Bayesian Extreme Quantile Regression for Hidden Markov Models

A thesis submitted for the degree of
Doctor of Philosophy

by

Antonios Koutsourelis

Supervised
by
Dr. Keming Yu
and
Dr. Antoaneta Serguieva

Department Mathematical Sciences
School of Information Systems, Computing and Mathematics
Brunel University, West London

Abstract

The main contribution of this thesis is the introduction of Bayesian quantile regression for hidden Markov models, especially when we have to deal with extreme quantile regression analysis, as there is a limited research to inference conditional quantiles for hidden Markov models, under a Bayesian approach.

The first objective is to compare Bayesian extreme quantile regression and the classical extreme quantile regression, with the help of simulated data generated by three specific models, which only differ in the error term's distribution. It is also investigated if and how the error term's distribution affects Bayesian extreme quantile regression, in terms of parameter and confidence intervals estimation. Bayesian extreme quantile regression is performed by implementing a Metropolis-Hastings algorithm to update our parameters, while the classical extreme quantile regression is performed by using linear programming.

Moreover, the same analysis and comparison is performed on a real data set. The results provide strong evidence that our method can be improved, by combining MCMC algorithms and linear programming, in order to obtain better parameter and confidence intervals estimation.

After improving our method for Bayesian extreme quantile regression, we extend it by including hidden Markov models. First, we assume a discrete time finite state-space hidden Markov model, where the distribution associated with each hidden state is a) a Normal distribution and b) an asymmetric Laplace distribution. Our aim is to explore the number of hidden states that describe the extreme quantiles of our data sets and check whether a different distribution associated with each hidden state can affect our estimation. Additionally, we also explore whether there are structural changes (break-points), by using break-point hidden Markov models. In order to perform this analysis we implement two new MCMC algorithms. The first one updates the parameters and the hidden states by using

a Forward-Backward algorithm and Gibbs sampling (when a Normal distribution is assumed), and the second one uses a Forward-Backward algorithm and a mixture of Gibbs and Metropolis-Hastings sampling (when an asymmetric Laplace distribution is assumed).

Finally, we consider hidden Markov models, where the hidden state (latent variables) are continuous. For this case of the discrete-time continuous state-space hidden Markov model we implement a method that uses linear programming and the Kalman filter (and Kalman smoother).

Our methods are used in order to analyze real interest rates by assuming hidden states, which represent different financial regimes. We show that our methods work very well in terms of parameter estimation and also in hidden state and break-point estimation, which is very useful for the real life applications of those methods.

Contents

List of Figures

List of Tables

Acknowledgements

After three years of research at Brunel University, which resulted on the completion of my thesis, I would like to thank my supervisor Dr. Keming Yu for his guidance, advice and encouragement in my work through our scientific discussions, which also provided me with new statistical ideas.

I am also grateful to my second supervisor, Dr. Antoaneta Serguieva, for her support throughout the three years of my research.

Special thanks goes to my family, who have been supporting me in many ways since I started my studies in Greece and then in UK.

Many thanks to all my friends, both in Greece and UK, who supported me in their own way.

Finally, deep thanks to all the stuff within the School of Information Systems, Computing and Mathematics of Brunel University.

Declaration Of Authorship

I declare that the work presented in this thesis is my original research and has not been presented for a higher degree at any other university or institute.

.....

Antonios Koutsourelis

Author's Publications

1. Koutsourelis, A. and Yu, K. (2011). Hidden Markov Model Extreme Quantile Regression and Applications. *Working paper*.

2. Koutsourelis, A. and Yu, K. (2011). Extreme Quantile Regression for Break-Point Hidden Markov Models and Applications. *Working paper*.

Chapter 1

Introduction

Time series data occur frequently in many applications. To analyze such data, it is of great practical importance to select an appropriate model. Hidden Markov Models (HMMs) have been successfully applied to various fields due to their ability to describe many processes in a mathematically tractable way. The specification of a hidden Markov model assumes that an underlying sequence of states, which follows a finite Markov chain, affects the distribution of the observed process of interest. Additionally, it is also very important to select an appropriate regression method to analyze time series data. Traditional mean regression developed for Gaussian models does not typically account for the heterogeneity encountered in time series data, which usually consists of an often loose association of different time series into one data set (Nuamah, 1986; Beck and Katz, 2007), whereas quantile regression can provide a broader statistical alternative to least squares in the real word of research. Quantile regression offers the possibility of investigating how covariate effects influence the location, scale and possibly the shape of the conditional response distribution. It has a random coefficient interpretation, allowing for slope heterogeneity drawing from non-Gaussian distributions.

Quantile regression can be used to measure the effect of covariates not only in the centre of a distribution, but also in the upper and lower tails. These tail measurements have many applications (Yu *et al*, 2003). For example, risk management regulations require banks to estimate market risk measures based on quantiles of loss distributions. Value at Risk (VaR) is to be calculated daily, using a 99th or 95th percentile, one-tailed confidence interval by banks. Extreme quantile regression has been proposed to model these tails of underlying distributions. Like mean regression models, quantile regression models also involve parameter uncertainty, so Bayesian quantile regression have

attracted much interests in literature,especially during the past ten years, since Yu and Moyeed (2001). However, most of the research on the topic focus on general quantile regression without specification on extreme quantile regression, while the latter needs special treatment.

During recent years interest lied in estimating time-varying quantiles, which can be fitted to a sequence of observations by formulating a time series model for the corresponding population quantile and iteratively applying a suitably modified state space signal extraction algorithm. Those quantiles can provide information about dispersion, asymmetry, VaR and many other aspects of a time series. De Rossi and Harvey (2006) noted that the criterion for choosing the estimated quantile so as to minimize the sum of absolute values around it, can be obtained from a model, where the observations are generated by an asymmetric double exponential distribution. Their method was based on the Kalman filter and smoother, in order to estimate the quantiles. De Rossi and Harvey (2009) used a model-based approach, which enabled time-varying quantiles to be used for forecasting and they proved that if the underlying time series model is a Wiener process, then the solution for quantiles is equivalent to fitting a spline. Moreover, Gerlach *et al*. (2011) extended the CAViaR model for dynamic quantile estimation to a fully nonlinear family. They performed Bayesian time-varying quantile forecasting for VaR by employing Bayesian adaptive Markov Chain Monte Carlo (MCMC) methods, which are based on the well-known link between the quantile criterion function and the asymmetric Laplace distribution.

Generally, Bayesian inference for hidden Markov models (Castellano and Scaccia, 2007) and quantile regression based on hidden Markov models (Farcomeni, 2010) have been used widely in practice. However, this thesis combines both methodologies in order to model complicated structures of real world applications. This combination produces a novel research, which takes into account both conditional quantiles of the response variable and the hidden (latent) states of the underlying Markov process.

In this thesis we use discrete-time finite state-space hidden Markov models and quantile regression through a Bayesian approach, in order to model financial time series. First, we perform Bayesian extreme quantile regression to three simulated data sets and one real data set using a Metropolis-Hastings algorithm to update our parameters. Our aim is to see whether there is a fast convergence and a good estimation of each parameter. Then we compare Bayesian extreme quantile regression with the classical approach, which uses linear programming for parameter estimation. The fact that we use simulated data sets enable us to compare the estimated parameters, for both Bayesian extreme quantile regression and classical approach, with the true values of the parameters. Additionally, the fact that

the difference between the simulated data sets is only the error term in the initial models enables us to explore if and how the error term affects the parameter estimation, for various quantiles (especially for the extreme quantiles). Finally, we can also check the similarities or differences between Bayesian extreme quantile regression and classical approach in terms of different quantiles.

Inference for hidden Markov models belongs to the general class of missing data problems. Missing data either arise naturally (when data that have been observed are missing), or intentionally (random variables that are not observable). Clearly, hidden Markov models is an example of the latter. In missing data problems the likelihood function is not tractable due to the existence of some unobserved data. In the Bayesian framework, such models are analyzed via MCMC utilizing the technique of data augmentation. This technique introduces additional (latent) variables to the parameter space so that the likelihood function becomes tractable.

We continue by using hidden Markov models and Bayesian extreme quantile regression, in order to analyze two real financial data sets. The complexity of a hidden Markov model is related to several aspects, such as the number of hidden states in the model and the connectivity of the states in the transition matrix. A simple special case of hidden Markov models, which is widely used in the econometrics literature, is the class of break-point models. These are used to model the occurrence of one or multiple structural breaks (changes) in a time series. Treating the model parameters as unknown variables leads to complex posterior distributions, or integrals that need to be calculated. Therefore, approximation algorithms are introduced to address any computational intractability. MCMC methods can be applied in that case in order to enable us simulate from complex posterior distributions.

