1.33 ± 0.02
DAX	set 1	1.21 ± 0.01	1.32 ± 0.01	1.20 ± 0.01	1.98 ± 0.01
	set 2	1.58 ± 0.02	1.75 ± 0.05	1.49 ± 0.02	1.92 ± 0.01
IBM	set 1	1.74 ± 0.02	1.76 ± 0.01	1.69 ± 0.01	1.71 ± 0.01
	set 2	1.92 ± 0.02	1.96 ± 0.03	1.85 ± 0.01	1.87 ± 0.02
COCA	set 1	1.84 ± 0.02	1.84 ± 0.02	1.77 ± 0.02	1.78 ± 0.02
	set 2	1.87 ± 0.02	1.87 ± 0.02	1.84 ± 0.01	1.86 ± 0.02
DEMUSD	set 1	1.03 ± 0.02	1.03 ± 0.02	0.98 ± 0.02	0.99 ± 0.02
	set 2	1.06 ± 0.02	1.06 ± 0.02	1.03 ± 0.02	1.03 ± 0.02
GBPUSD	set 1	1.05 ± 0.02	1.05 ± 0.02	0.98 ± 0.02	0.99 ± 0.02
	set 2	1.03 ± 0.02	1.03 ± 0.02	0.96 ± 0.02	0.96 ± 0.02
USBONDS	set 1	1.37 ± 0.02	1.81 ± 0.07	1.36 ± 0.02	1.59 ± 0.04
	set 2	0.83 ± 0.02	1.04 ± 0.01	0.81 ± 0.02	0.98 ± 0.01
USNOTES	set 1	0.82 ± 0.02	1.01 ± 0.06	0.79 ± 0.02	0.89 ± 0.03
	set 2	0.43 ± 0.02	0.55 ± 0.02	0.40 ± 0.02	0.48 ± 0.01
SILVER	set 1	2.30 ± 0.03	2.54 ± 0.09	2.16 ± 0.02	2.24 ± 0.02
	set 2	1.88 ± 0.02	2.01 ± 0.02	1.79 ± 0.02	1.86 ± 0.02
COFFEE	set 1	2.23 ± 0.03	2.25 ± 0.03	2.09 ± 0.02	2.11 ± 0.02
	set 2	2.34 ± 0.03	2.36 ± 0.04	2.25 ± 0.02	2.27 ± 0.02

Table D.3: Negative log likelihood E and cross log-likelihood E^* for the Gaussian distribution obtained by 1000 bootstrap runs and for the stable Paretian distrbution using 100 bootstrap runs on each dataset

Set	$E_{.\|1}$	$E_{.\|2}$	$E_{.\|3}$	$E_{.\|4}$
set 1	$\mathbf{1.43 \pm 0.01}$	1.68 ± 0.02	1.62 ± 0.04	1.44 ± 0.02
set 2	2.57 ± 0.09	$\mathbf{2.03 \pm 0.03}$	4.10 ± 0.22	2.75 ± 0.10
set 3	1.17 ± 0.01	1.60 ± 0.02	$\mathbf{1.05 \pm 0.02}$	1.13 ± 0.01
set 4	1.37 ± 0.01	1.66 ± 0.02	1.49 ± 0.03	$\mathbf{1.36 \pm 0.01}$

Table D.4: Negative log-likelihood and cross log-likelihoods for the Gaussian distribution obtained by 1000 maximum likelihood bootstrap runs on each DJIA dataset