In this thesis we implement a Bayesian approach to inference for two different hidden Markov models (Normal hidden Markov models and asymmetric Laplace distribution hidden Markov models), based on data augmentation for the extreme quantiles. The latent variables, in our case, consists of a sequence of hidden states which are modeled by a finite state-space Markov chain. We construct an MCMC algorithm which consists of updates of the hidden sequence of states and the model parameters. The hidden states are updated given the model parameters using the Forward-Backward algorithm, while the model parameters are updated given the states via simple Gibbs steps (for the Normal hidden Markov model) and via a mixture of Gibbs and Metropolis-Hastings steps (for the asymmetric Laplace distribution hidden Markov model).

The data we consider concern the US ex-post real interest rates and the US treasury bill real interest rates. We are interested in inferring the number of states, or the number of break-points, that

describe the conditional quantiles of the data in the best possible way. For this purpose we consider various hidden Markov models and break-point models for each data set, which we compare using the deviance information criterion (DIC) in order to select the best model. Finally, it is of interest to infer the dates of the structural changes in the series, since they represent shifts between different economic regimes or changes in the economic environment. It is also important to check if there is a fast convergence and a good estimation of our parameters, for both extreme quantiles, for the two different hidden Markov models. Then, we compare our MCMC algorithms in terms of parameter estimation, estimated number of hidden states and estimated number and dates of break-points.

1.1 Advantages of a Bayesian Approach

On a theoretical level, many classical and standard models (for example, Hayashi, 1982; Abel and Eberly, 1994) explain and predict that the real interest rate should have a major impact on investment. However, on a practical level, this influence is hard to be found and measured. Under a Bayesian approach for hidden Markov models, though, we can use hidden states to represent various factors, such as price segmentation, segments of transactions, unobserved instantaneous volatility, jump intensity, information flow and financial regimes. In this way we can tackle the complexity of the real world financial applications.

During recent years, the development of new and advanced MCMC methods, combined with the vast increase in computing power, has made Bayesian approaches to inference feasible, easier and more attractive. Additionally, and most important, these techniques offer various significant advantages compared to the classical methods.

First, they enable us to include certain model parameters into the economically sensible range by defining their prior distributions. They also treat both model parameters and latent variables as random variables, which have a joint distribution with the observed variables. Moreover, Bayesian methods are more robust and reliable than the classical methods, particularly when we deal with the evaluation of the likelihood of an observed time-series. Finally, Bayesian methods provide us with exact confidence intervals for the parameters and, on some occasions, for functions of the parameters as well.

1.2 Literature Review : HMMs in Financial Econometrics

Hidden Markov models have been used in many applications in financial econometrics. One of those applications is modeling financial prices. Financial prices usually show non linear dynamics, which are often due to the existence of two or more regimes within which returns and/or volatilities display different behavior. Using these models, Rydén *et al.* (1998), reproduce most of the stylized facts about daily series of returns while Rossi and Gallo (2006) provide accurate estimates of stochastic volatility. Engel and Hamilton (1990) model segmented time-trends in the US dollar exchange rates via HMMs. Robert *et al.* (2000) use HMMs to study daily returns of the S&P index, assuming the existence of different regimes characterized by different levels of volatility.

Moreover, hidden Markov models have been successfully applied in modeling option pricing or modeling defaults within a bond portfolio and prediction of financial time series can be achieved using Hidden Markov models. The special case of break-point models have been even more widely applied in the econometrics literature (for example, see Bai, J. and Perron, P. (2002)). In this thesis we give a brief overview of the econometrics literature related to hidden Markov and break-point models.

The option pricing theory has been intensively studied since the works of Black and Scholes (1973) and Merton (1973). In Black and Scholes model, the underlying asset price process is described by geometric Brownian motion in which the drift and volatility are assumed to be deterministic. However, the volatility in asset price processes in the financial market would depend on the past information. The phenomenon known as volatility smile occurs when the volatility, which is obtained when the market price of European call option is equated with the Black and Scholes model, is not constant but varying with respect to the time to maturity and strike price of option. Due to this phenomenon, several models were introduced to characterize the volatility dynamics, such as ARCH (Engle, 1982), GARCH (Bollerslev, 1986) and others introduced by Hull and White (1987) and Heston (1993).

Ishijima and Kihara (2005) combined the above models and managed to derive an analytic formula for pricing European call option, when asset price processes are subject to hidden Markov models, under the setting of an n-state hidden Markov model in discrete-time framework. Their hidden Markov model was specified by a state equation with the time-homogeneous transition probability matrix and an observation equation which describes asset prices by the log-normal model in which

both drift and volatility parameters switch according to the state.

Apart from the previous formula, they also estimated their model by using the Baum-Welch algorithm with scaling in computing Forward-Backward probabilities. They applied their model on the Japanese financial market data in order to estimate the parameters of the model. They found that their formula, compared to the existing option pricing models which characterize stochastic volatility in asset prices, had mainly three advantages. It is an analytic formula with meanings easy to interpret and it enables us to capture the persistence of volatility in the risky asset prices.

Interaction effects are an important component of portfolio credit risk, but finding a way to quantify these effects is considered to be a controversial issue. Especially for large portfolios it is generally unfeasible to model the default risk of each individual issuer and the correlation with other issuers, as this leads to a high-dimensional model with a large number of parameters, which can not be reliably estimated.

Several models which describe the previous interaction process in a more simple way have been proposed. These models have a small number of parameters, but they can be described as static models, in a way that they only concern the total number of defaults in a specific period. Furthermore, for some applications, the timing of defaults is as important as their total number and a dynamic model is needed. The enhanced model was defined by Davis and Lo (2001) as a dynamic version of infectious defaults. According to that model, the portfolio is assumed to be in one of the following states; normal risk or enhanced risk. It starts in normal risk and when a default occurs it moves to enhanced risk. The portfolio stays in that state for an exponentially-distributed random time before going back to normal risk.

Davis, Giamperi and Crowder (2005) considered a simplified enhanced model with two, not directly observed, states corresponding to normal and enhanced risk. They simply considered a hidden Markov model, and they supposed that the hidden variable is a two-state Markov process in discrete time and it is not depending on the default events. Within each time period defaults are supposed to be binomially distributed, with higher mean in the enhanced risk state. They found that their model has good explanatory power, despite the fact that it is very simple. They managed to obtain estimates for the model parameters and reconstruct the most likely sequence of the risk state. Moreover, by extending that model to include independent hidden risk sequences, they could disentangle the risk associated with the business cycle from that specific to individual sector.

Zang (2001) applied a hidden Markov model, rather than a model based on regression equations,

in order to analyze and predict financial time series. That model addressed two of the most important challenges of financial time series modeling; non-stationarity and non-linearity. Zang (2004) extended the hidden Markov model to include a novel exponentially weighted EM algorithm to handle these challenges. He found that this extension enables us to model both sequence data and dynamic financial time series. Based on the exponentially weighted EM algorithm, he proposed a double weighted EM algorithm which is able to adjust training sensitivity automatically and solves the drawback of the previous EM algorithm caused by over-sensitivity. Additionally, he managed to show that both EM algorithm can be written in a form of exponentially moving averages of the model variables of the hidden Markov model. This allows us to take advantage of the existing technical analysis techniques.

The statistics and econometrics literature contain a vast amount of work concerning structural changes, while most of them are designed to investigate the case of a single break-point. However, there exist many studies which are related to hypothesis testing in the context of multiple changes including Andrews, Lee and Ploberger (1996), Garcia and Perron (1996), Liu, Wu and Zidek (1997), Pesaran and Timmermann (1999), Lumsdaine and Papell (1997) and Morimune and Nakagawa (1997). General discrete-time finite state-space hidden Markov models have been also used in econometric applications.

Typically, the break-point model is specified through a hierarchical specification in which, for every time point, the probability distribution of a break-point given the previous break-points is modeled first. The next step is to process the parameters in the current regime, given the current break-points and previous parameters. Finally, the data is generated, given the parameters and the break-points. Chernoff and Zacks (1964) proposed a special case of this model in which there is a constant probability of a break at each time point (not dependent on the history of break-points). Yao (1984) specified the same model for the break-points, but assumed that the joint distribution of the parameters is exchangeable and independent of the break-points. Similar exchangeable models for the parameters have been studied by Carlin et al. (1992) in the context of a single break-point and by Inclan (1993) and by Stephens (1994) in the context of multiple break-points.

Chib (1998) showed, as an extension, that it is possible to fit models in which the probability of a break-point is not a constant but depends on the regime. In this case, the probability distribution of the break-points is characterized by a set of parameters and not by just one parameter. His approach was based on a formulation of the break-point model in terms of an unobserved discrete state variable that indicates the regime from which a particular observation has been drawn. This state variable is specified to evolve according to a discrete-time discrete-state Markov process with the transition

probabilities constrained so that the state variable can either stay at the current value or jump to the next higher value. This parametrization exactly reproduces the break-point model and, additionally, MCMC simulations of this model are straightforward and improve on existing approaches in terms of computing time and speed of convergence.

1.3 Structure of the Thesis

Chapter 2 starts with describing quantile regression. Then, there is a brief comparison between quantile regression and mean regression, in order to highlight their differences. This is followed by a short reference and the definition of the extreme regression quantiles and also a short literature review on Bayesian inference on quantile regression. The second, and most important, part of this chapter describes Bayesian inference for extreme quantiles using a suitably chosen τ-quantile loss function. It also explains how this Bayesian method corresponds to solving minimization problems and, as a consequence, has a strong connection with linear programming. Chapter 2 continues with applying Bayesian extreme quantile regression on three simulated data sets and one real data set and comparing that method with the classical extreme quantile regression, which uses linear programming. Finally, based on the comparison of those two methods, it introduces a way to perform Bayesian extreme quantile regression, by combining MCMC methods and linear programming, in order to obtain better and more accurate results.

Chapter 3 presents hidden Markov models' history and applications, but focuses and describes in a more specific way the discrete-time finite state-space hidden Markov model. Then, this chapter makes a reference to some other types of hidden Markov models, but it focuses on the break-point models. This is followed by a reference to the three basic problems, that appear when hidden Markov models are used in real-world applications, and a short description of their solutions. The final part of the chapter mentions the three useful algorithms used within those solutions and presents the Forward-Backward algorithm for discrete-time finite state-space hidden Markov models, with application to the m-state hidden Markov model.

Chapter 4 describes analytically Bayesian inference for extreme quantiles using hidden Markov models. It focuses on the m-state Normal hidden Markov model (and the corresponding Normal break-point hidden Markov model) and the m-state asymmetric Laplace distribution (ALD) hidden Markov model (and the corresponding ALD break-point hidden Markov model). It explains in detail how the distribution associated with the Markov chain is used within the Forward-Backward algo-

rithm, in order to obtain the likelihood, as a sum of the forward variables (and not as a sum over all the possible sets of states) and the hidden states at time t. This chapter also explains in detail how to construct MCMC algorithms to update our parameters. It is clearly shown why and how Gibbs sampling is used in the case of the Normal hidden Markov model (and the Normal break-point hidden Markov model) and why and how a mixture of Gibbs sampling and Metropolis-Hastings sampling is used in the case of the ALD hidden Markov model (and the ALD break-point hidden Markov model). After every method there are applications on real data sets. These applications enable us to explore how our method (Bayesian extreme quantile regression) is affected by using different hidden Markov models, in terms of estimation (parameters, hidden states, number and dates of break-points) and Markov chain convergence within the MCMC algorithm. Finally, chapter 4 refers to the deviance information criterion (DIC), which is used in this thesis as a Bayesian model comparison criterion.

Chapter 5 starts with a brief description of a hidden Markov model, where the underlying hidden state is continuous. Then, it describes analytically the Kalman filter, which is a very good and reliable method to perform extreme quantile regression, when the latent variables of the assumed hidden Markov model are continuous. This chapter provides information about the computational origins of the algorithm and, finally, it describes the Kalman smoothing algorithm, which improves the parameter estimation obtained by the Kalman filter. In the end there are applications on two real data sets.

Chapter 6 presents the model comparison of our Bayesian extreme quantile regression methods for hidden Markov models. It also describes various general algorithms, which helped us implementing our methods.

Chapter 7 concludes this thesis by summarizing and discussing the main results in relation to the objectives of this thesis. Additionally, it proposes recommendations for possible future research directions.

Examples of the MCMC algorithms we used in this thesis and examples of density plots and traceplots of the models' parameters are included in the Appendix. There is also additional and useful information, in order to help the understanding of various points of our methodology. This includes information about linear programming (LP), properties of extreme quantile regression (enabling us to compute approximate confidence intervals), Dirichlet distribution, Metropolis-Hastings algorithm, Gibbs sampling and Kalman filter.

Chapter 2

Bayesian Extreme Quantile Regression

2.1 Quantile Regression

Let us consider the standard regression model:

$$y_i = \mathbf{x}_i'\boldsymbol{\beta} + u_i, \tag{2.1}$$

for $i = 1, 2, ..., n$, where x_i are the independent variables, y_i are the values of the dependent variable $\mathbf{Y}$, u_i is the error term with zero mean and $\boldsymbol{\beta}$ are the parameters to be estimated.

By performing regression analysis we can model the relationship between a response variable and predictor variables. But, in real applications, the response variable cannot be predicted exactly from the predictor variables. That is why we summarize the behavior of the response variable by using measures of central tendency, like the mean (average value), the median (middle value) and the mode (most likely value). Simple regression analysis is focused on the mean, which means that the relationship between the response variable and predictor variables is summarized by describing the mean of the response, for each fixed value of the predictors, using a function (conditional mean function) of the response. Models based on the previous analysis (conditional mean models) are capable of providing a complete description of the relationship mentioned before, under ideal conditions. Those models also lead to estimators (least-squares estimation) easy to calculate and interpret.

However, the mean regression methodology has some limitations, which make it very difficult for researchers to study the properties of the whole distribution, due to the fact that this methodology cannot be extended to noncentral locations. Most of the times, noncentral locations is where the

interest of the analysis lies. For instance, sometimes we are not interested in mean returns, or we may be interested in mean returns above which are 90% of our data. This happens because model assumptions are not always met in real world applications and as a consequence, the conditions are not ideal for obtaining a complete description of the response and the predictors. Additionally, this kind of models are heavily influenced by outliers and in social phenomena heavy-tailed/skewed distributions commonly occur and lead to outliers. For that reason, other regression methods were developed, in order to overcome those problems. Quantile regression offers a more complete statistical approach and now has widespread applications.

Quantile regression is a statistical technique for estimating and conducting inference about conditional quantile functions. It models the relationship between a set of predictor variables and specific quantiles of the response variable. This allows us to see and compare how some quantiles of the response variable may be more affected by the predictor variables, than other quantiles (for instance, we can check if large values of the response variable are more affected than the lower values). This is what makes quantile regression models more robust to outliers, than the linear regression (mean regression) models. Quantile regression, which is a natural extension of the linear regression, enables us to see how the conditional mean of Y depend on the covariates at each quantile. As a result, we have a more complete view, than in standard regression, of how the conditional distribution of Y, given $\mathbf{X} = \mathbf{x}$, depend on $\mathbf{x}$. Also, with quantile regression we can explore potential effects on the shape of the distribution and not only on the location or the scale of the distribution. Standard regression only models the average relationship between the response variable and the covariates, whereas quantile regression describes that relationship for a range of values of τ (quantiles). In that way, an approximation of the full response probability distribution can be produced.

There are various important applications of quantile regression, which involve the study of extremal phenomena. In econometrics, this kind of phenomena are the analysis of factors which contribute to extremely low infant birth-weights (Abrevaya, 2001), the analysis of the highest bids in auctions (Donald and Paarsch, 1993) and estimation of factors of high risk in finance (Tsay, 2002 and Chernozhukov and Umantsev, 2001). In biostatistics and other areas our interest lies in the analysis of survival at extreme durations (Koenker and Geling, 2001), the analysis of factors that impact the approximate boundaries of biological processes (Cade, 2003), image reconstruction and other problems where conditional quantiles near maximum or minimum are of interest (Korostelev, Simar and Tsybakov, 1995). Details of quantile regression applications could be also found in the paper "Quantile regression : applications and current research areas", (Yu, Lu and Stander, 2003).

Very often time series of financial asset values exhibit well known statistical features, such as heavy tails. Additionally, in various applications (climate research, medicine, insurance and finance) interest lies on estimating extremal risk measures, such as record values, return periods and high level crossings. In these situations the usage of extreme quantiles is more sufficient than using 10% or 90% quantiles.

In order to define extreme regression quantiles, let us re-consider the regression model:

$$y_i = \mathbf{x}_i' \boldsymbol{\beta} + u_i, \;\; \mathbf{x}_i \in \mathbf{R}^p, \;\; i = 1, ..., n, \tag{2.2}$$

where $u_i \overset{iid}{\sim} f(u)$ and the first coordinate of $\mathbf{x}_i$ is 1. We may assume, without loss of generality, that $\sum_{i=1}^{n} x_{ij} = 0$, $j = 2, ..., p$, to simplify calculations. Then, we define the regression quantiles, $\hat{\beta}(\tau)$, for $0 < \tau < 1$, to satisfy the following optimization problem :

$$\min_{b \in R^p} \sum_{i=1}^{n} \rho_\tau (y_i - \mathbf{x}_i' \boldsymbol{\beta}),$$

$$\rho_\tau(u) = \tau \; u \; I_{(0,\infty)}(u) - (1 - \tau) \; u \; I_{(-\infty,0)}(u) \quad (lossfunction),$$

or

$$\rho_\tau(u) = \tau \; u^+ - (1 - \tau) \; u^-, \;\; u \in \mathbf{R},$$

with u^+ and u^- denoting the positive and negative parts of u. The extreme regression quantiles correspond to the cases of $\tau = 0$ and $\tau = 1$, which means that $\rho_0(u) = -u^-$ and $\rho_1(u) = u^+$, $u \in \mathbf{R}$. So, $\hat{\boldsymbol{\beta}}(0)$ and $\hat{\boldsymbol{\beta}}(1)$ are respective solutions of the following minimization problems :