Dataset	α	β	γ	δ
DJIA	1.71 ± 0.02	-0.21 ± 0.02	0.58 ± 0.01	$+0.02 \pm 0.01$
	1.47 ± 0.03	-0.08 ± 0.03	0.85 ± 0.02	-0.01 ± 0.03
	1.74 ± 0.02	-0.25 ± 0.02	0.39 ± 0.01	$+0.02 \pm 0.01$
	1.77 ± 0.02	$+0.00 \pm 0.02$	0.56 ± 0.01	$+0.04 \pm 0.01$
SP500	1.64 ± 0.02	-0.21 ± 0.03	0.39 ± 0.00	$+0.03 \pm 0.01$
	1.75 ± 0.02	-0.01 ± 0.03	0.54 ± 0.01	$+0.04 \pm 0.01$
DAX	1.88 ± 0.01	$+0.09 \pm 0.12$	0.53 ± 0.01	$+0.00 \pm 0.01$
	1.76 ± 0.02	-0.21 ± 0.06	0.66 ± 0.01	$+0.06 \pm 0.02$
IBM	1.77 ± 0.02	$+0.16 \pm 0.05$	0.82 ± 0.01	$+0.01 \pm 0.02$
	1.75 ± 0.02	$+0.10 \pm 0.06$	0.96 ± 0.01	$+0.05 \pm 0.03$
COCA	1.67 ± 0.02	$+0.06 \pm 0.06$	0.84 ± 0.02	$+0.03 \pm 0.03$
	1.83 ± 0.02	$+0.21 \pm 0.06$	0.98 ± 0.02	$+0.10 \pm 0.02$
DEMUSD	1.65 ± 0.03	$+0.17 \pm 0.04$	0.38 ± 0.01	$+0.01 \pm 0.01$
	1.75 ± 0.03	$+0.02 \pm 0.06$	0.42 ± 0.01	$+0.00 \pm 0.01$
GBPUSD	1.56 ± 0.04	-0.09 ± 0.04	0.36 ± 0.01	-0.02 ± 0.01
	1.61 ± 0.03	-0.09 ± 0.04	0.36 ± 0.01	$+0.00 \pm 0.01$
USBONDS	1.80 ± 0.03	$+0.03 \pm 0.06$	0.60 ± 0.01	$+0.00 \pm 0.02$
	1.83 ± 0.04	-0.18 ± 0.16	0.35 ± 0.01	$+0.01 \pm 0.01$
USNOTES	1.75 ± 0.04	-0.01 ± 0.06	0.33 ± 0.01	$+0.01 \pm 0.01$
	1.78 ± 0.03	-0.14 ± 0.04	0.23 ± 0.01	$+0.01 \pm 0.01$
SILVER	1.61 ± 0.03	-0.06 ± 0.03	1.21 ± 0.03	$+0.03 \pm 0.04$
	1.52 ± 0.03	-0.05 ± 0.04	0.79 ± 0.02	-0.01 ± 0.03
COFFEE	1.55 ± 0.03	-0.05 ± 0.05	1.08 ± 0.02	$+0.02 \pm 0.04$
	1.61 ± 0.03	-0.08 ± 0.04	1.33 ± 0.03	-0.01 ± 0.05

Table D.5: Estimates for the parameters α, β, γ and δ of the stable Paretian distribution obtained by 100 quasi-newton maximum likelihood bootstrap runs on each dataset

Set	$\boldsymbol{E}_{.\|1}$	$\boldsymbol{E}_{.\|2}$	$\boldsymbol{E}_{.\|3}$	$\boldsymbol{E}_{.\|4}$
set 1	$\mathbf{1.37 \pm 0.01}$	1.47 ± 0.01	1.48 ± 0.02	1.37 ± 0.01
set 2	2.04 ± 0.03	$\mathbf{1.90 \pm 0.02}$	2.37 ± 0.04	2.08 ± 0.03
set 3	1.05 ± 0.01	1.28 ± 0.01	$\mathbf{0.96 \pm 0.01}$	1.04 ± 0.01
set 4	1.32 ± 0.01	1.44 ± 0.01	1.41 ± 0.02	$\mathbf{1.32 \pm 0.01}$

Table D.6: Negative log-likelihood and cross log-likelihoods for the Stable distribution obtained by 100 maximum likelihood bootstrap runs on each DJIA dataset