$$\min_{b \in R^p} \{ - \sum_{i=1}^{n} (y_i - \mathbf{x}_i' \boldsymbol{\beta})^- \}$$

and

$$\min_{b \in R^p} \sum_{i=1}^{n} (y_i - \mathbf{x}_i' \boldsymbol{\beta})^+.$$

Bayesian inference on quantile regression has attracted much interest recently. A few of the different models and sampling algorithms for Bayesian quantile regression include MCMC (Markov chain Monte Carlo) or RJMCMC (Reversible Jump Markov Chain Monte Carlo) methods via an asymmetric Laplace (AL) distribution for the likelihood function (Yu and Moyeed, 2001; Yu and Stander, 2007; Chen and Yu, 2008; Tsionas, 2003; Geraci and Bottai, 2007; Liu and Bottai, 2009), Dirichlet process mixing based nonparametric zero median distribution for the regression model error (Kottas and Gelfand, 2001), an MCMC algorithm using Jeffrey's (Jeffrey, 1961) substitution posterior

for the median (Dunson and Taylor, 2005), the expectation-maximizing (EM) algorithm using the AL distribution (Geraci and Bottai, 2007), the empirical likelihood based algorithm (Lancaster and Jun, 2008), the mixture distribution algorithm (Reich, Bondell and Wang, 2010) and Gibbs sampling (Tsionas, 2003; Kozumi and Kobayashi, 2009; Reed and Yu, 2009). Li, Xi and Lin (2010) even study regularization in quantile regressions from a Bayesian perspective, Reed, Dunson and Yu (2010) discuss Bayesian variable selection for quantile regression and Reich, Fuentes, and Dunson (2010) proposes Bayesian spatial quantile regression. However, there is no research on Bayesian inference for extreme quantiles.

2.2 Bayesian Inference for Extreme Quantiles

A random variable U follows the asymmetric Laplace distribution, when its probability density is given by

$$f_\tau(u) = \tau(1-\tau)\exp\left\{-p_\tau(u)\right\},$$

where $p_\tau(u)$ is the loss function and $0 < \tau < 1$.

The τ^{th} conditional quantile of y_i, given $\mathbf{x}_i$, is denoted as $q_\tau(y_i|\mathbf{x}_i) = \mathbf{x}_i'\boldsymbol{\beta}(\tau)$, where $\boldsymbol{\beta}(\tau)$ is a vector of coefficients dependent on τ. When we are interested in $q_\tau(y_i|\mathbf{x}_i)$, we can assume that : (i) $f(\mathbf{y};\mu_i)$ is asymmetric Laplace distribution (ALD) and (ii) $g(\mu_i) = \mathbf{x}_i'\boldsymbol{\beta}(\tau) = q_\tau(y_i|\mathbf{x}_i)$, $0 < \tau < 1$. Given the observations $y = (y_1, y_2, ..., y_n)$, then the posterior distribution of β is given by

$$\pi(\boldsymbol{\beta}|\mathbf{y}) \propto L(\mathbf{y}|\boldsymbol{\beta})\pi(\boldsymbol{\beta}),$$

where $\pi(\boldsymbol{\beta})$ is the prior distribution of $\boldsymbol{\beta}$ and $L(\mathbf{y}|\boldsymbol{\beta})$ is the likelihood function, given by

$$L(\mathbf{y}|\boldsymbol{\beta}) = \tau^n(1-\tau)^n \exp\left\{-\sum_{i=1}^{n} p_\tau(y_i - \mathbf{x}_i'\beta)\right\}, \tag{2.3}$$

where $p_\tau(z)$ is the loss function (as defined in the previous section). It is important to say that we can use any prior distribution, $\pi(\boldsymbol{\beta})$, but when there is not any realistic information we can use improper uniform prior distributions for $\boldsymbol{\beta}$, because the joint posterior distribution will be proper (Yu and Moyeed, 2001). This form of likelihood is very useful because the minimization of the loss function is equivalent to the maximization of that likelihood function, which is formed by combining independently distributed asymmetric Laplace densities. So, the estimation of q_τ of a random variable Y is equivalent to the estimation of the location parameter μ of an asymmetric Laplace distribution,

with density

$$f_\tau(y) = \tau(1-\tau)exp\left\{-p_\tau(y-\mu)\right\}.$$

So, the likelihood function we described before (equation 2.3), is a combination of n independently distributed asymmetric Laplace distributions with a location parameter $\mu_i = \mathbf{x}_i'\boldsymbol{\beta}$.

Bayesian methods can be also used in order to estimate the parameters in extreme quantile regression. This means estimating the parameters $\boldsymbol{\beta}(\tau)$, when $\tau \to 0$, or $\tau \to 1$. In Bayesian extreme quantile regression, our interest lies on $\hat{\boldsymbol{\beta}}(0)$ and $\hat{\boldsymbol{\beta}}(1)$, and as we said in a previous section, they are respective solutions of the following minimization problems:

$$\min_{b\in R^p}\{-\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^-\}$$

and

$$\min_{b\in R^p}\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^+,$$

where

$$\rho_1(u_i) = \rho_1(y_i - \mathbf{x}_i'\boldsymbol{\beta}) = (y_i - \mathbf{x}_i'\boldsymbol{\beta})^+ = |y_i - \mathbf{x}_i'\boldsymbol{\beta}| \cdot I(y_i \geq \mathbf{x}_i'\boldsymbol{\beta}),$$

$$\rho_0(u_i) = \rho_0(y_i - \mathbf{x}_i'\boldsymbol{\beta}) = -(y_i - \mathbf{x}_i'\boldsymbol{\beta})^- = |y_i - \mathbf{x}_i'\boldsymbol{\beta}| \cdot I(y_i \leq \mathbf{x}_i'\boldsymbol{\beta}).$$

As a consequence, $\hat{\boldsymbol{\beta}}(1)$ is given by

$$\min_{b\in R^p}\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^+ =$$

$$= \max_{b\in R^p} exp\left\{-\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^+\right\}.$$

Therefore, the likelihood function $L(\mathbf{y}|\boldsymbol{\beta})$ for Bayesian inference for $\boldsymbol{\beta}(1)$ is given by

$$L_1(\mathbf{y}|\boldsymbol{\beta}) = e^{-\sum_{i=1}^{n} u_i\ I(u_i \geq 0)}.$$

In the same way, $\hat{\boldsymbol{\beta}}(0)$ is given by

$$\min_{b\in R^p}\{-\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^-\} =$$

$$= \max_{b\in R^p} exp\left\{\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\boldsymbol{\beta})^-\right\}.$$

The likelihood function $L(\mathbf{y}|\boldsymbol{\beta})$ for Bayesian inference for $\boldsymbol{\beta}(0)$ is given by

$$L_2(\mathbf{y}|\boldsymbol{\beta}) = e^{\sum_{i=1}^{n} u_i\ I(u_i \leq 0)}.$$

2.2.1 Prior selection

In order to perform Bayesian extreme quantile regression via MCMC algorithms, we have to choose the appropriate prior distribution for our parameters. In the following sections we will deal with three simulated data sets and one real data set and we decide to use uninformative prior distributions for our parameters $\boldsymbol{\beta}(\tau)$, where $\tau \to 0$, or $\tau \to 1$ (in our MCMC algorithm we use $\tau = 0.001$ and $\tau = 0.999$, in order to approximate 0 and 1, respectively).

Based on our results, in terms of estimation and Markov chain convergence, we do not have any evidence of needing to use informative prior distributions. Additionally, our aim is not to see how the estimation is affected by using different kinds of prior distributions, but to see if and how different error distributions of our model affect the estimation of the parameters. However, in other cases, using priors which do not contain much information leads to bad results or a slow convergence of the Markov chain. That means that priors which contain some kind of information must be used. But how can we obtain informative prior distributions without having any prior knowledge / information?

We suggest that this kind of informative prior distributions can be obtained and used in a MCMC algorithm in the following way: we first use a MCMC algorithm, which assumes improper uniform priors for the parameters, in order to estimate a large number of simulated values of $\boldsymbol{\beta}(\tau)$ and obtain its posterior distributions $pr(\boldsymbol{\beta}(\tau)|\mathbf{y})$. Then, these posterior distributions are used as prior distributions in a new MCMC algorithm, which will have better results and faster convergence. However, the posterior distributions of $\boldsymbol{\beta}(\tau)$ or the $pr(\boldsymbol{\beta}(\tau)|\mathbf{y})$ may not follow a simple parametric distribution, but our much empirical experience found that a mixture of normal distributions fits it well. Actually, the following eight mixture of normal distributions, which have different properties, cover most of posterior distributions well for our purpose. Plots of these densities are in the Appendix E, figure 7.1.

- Gaussian (Gau): $\mathrm{N}\left(0, 1^2\right)$,
- Skewed (Skew): $\frac{1}{5}\mathrm{N}\left(0, 1^2\right) + \frac{1}{5}\mathrm{N}\left(\frac{1}{2}, (\frac{3}{2})^2\right) + \frac{3}{5}\mathrm{N}\left(\frac{13}{12}, (\frac{5}{9})^2\right)$,
- Kurtotic (Kur): $\frac{2}{3}\mathrm{N}\left(0, 1^2\right) + \frac{1}{3}\mathrm{N}\left(0, (\frac{1}{10})^2\right)$,
- Outlier (Outl): $\frac{1}{10}\mathrm{N}\left(0, 1^2\right) + \frac{9}{10}\mathrm{N}\left(0, (\frac{1}{10})^2\right)$,
- Bimodal (Bim): $\frac{1}{2}\mathrm{N}\left(-1, (\frac{2}{3})^2\right) + \frac{1}{2}\mathrm{N}\left(1, (\frac{2}{3})^2\right)$,
- Bimodal, separate modes (Sepa): $\frac{1}{2}\mathrm{N}\left(-\frac{3}{2}, (\frac{1}{2})^2\right) + \frac{1}{2}\mathrm{N}\left(\frac{3}{2}, (\frac{1}{2})^2\right)$,

- Skewed bimodal (Skeb): $\frac{3}{4}\mathrm{N}\left(0, 1^2\right) + \frac{1}{4}\mathrm{N}\left(\frac{3}{2}, (\frac{1}{3})^2\right)$,
- Trimodal (Tri): $\frac{9}{20}\mathrm{N}\left(-\frac{6}{5}, (\frac{3}{5})^2\right) + \frac{9}{20}\mathrm{N}\left(\frac{6}{5}, (\frac{3}{5})^2\right) + \frac{1}{10}\mathrm{N}\left(0, (\frac{1}{4})^2\right)$.

Whichever prior distribution is assumed will be proper as a mixture of normal distributions. Therefore, the posterior distribution $pr(\boldsymbol{\beta}(\tau)|\mathbf{y})$, under extreme quantile regression, is proper. In a situation where we need to use informative prior distributions we perform a model fit, given the data we have, in order to choose the most appropriate distribution (of those mentioned above) and also choose the appropriate parameters for those priors.

To sum up, the method we mentioned above is not necessary in this thesis, as we have very good results by using improper priors. So, in the following sections we will use only improper prior distributions for our parameters. However, there are situations where improper prior distributions do not lead to good estimations. In those cases it would be very useful to use that method.

2.2.2 Computation

For the model given by the equation 2.2, we consider an exponential probability density based likelihood of the form:

$$L_1(\mathbf{y}|\boldsymbol{\beta}) = e^{-\sum_{i=1}^{n} u_i \ I(u_i \geq 0)}.$$

Then, obtaining $\hat{\boldsymbol{\beta}}(1)$ is equivalent to obtaining the maximum likelihood estimator (MLE) of the previous likelihood function. Note that

$$L_1(\mathbf{y}|\boldsymbol{\beta}) = \lim_{\tau \to 1^-} e^{-\sum_{i=1}^{n} \rho_\tau(u_i)}.$$

Following the same method, considering the following likelihood function

$$L_2(\mathbf{y}|\boldsymbol{\beta}) = e^{\sum_{i=1}^{n} u_i \ I(u_i \leq 0)},$$

we can obtain $\hat{\boldsymbol{\beta}}(0)$ as the MLE of that likelihood function. Note that

$$L_2(\mathbf{y}|\boldsymbol{\beta}) = \lim_{\tau \to 0^+} e^{-\sum_{i=1}^{n} \rho_\tau(u_i)},$$

where $\rho_\tau(u_i)$ is the loss function described in section 2.1.

Independent improper uniform priors for all the components of $\boldsymbol{\beta}$ are used in the following examples and by performing a Metropolis-Hastings algorithm, each of the parameters was updated based on a Gaussian proposal density centered at the current state of the chain. The asymmetric Laplace

distribution was used, in order to model the Bayesian extreme quantile regression parameters and there is a fast convergence of the Markov chains in all the examples.

2.2.3 Simulated Data

We apply Bayesian extreme quantile regression on three simulated data sets, which differ in the distribution of the error term ϵ_t, $t = 1, ..., n$. The distributions we use are : Uniform, Beta and Weibull.

The Metropolis-Hastings algorithm, mentioned in the previous section, is used and a burn-in period of 2000 iterations was excluded, so a sample of 8000 values from the posterior distribution of each of the elements of $\boldsymbol{\beta}$ was collected, in order to get our results. Then we compare our parameter estimations with the true values of the parameters. The true values are given by :

$$\beta_0(\tau) = 1 + F^{-1}(\tau),$$

$$\beta_1(\tau) = 1,$$

where $F^{-1}(\tau)$ is the quantile function (inverse cumulative distribution function) of our distributions. For more details see Appendix A.

We will simulate observations from the model

$$y_t = \boldsymbol{\beta}\mathbf{x}_t' + \epsilon_t, \quad t = 1, 2, ..., n$$

or

$$y_t = \beta_0 + \beta_1 x_{1t} + \epsilon_t, \quad t = 1, 2, ..., n$$

assuming that $\mathbf{x}_t \sim Beta(3, 3)$, $t = 1, 2, ..., n$ and $\boldsymbol{\beta} = (\beta_0, \beta_1) = (1, 1)$. For the error term ϵ_t we have three different options (distributions), so as to have three different models.

- $\epsilon_t \sim Uniform(0, 1)$, $t = 1, 2, ..., n$ (model 1)
- $\epsilon_t \sim Beta(1, 1)$, $t = 1, 2, ..., n$ (model 2)
- $\epsilon_t \sim Weibull(2, 1)$, $t = 1, 2, ..., n$ (model 3)

For each model we simulate 1000 data sets, each one consisting of n data, and we will present the results based on the average of the data sets, for each model.

We have

$$q_\tau(y_t|x_t) = \beta_0(\tau) + \beta_1(\tau)x_{1t}$$

and we are interested in the extreme quantiles, $\tau = 0$ and $\tau = 1$. However, the asymmetric Laplace distribution used within our MCMC algorithm needs $0 < \tau < 1$. Therefore, we can approximate these extreme quantiles by using $\tau = 0.001$ and $\tau = 0.999$. In this way, we can have a very good estimation of $\boldsymbol{\beta}(0) = (\beta_0(0), \beta_1(0))$ and $\boldsymbol{\beta}(1) = (\beta_0(1), \beta_1(1))$, because $\hat{\boldsymbol{\beta}}(0.001) \approx \hat{\boldsymbol{\beta}}(0)$ and $\hat{\boldsymbol{\beta}}(0.999) \approx \hat{\boldsymbol{\beta}}(1)$.

Our aim is to estimate the model parameters under Bayesian extreme quantile regression. The number of simulated data, n, does not affect the true values of the parameters, because we know the model that generates these data. In other words, we are interested in estimating the parameters of three specific models and not the parameters that describe three specific data sets. Therefore, we can simulate as many data we want. We simulated 100, 500, 1000 and 10000 observations from our models and we saw that as n gets larger the estimation gets slightly better and the convergence of the Markov chain becomes faster.

Here we will show the results for $n = 500$. Table 2.1 shows the true values, the estimated mean and median of the model parameters, as well as the 95% confidence intervals, based on the different types of the error term's distribution. Figures 7.2, 7.3, 7.4, 7.5, 7.6 and 7.7, show the convergence of the parameters of our model, for the different types of the error term's distribution, as well as the posterior density of those parameters, based on simulations using our MCMC algorithm.

In order to compare Bayesian extreme quantile regression and the classical approach, we can obtain the smallest and largest extreme regression quantiles, as we said in a previous section, by solving the next two linear programming problems :

$$\max_{b \in R^p} \sum_{i=1}^{n} \mathbf{x}_i' \boldsymbol{\beta}, \quad subject\ to\ \ y_i \geq \mathbf{x}_i' \boldsymbol{\beta},\ i = 1, ..., n, \tag{2.4}$$

in order to obtain $\hat{\boldsymbol{\beta}}(0)$, and

$$\min_{b \in R^p} \sum_{i=1}^{n} \mathbf{x}_i' \boldsymbol{\beta}, \quad subject\ to\ \ y_i \leq \mathbf{x}_i' \boldsymbol{\beta},\ i = 1, ..., n,$$

which is equivalent to :

$$\max_{b \in R^p} - \sum_{i=1}^{n} \mathbf{x}_i' \boldsymbol{\beta}, \quad subject\ to\ \ - y_i \geq -\mathbf{x}_i' \boldsymbol{\beta},\ i = 1, ..., n, \tag{2.5}$$

in order to obtain $\hat{\boldsymbol{\beta}}(1)$.