Dataset	Type	δ	γ	E	E^*
DJIA	set 1	$+0.07 \pm 0.01$	0.50 ± 0.01	1.47 ± 0.01	–
	set 2	$+0.05 \pm 0.02$	0.75 ± 0.02	1.95 ± 0.02	–
	set 3	$+0.05 \pm 0.01$	0.33 ± 0.00	1.07 ± 0.01	–
	set 4	$+0.04 \pm 0.01$	0.48 ± 0.01	1.42 ± 0.01	–
SP500	set 1	$+0.06 \pm 0.01$	0.34 ± 0.00	1.10 ± 0.01	1.13 ± 0.01
	set 2	$+0.04 \pm 0.01$	0.45 ± 0.01	1.37 ± 0.01	1.39 ± 0.01
DAX	set 1	-0.01 ± 0.01	0.46 ± 0.01	1.34 ± 0.01	1.37 ± 0.01
	set 2	$+0.09 \pm 0.02$	0.57 ± 0.01	1.60 ± 0.01	1.62 ± 0.02
IBM	set 1	-0.03 ± 0.02	0.69 ± 0.01	1.80 ± 0.01	1.81 ± 0.01
	set 2	$+0.02 \pm 0.02$	0.84 ± 0.01	1.96 ± 0.01	1.98 ± 0.02
COCA	set 1	$+0.00 \pm 0.02$	0.73 ± 0.01	1.86 ± 0.02	1.87 ± 0.01
	set 2	$+0.04 \pm 0.03$	0.85 ± 0.01	1.97 ± 0.01	1.98 ± 0.02
DEMUSD	set 1	-0.02 ± 0.01	0.32 ± 0.01	1.05 ± 0.02	1.06 ± 0.02
	set 2	-0.02 ± 0.01	0.35 ± 0.01	1.12 ± 0.02	1.13 ± 0.02
GBPUSD	set 1	$+0.00 \pm 0.01$	0.30 ± 0.01	1.04 ± 0.02	1.04 ± 0.02
	set 2	$+0.01 \pm 0.01$	0.31 ± 0.01	1.04 ± 0.02	1.04 ± 0.02
USBONDS	set 1	$+0.01 \pm 0.02$	0.50 ± 0.01	1.47 ± 0.02	1.55 ± 0.03
	set 2	$+0.02 \pm 0.01$	0.30 ± 0.01	0.93 ± 0.02	1.01 ± 0.01
USNOTES	set 1	$+0.02 \pm 0.01$	0.28 ± 0.01	0.88 ± 0.02	0.92 ± 0.03
	set 2	$+0.01 \pm 0.01$	0.19 ± 0.00	0.51 ± 0.02	0.55 ± 0.02
SILVER	set 1	$+0.07 \pm 0.03$	1.01 ± 0.02	2.22 ± 0.02	2.27 ± 0.02
	set 2	$+0.01 \pm 0.02$	0.69 ± 0.01	1.84 ± 0.02	1.89 ± 0.01
COFFEE	set 1	$+0.04 \pm 0.03$	0.95 ± 0.02	2.16 ± 0.02	2.17 ± 0.02
	set 2	$+0.06 \pm 0.04$	1.13 ± 0.02	2.32 ± 0.02	2.33 ± 0.02

Table D.7: Estimates for the parameters δ and γ for the Cauchy distribution obtained by 1000 maximum likelihood bootstrap runs on each dataset with the corresponding negative log-likelihood E and cross log-likelihood E^*

Set	$E_{.\|1}$	$E_{.\|2}$	$E_{.\|3}$	$E_{.\|4}$
set 1	$\mathbf{1.47 \pm 0.01}$	1.52 ± 0.01	1.51 ± 0.01	1.47 ± 0.01
set 2	2.00 ± 0.02	$\mathbf{1.95 \pm 0.02}$	2.12 ± 0.03	2.00 ± 0.02
set 3	1.11 ± 0.01	1.26 ± 0.01	$\mathbf{1.07 \pm 0.01}$	1.11 ± 0.01
set 4	1.42 ± 0.01	1.48 ± 0.01	1.46 ± 0.01	$\mathbf{1.42 \pm 0.01}$