The classical approach provides good results, even if we simulate $n = 100$ observations from our models. However, linear programming cannot provide the confidence intervals for the model parameters, because what we obtain is the optimal solution, but we can work out the approximate confidence intervals, by using the asymptotic results described in Appendix A. Table 2.2 shows the estimated and true values of our model parameters, as well as the 95% confidence intervals, based on the different types of the error term's distribution, using linear programming, for $n = 100$.

Another way to construct the confidence intervals for the model parameters could be a bootstrapping method. However, bootstrapping belongs to the general class of resampling methods, which cannot be considered as a classical approach. This is why we preferred to use the method described in Appendix A.

Type	Parameter	True Value	Mean	Median	Lower Limit	Upper Limit
Uniform	$\beta_0(0)$	1	0.92	0.94	0.61	1.09
	$\beta_1(0)$	1	1.03	1.04	0.59	1.44
	$\beta_0(1)$	2	2.08	2.06	1.89	2.42
	$\beta_1(1)$	1	0.97	0.96	0.61	1.39
Beta	$\beta_0(0)$	1	0.91	0.94	0.64	1.13
	$\beta_1(0)$	1	0.98	0.97	0.58	1.38
	$\beta_0(1)$	2	2.08	2.06	1.91	2.32
	$\beta_1(1)$	1	1.02	0.99	0.64	1.39
Weibull	$\beta_0(0)$	1	0.96	0.98	0.65	1.22
	$\beta_1(0)$	1	1.03	1.06	0.51	1.49

Table 2.1: True values, estimated (posterior) mean, estimated (posterior) median and 95% confidence intervals of the model parameters, based on the different types of the error term's distribution, using Bayesian extreme quantile regression (MCMC algorithm).

2.2.4 Real Data Set

This data set consists of the US ex-post real interest rates, which was considered by Garcia and Perron (1996). The series represents the three-month treasury bill rate deflated by the CPI inflation rate taken

Type	Parameter	True Value	Mean	Median	Lower Limit	Upper Limit
Uniform	$\beta_0(0)$	1	1.02	1.01	0.17	1.91
	$\beta_1(0)$	1	1	1	0.19	1.98
	$\beta_0(1)$	2	1.98	1.99	1.31	4.17
	$\beta_1(1)$	1	1.02	1	0.12	1.94
Beta	$\beta_0(0)$	1	1.02	1.01	0.28	1.93
	$\beta_1(0)$	1	1	1	0.21	1.81
	$\beta_0(1)$	2	1.98	1.99	1.35	4.32
	$\beta_1(1)$	1	1.02	1	0.22	1.96
Weibull	$\beta_0(0)$	1	1.01	1.01	0.19	1.78
	$\beta_1(0)$	1	1	1	0.17	1.87

Table 2.2: True values, estimated mean, estimated median and approximate 95% confidence intervals of the model parameters, based on the different types of the error term's distribution, using linear programming and classical extreme quantile regression approach.

from the Citibase data bank. Bai and Perron (2003) were interested in the presence of abrupt structural changes in the mean of this series, so they applied a linear regression model estimated by least squares, in order to estimate the multiple break-point model. They achieved that using dynamic programming by implementing an efficient algorithm to obtain global minimisers of the sum of squared residuals.

Methodology

This time we do not have a specific model, like in section 2.2.3, but a specific data set. Our aim is to model the extreme quantiles of the US ex-post real interest rates under Bayesian extreme quantile regression and compare it with the classical approach. Again, the extreme quantiles $\tau \to 0$ and $\tau \to 1$ will be approximated by $\tau = 0.001$ and $\tau = 0.999$ for the Bayesian extreme quantile regression (using the same MCMC algorithm as in the previous section). For the classical approach, though, using linear programming (equations of linear programs 2.4 and 2.5) ensures that $\tau = 0$, for the lowest extreme quantile and $\tau = 1$, for the highest extreme quantile.

Even if our aim is to explore the extreme quantiles of the data set, we also decide to check whether those methods have similarities, or differences, for other, non-extreme, values of τ. Therefore, using

the same MCMC algorithm, but this time for τ equal to $0.1, 0.5$ and 0.9, we perform Bayesian quantile regression. For the classical approach, obviously, we are not able to use the linear programming problems we used before, as they only correspond to τ equal to 0 and 1. In that case, we will use the "rq" function (R statistical package), which uses a very similar methodology, for τ equal to $0.1, 0.5$ and 0.9.

We can briefly point out that the "rq" function computes an estimate on the τ_{th} conditional quantile function of the response, given the covariates. It presumes a linear specification for the quantile regression model. In other words, it uses a model which is linear in parameters. Then, it minimizes a weighted sum of absolute residuals that can be formulated as a linear programming problem.

Table 2.3 summarizes our methodology for the real data set.

Extreme Quantiles
Bayesian extreme quantile regression MCMC for $\tau = 0.001$ and $\tau = 0.999$
Extreme quantile regression Linear Programming for $\tau = 0$ and $\tau = 1$
Non-Extreme Quantiles
Bayesian quantile regression MCMC for $\tau = 0.1$, $\tau = 0.5$ and $\tau = 0.9$
Quantile regression "rq" function for $\tau = 0.1$, $\tau = 0.5$ and $\tau = 0.9$

Table 2.3: Methodology for analyzing the real data set.

In section 2.2.3 it was shown that Bayesian extreme quantile regression works very well when we need to estimate the parameters of a given (or assumed) model. Therefore, our methodology for the real data set starts by assuming a model which describes the extreme quantiles of the data as good as possible and then we will try to estimate the parameters of the assumed model, under Bayesian extreme quantile regression and under the classical approach.

Linear model fitting

The US ex-post real interest rates data set consists of 103 observations. Following Smith (1994), we define $x_i = i - 52$, $i = 1, 2, ..., 103$, where i is the number of the observation. With this method we

avoid using large numbers within our calculations and algorithms, especially when we need to obtain x_i^2 (quadratic model) and x_i^3 (cubic model).

First, we use a linear model of the following form:

$$y_i = \beta_0 + \beta_1 \, x_i + \varepsilon_i, \;\; i = 1, 2, ..., 103.$$

Using our MCMC algorithm, for $\tau = 0.001$, we obtain

$$\hat{y}_i = -6.63 - 0.04 \, x_i, \;\; i = 1, 2, ..., 103$$

and for $\tau = 0.999$, we have

$$\hat{y}_i = 9.20 + 0.09 \, x_i, \;\; i = 1, 2, ..., 103.$$

Using linear programming, for $\tau = 0$, we obtain

$$\hat{y}_i = -6.31 + 0.02 \, x_i, \;\; i = 1, 2, ..., 103$$

and for $\tau = 1$, we have

$$\hat{y}_i = 8.45 + 0.11 \, x_i, \;\; i = 1, 2, ..., 103.$$

However, it is useful to compare those two methods not only for the extreme quantiles, but also for other values, when τ is equal to $0.1, 0.5$ and 0.9. From figure 2.1 it is obvious that the two methods are similar for the previous values of τ, but not for the extreme quantiles. Additionally, it is clear that the fitting is not good, as it does not follow the shape of the series. So, a quadratic fit should be more appropriate for our modeling.

Quadratic model fitting

Assuming a quadratic fitting means having a model of the following form :

$$y_i = \beta_0 + \beta_1 \, x_i + \beta_2 \, x_i^2 + \varepsilon_i, \;\; i = 1, 2, ..., 103.$$

Using our MCMC algorithm, for $\tau = 0.001$, we obtain

$$\hat{y}_i = -7.6 - 0.02 \, x_i + 0.001 \, x_i^2, \;\; i = 1, 2, ..., 103,$$

and for $\tau = 0.999$, we obtain

$$\hat{y}_i = 6.02 + 0.11 \, x_i + 0.003 \, x_i^2, \;\; i = 1, 2, ..., 103.$$

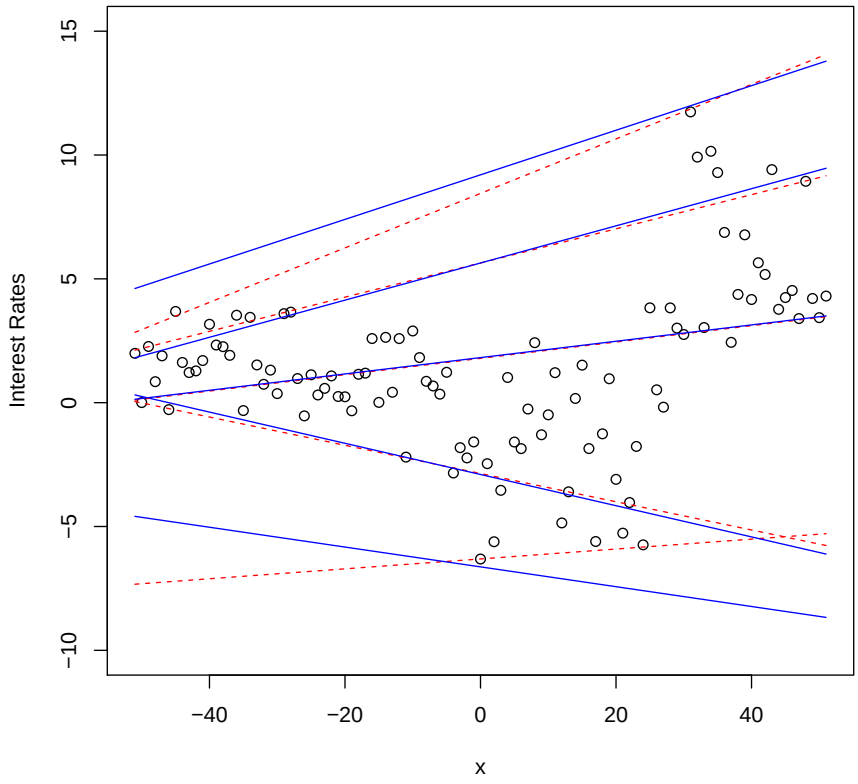

Figure 2.1: Fitting lines for all values of τ, using MCMC algorithm (straight line) and linear programming (dashed line).