Table D.8: Negative log-likelihood and cross log-likelihoods for the Cauchy distribution obtained by 1000 maximum likelihood bootstrap runs on each DJIA dataset

Dataset	Type	α	λ	E	E^*
DJIA	set 1	1.07 ± 0.01	1.37 ± 0.02	–	–
	set 2	1.07 ± 0.12	0.85 ± 0.20	–	–
	set 3	1.01 ± 0.05	2.07 ± 0.08	–	–
	set 4	1.10 ± 0.01	1.45 ± 0.02	–	–
SP500	set 1	0.99 ± 0.03	1.97 ± 0.04	1.03 ± 0.01	1.06 ± 0.01
	set 2	1.08 ± 0.01	1.52 ± 0.02	1.27 ± 0.01	1.31 ± 0.02
DAX	set 1	0.95 ± 0.02	1.60 ± 0.02	1.22 ± 0.01	1.26 ± 0.01
	set 2	1.09 ± 0.03	1.18 ± 0.05	1.50 ± 0.02	1.55 ± 0.02
IBM	set 1	1.11 ± 0.06	0.97 ± 0.10	1.70 ± 0.02	1.73 ± 0.04
	set 2	1.17 ± 0.11	0.82 ± 0.15	1.89 ± 0.03	1.90 ± 0.03
COCA	set 1	1.32 ± 0.10	0.77 ± 0.12	1.84 ± 0.03	1.82 ± 0.04
	set 2	1.18 ± 0.10	0.83 ± 0.13	1.87 ± 0.03	1.88 ± 0.03
DEMUSD	set 1	0.97 ± 0.05	2.08 ± 0.07	0.97 ± 0.02	0.97 ± 0.02
	set 2	1.02 ± 0.05	1.98 ± 0.08	1.02 ± 0.02	1.03 ± 0.02
GBPUSD	set 1	0.87 ± 0.03	2.01 ± 0.05	0.95 ± 0.02	0.96 ± 0.02
	set 2	0.96 ± 0.05	2.11 ± 0.06	0.96 ± 0.02	0.97 ± 0.02
USBONDS	set 1	1.11 ± 0.02	1.39 ± 0.03	1.35 ± 0.02	1.54 ± 0.05
	set 2	0.99 ± 0.10	2.39 ± 0.17	0.83 ± 0.02	0.96 ± 0.01
USNOTES	set 1	0.99 ± 0.09	2.46 ± 0.17	0.79 ± 0.02	0.88 ± 0.06
	set 2	0.84 ± 0.11	3.10 ± 0.29	0.45 ± 0.04	0.48 ± 0.03
SILVER	set 1	1.11 ± 0.19	0.73 ± 0.25	2.25 ± 0.08	2.29 ± 0.09
	set 2	1.07 ± 0.09	0.91 ± 0.17	1.80 ± 0.03	1.86 ± 0.08
COFFEE	set 1	1.10 ± 0.17	0.77 ± 0.24	2.18 ± 0.07	2.19 ± 0.07
	set 2	1.12 ± 0.21	0.70 ± 0.27	2.36 ± 0.10	2.37 ± 0.11

Table D.9: Estimates for the parameters α and λ for the symmetric double-sided Weibull distribution obtained by 1000 maximum likelihood bootstrap runs on each dataset with the corresponding negative log-likelihood E and cross log-likelihood E^*

Set	$E_{.\|1}$	$E_{.\|2}$	$E_{.\|3}$	$E_{.\|4}$
set 1	$\mathbf{1.37 \pm 0.01}$	1.50 ± 0.09	1.47 ± 0.02	1.37 ± 0.01
set 2	2.10 ± 0.04	$\mathbf{1.93 \pm 0.04}$	2.52 ± 0.10	2.17 ± 0.04
set 3	1.05 ± 0.01	1.32 ± 0.15	$\mathbf{0.98 \pm 0.01}$	1.04 ± 0.01
set 4	1.32 ± 0.01	1.46 ± 0.09	1.40 ± 0.02	$\mathbf{1.31 \pm 0.01}$

Table D.10: Negative log-likelihood and cross log-likelihoods for the symmetric double-sided Weibull distribution obtained by 1000 maximum likelihood bootstrap runs on each DJIA dataset

Dataset	Type	μ	λ	E	E^*
DJIA	set 1	$+0.06 \pm 0.01$	1.38 ± 0.01	1.37 ± 0.01	–
	set 2	$+0.03 \pm 0.02$	0.82 ± 0.02	1.89 ± 0.02	–
	set 3	$+0.05 \pm 0.01$	2.05 ± 0.02	0.97 ± 0.01	–
	set 4	$+0.04 \pm 0.01$	1.45 ± 0.02	1.32 ± 0.01	–
SP500	set 1	$+0.06 \pm 0.01$	1.95 ± 0.02	1.03 ± 0.01	1.05 ± 0.01
	set 2	$+0.04 \pm 0.01$	1.52 ± 0.02	1.27 ± 0.01	1.31 ± 0.02
DAX	set 1	-0.00 ± 0.00	1.61 ± 0.02	1.22 ± 0.01	1.26 ± 0.01
	set 2	$+0.07 \pm 0.02$	1.21 ± 0.02	1.51 ± 0.02	1.56 ± 0.02
IBM	set 1	-0.00 ± 0.01	1.00 ± 0.01	1.69 ± 0.01	1.71 ± 0.01
	set 2	$+0.01 \pm 0.03$	0.84 ± 0.01	1.87 ± 0.01	1.88 ± 0.02
COCA	set 1	$+0.00 \pm 0.00$	0.93 ± 0.02	1.76 ± 0.02	1.77 ± 0.01
	set 2	$+0.01 \pm 0.03$	0.85 ± 0.01	1.85 ± 0.01	1.86 ± 0.02
DEMUSD	set 1	-0.01 ± 0.01	2.07 ± 0.03	0.96 ± 0.02	0.97 ± 0.02
	set 2	-0.01 ± 0.01	1.96 ± 0.03	1.02 ± 0.02	1.02 ± 0.02
GBPUSD	set 1	$+0.00 \pm 0.00$	2.08 ± 0.04	0.96 ± 0.02	0.96 ± 0.02
	set 2	$+0.00 \pm 0.01$	2.10 ± 0.04	0.95 ± 0.02	0.95 ± 0.02
USBONDS	set 1	$+0.00 \pm 0.02$	1.34 ± 0.02	1.36 ± 0.02	1.54 ± 0.04
	set 2	$+0.01 \pm 0.01$	2.41 ± 0.04	0.82 ± 0.02	0.94 ± 0.00
USNOTES	set 1	$+0.01 \pm 0.01$	2.49 ± 0.05	0.78 ± 0.02	0.86 ± 0.04
	set 2	$+0.00 \pm 0.01$	3.65 ± 0.07	0.40 ± 0.02	0.47 ± 0.01
SILVER	set 1	$+0.06 \pm 0.04$	0.63 ± 0.01	2.16 ± 0.02	2.24 ± 0.04
	set 2	$+0.00 \pm 0.02$	0.92 ± 0.02	1.77 ± 0.02	1.84 ± 0.01
COFFEE	set 1	$+0.02 \pm 0.03$	0.67 ± 0.01	2.09 ± 0.02	2.11 ± 0.01
	set 2	$+0.02 \pm 0.03$	0.58 ± 0.01	2.24 ± 0.02	2.25 ± 0.03

Table D.11: Estimates for the parameters μ and λ for the Laplace distribution obtained by 1000 maximum likelihood bootstrap runs on each dataset with the corresponding negative log-likelihood E and cross log-likelihood E^*