Using linear programming, for $\tau = 0$, we obtain

$$\hat{y}_i = -6.31 - 0.02\, x_i + 0.002\, x_i^2, \;\; i = 1, 2, ..., 103,$$

and for $\tau = 1$, we obtain

$$\hat{y}_i = 5.9 + 0.13\, x_i + 0.002\, x_i^2, \;\; i = 1, 2, ..., 103.$$

Clearly, this fit is much better than the linear one, as we can see in figure 2.2. We must say that the two methods are similar when modeling non-extreme quantiles. For instance, when τ is equal to $0.1, 0.5$ and 0.9, it is obvious that the two methods are almost the same. Figures 7.8 and 7.9 show the convergence of our quadratic model parameters, based on the extreme quantiles.

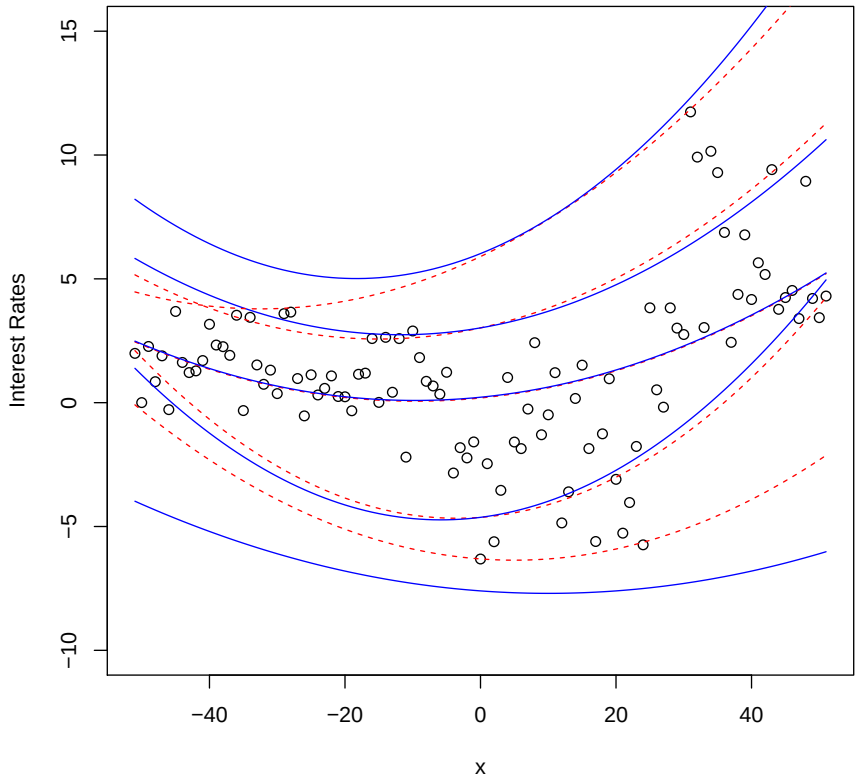

Figure 2.2: Quadratic model fit for all values of τ, using MCMC algorithm (straight line) and linear programming (dashed line).

Cubic model fitting

The quadratic model fits the shape of the data very well. However, it would be useful to check whether a cubic model is more appropriate. This means assuming a model of the following form:

$$y_i = \beta_0 + \beta_1\, x_i + \beta_2\, x_i^2 + \beta_3\, x_i^3 + \varepsilon_i, \;\; i = 1, 2, ..., 103.$$

Using our MCMC algorithm, for $\tau = 0.001$, we get

$$\hat{y}_i = -7.1 - 0.1\, x_i + 0.002\, x_i^2 + 0.00004\, x_i^3, \;\; i = 1, 2, ..., 103,$$

and for $\tau = 0.999$, we get

$$\hat{y}_i = 4.5 + 0.05\, x_i + 0.006\, x_i^2 + 0.00005\, x_i^3, \;\; i = 1, 2, ..., 103.$$

Using linear programming, for $\tau = 0$, we get

$$\hat{y}_i = -6.5 - 0.081\, x_i + 0.003\, x_i^2 + 0.000045\, x_i^3, \;\; i = 1, 2, ..., 103,$$

and for $\tau = 1$, we get

$$\hat{y}_i = 3.3 + 0.058\, x_i + 0.0049\, x_i^2 + 0.000073\, x_i^3, \;\; i = 1, 2, ..., 103.$$

Obviously, as we can see from figures 2.3 and 2.4 this fit is better than the quadratic fit and as a consequence better than the linear fit as well. If the cubic fit was similar to the quadratic fit, we would have chosen the quadratic one, as it has one less parameter to estimate (the cubic model has an extra parameter, β_3). However, the cubic model fits the extreme quantiles in a better way and models more accurately the shape of the data. We have to say that, again, like in the previous assumed models (linear and quadratic), Bayesian extreme quantile regression and the classical approach are similar when modeling non-extreme quantiles (when τ is equal to $0, 1$, 0.5 and 0.9).

By using the MCMC algorithm we can get values from the posterior density of our model parameters, $\beta_0, \beta_1, \beta_2$ and β_3. If we simulate 5000 values for each parameter we can obtain the mean, the median and the confidence intervals of every parameter (Table 2.4). Using the classical extreme quantile regression estimation we can get the mean and the approximate confidence intervals (Table 2.5).

Parameters	**2.5 % Quantile**	**97.5 % Quantile**	**Mean**	**Median**
$\beta_0(0.001)$	-11.82	-4.86	-7.10	-7.14
$\beta_1(0.001)$	-0.22	0.08	-0.10	-0.10
$\beta_2(0.001)$	-0.005	0.006	0.002	0.002
$\beta_3(0.001)$	0.00002	0.00006	0.00004	0.00004
$\beta_0(0.999)$	1.76	7.88	4.50	4.47
$\beta_1(0.999)$	-0.01	0.18	0.05	0.046
$\beta_2(0.999)$	0.001	0.021	0.006	0.006
$\beta_3(0.999)$	0.00001	0.00007	0.00005	0.00005

Table 2.4: Estimated (posterior) mean, estimated (posterior) median and 95% confidence intervals of the parameters, using Bayesian extreme quantile regression (MCMC algorithm), when assuming a cubic model.

Parameters	**Lower**	**Upper**	**Mean**
$\beta_0(0)$	-16.21	3.44	-6.50
$\beta_1(0)$	-3.67	4.18	-0.081
$\beta_2(0)$	-2.89	3.16	0.003
$\beta_3(0)$	-1.76	2.11	0.000045
$\beta_0(1)$	-4.81	11.06	3.30
$\beta_1(1)$	-3.91	4.12	0.058
$\beta_2(1)$	-2.72	3.11	0.0049
$\beta_3(1)$	-1.91	2.88	0.000073

Table 2.5: Estimated mean and approximate 95% confidence intervals of the parameters, using linear programming and classical extreme quantile regression approach, when assuming a cubic model.