Set	$E_{.\|1}$	$E_{.\|2}$	$E_{.\|3}$	$E_{.\|4}$
set 1	$\mathbf{1.37 \pm 0.01}$	1.49 ± 0.01	1.47 ± 0.02	1.38 ± 0.01
set 2	2.06 ± 0.03	$\mathbf{1.89 \pm 0.02}$	2.48 ± 0.05	2.09 ± 0.03
set 3	1.04 ± 0.01	1.29 ± 0.01	$\mathbf{0.97 \pm 0.01}$	1.03 ± 0.01
set 4	1.32 ± 0.01	1.46 ± 0.01	1.39 ± 0.02	$\mathbf{1.32 \pm 0.01}$

Table D.12: Negative log-likelihood and cross log-likelihoods for the Laplace distribution obtained by 1000 maximum likelihood bootstrap runs on each DJIA dataset

Dataset	Type	GMM $\boldsymbol{E}$	GMM $\boldsymbol{E}^*$	GLMM $\boldsymbol{E}$	GLMM $\boldsymbol{E}^*$
DJIA	set 1	1.36 ± 0.01	1.47 ± 0.01	1.37 ± 0.01	1.47 ± 0.01
	set 2	1.89 ± 0.02	2.09 ± 0.04	1.89 ± 0.02	2.04 ± 0.03
	set 3	0.96 ± 0.01	1.04 ± 0.01	0.96 ± 0.01	1.04 ± 0.01
	set 4	1.31 ± 0.01	1.40 ± 0.02	1.31 ± 0.01	1.39 ± 0.02
SP500	set 1	1.02 ± 0.01	1.06 ± 0.01	1.02 ± 0.01	1.06 ± 0.01
	set 2	1.26 ± 0.01	1.31 ± 0.02	1.27 ± 0.01	1.31 ± 0.01
DAX	set 1	1.20 ± 0.01	1.25 ± 0.01	1.20 ± 0.01	1.25 ± 0.01
	set 2	1.49 ± 0.01	1.62 ± 0.03	1.49 ± 0.01	1.57 ± 0.02
IBM	set 1	1.69 ± 0.01	1.71 ± 0.01	1.69 ± 0.01	1.70 ± 0.01
	set 2	1.85 ± 0.02	1.88 ± 0.02	1.85 ± 0.01	1.87 ± 0.02
COCA	set 1	1.76 ± 0.02	1.78 ± 0.01	1.76 ± 0.02	1.77 ± 0.01
	set 2	1.84 ± 0.01	1.85 ± 0.01	1.83 ± 0.01	1.85 ± 0.02
DEMUSD	set 1	0.96 ± 0.02	0.97 ± 0.02	0.96 ± 0.02	0.97 ± 0.02
	set 2	1.02 ± 0.02	1.02 ± 0.02	1.02 ± 0.02	1.02 ± 0.02
GBPUSD	set 1	0.96 ± 0.02	0.97 ± 0.02	0.96 ± 0.02	0.97 ± 0.02
	set 2	0.95 ± 0.02	0.96 ± 0.02	0.95 ± 0.02	0.95 ± 0.02
USBONDS	set 1	1.35 ± 0.02	1.63 ± 0.05	1.35 ± 0.02	1.58 ± 0.04
	set 2	0.80 ± 0.02	0.96 ± 0.01	0.80 ± 0.02	0.96 ± 0.01
USNOTES	set 1	0.77 ± 0.02	0.89 ± 0.03	0.78 ± 0.02	0.88 ± 0.04
	set 2	0.40 ± 0.02	0.47 ± 0.02	0.39 ± 0.02	0.47 ± 0.02
SILVER	set 1	2.15 ± 0.02	2.27 ± 0.04	2.15 ± 0.02	2.23 ± 0.03
	set 2	1.77 ± 0.02	1.85 ± 0.02	1.77 ± 0.02	1.83 ± 0.02
COFFEE	set 1	2.09 ± 0.02	2.10 ± 0.02	2.08 ± 0.02	2.10 ± 0.02
	set 2	2.24 ± 0.02	2.26 ± 0.02	2.24 ± 0.02	2.26 ± 0.02

Table D.13: Negative log-likelihoods for mixture model distributions obtained by ML on each dataset