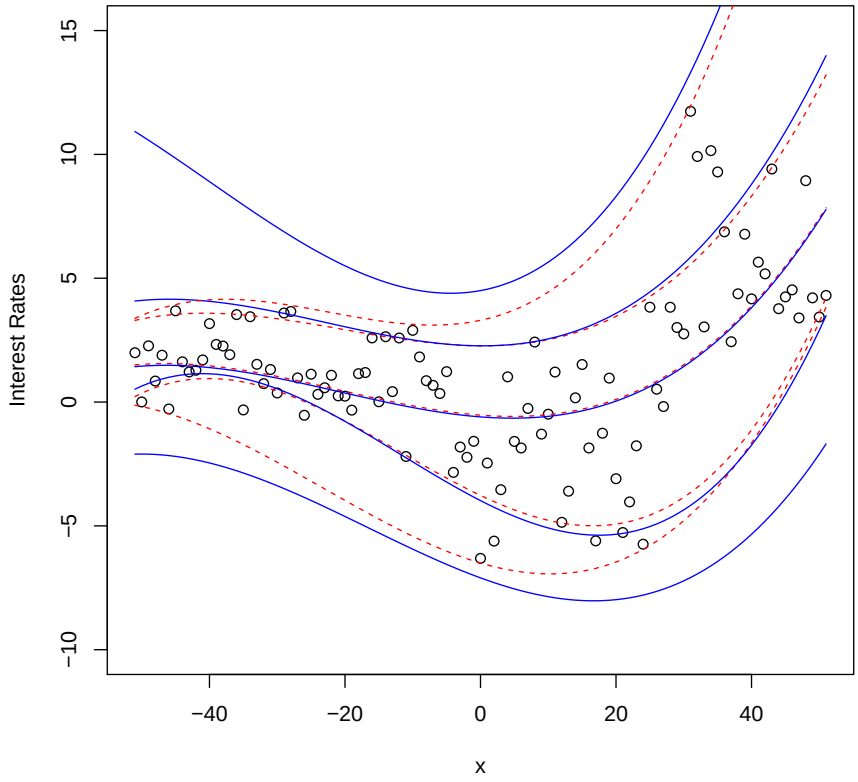

Figure 2.3: Cubic model fit for all values of τ, using MCMC algorithm (straight line) and linear programming (dashed line).

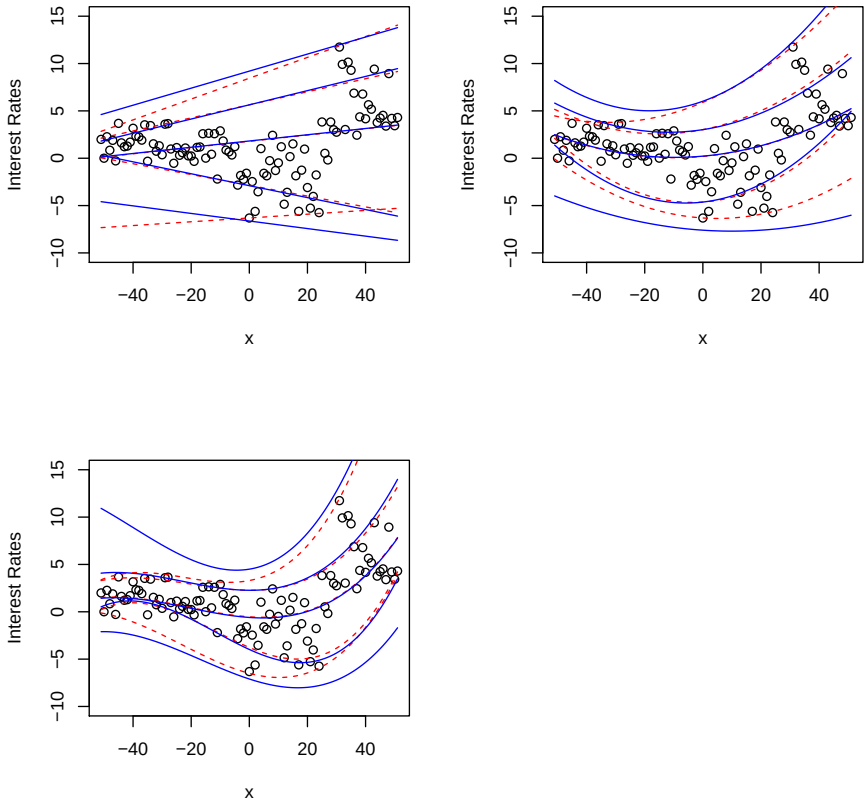

Figure 2.4: Comparison of linear, quadratic and cubic models, for all values of τ and for both methods; MCMC algorithm (straight line) and linear programming (dashed line).

2.3 Comparison of the two methods

Extreme quantile regression is an approach that requires methods to extrapolate information beyond the observed values of the dependent variable. In this chapter two methods were used in order to do that. We used Bayesian extreme quantile regression via a MCMC algorithm and we also used the classical extreme quantile regression via linear programming. After our analysis, we can conclude that Bayesian extreme quantile regression is an appropriate method, with a fast convergence and a good estimation of the model parameters and their confidence intervals. However, it is interesting and useful to compare the two methods.

Based on the simulated data set, we are certain that both methods have very good and similar results, concerning the parameter estimation, for all the different models (for all different types of the error term's distribution), because we know the true values of the parameters. However, the classical approach estimation needs a smaller number of data in order to estimate the parameters, than the Bayesian extreme quantile regression method. On the other hand, the confidence intervals obtained by Bayesian extreme quantile regression method are much better than those obtained by the classical approach.

For the real data set we used three different models to fit the extreme quantiles. A linear, a quadratic and a cubic model were used in order to obtain the best possible fitting. When we deal with non-extreme quantiles (τ is between 0.1 and 0.9) the estimation of the parameters is similar for both methods, for all three models. For the extreme quantiles, though, the parameter estimation obtained by the classical approach was better than the estimation obtained by the Bayesian extreme quantile regression. We concluded that cubic model fits the data in the best possible way and as a result we decided to obtain the confidence intervals of the parameters as well. This time, the results obtained by the Bayesian extreme quantile regression were much better than the results obtained by the classical approach.

To sum up, concerning the extreme quantiles, we are able to conclude that based on simulated data sets and a real data set, the Bayesian extreme quantile regression provides much better confidence intervals for the parameters than the classical approach (using the method in Appendix A - Properties of Quantile Regression), but the parameter estimation is better when we use the classical approach (using linear programming).

2.4 Combination of Linear Programming and MCMC algorithm

We showed that Bayesian extreme quantile regression and the classical quantile regression have similarities and differences as well. Concerning the extreme quantiles, it was shown that the classical approach provides better parameter estimation, but Bayesian extreme quantile regression provides better confidence intervals. This happens probably because the classical approach is based on approximated and asymptotic results. This means that this method requires large values of n ($n \to +\infty$, where n is the number of data). So, even if the classical approach can provide a very good parameter estimation, for a small value of n, it will not be able to provide good confidence intervals for the parameters. Additionally, Bayesian extreme quantile regression will always provide better confidence intervals, but not better parameter estimation.

For these reasons we thought of a similar approach, which will combine linear programming and an MCMC algorithm, in order to perform Bayesian extreme quantile regression. The idea behind that is that linear programming will ensure a very good parameter estimation and the MCMC algorithm will ensure a very good estimation of the confidence intervals.

That new approach first assumes a model, to fit the extreme quantiles, exactly as we did in the previous sections, and then uses linear programming in order to estimate the parameters of the model. Based on that model we simulate new data, which, as a consequence, correspond to the extreme quantiles. We apply our MCMC algorithm to these new data and we re-estimate the model parameters and obtain their confidence intervals as well.

The fact that linear programming enable us to obtain data which correspond to the extreme quantiles is very useful and important, because we can use our MCMC algorithm for $\tau = 0.5$ (median) and not for $\tau \to 0$ or $\tau \to 1$. And we know from a previous section that our MCMC algorithm provides a very good estimation (exactly similar to the linear programming estimation) when we deal with non-extreme quantiles. Alternatively, we are able to use a different MCMC algorithm, which will model the mean of the new data.

To be more specific, we will show: a) how that new method, which combines linear programming and MCMC methods, provides a very good estimation of the model parameters and their confidence intervals, and b) why the two MCMC algorithms (for modeling the median and the mean of the new data) have similar results.

2.4.1 Real Data Set Application

We will analyze again the US ex-post real interest rates, from section 2.2.4. As we saw in that section, the most appropriate model to fit the extreme quantiles of the series is a cubic model. First, we use linear programming to estimate the model parameters and we obtain:

$$\hat{y}_i = -6.5 - 0.081\, x_i + 0.003\, x_i^2 + 0.000045\, x_i^3, \;\; i = 1, 2, ..., 103, \;\; (\text{for } \tau = 0)$$

and

$$\hat{y}_i = 3.3 + 0.058\, x_i + 0.0049\, x_i^2 + 0.000073\, x_i^3, \;\; i = 1, 2, ..., 103, \;\; (\text{for } \tau = 1),$$

exactly as in section 2.2.4.

Then, we simulate new data which correspond to the extreme quantiles in the following way:

We simulate new data, for $\tau = 0$, from

$$\tilde{y}_i = -6.5 - 0.081\, x_i + 0.003\, x_i^2 + 0.000045\, x_i^3 + \varepsilon_i, \;\; i = 1, 2, ..., 103$$

and we simulate new data, for $\tau = 1$, from

$$\tilde{y}_i = 3.3 + 0.058\, x_i + 0.0049\, x_i^2 + 0.000073\, x_i^3 + \varepsilon_i, \;\; i = 1, 2, ..., 103,$$

where $\varepsilon_i \sim N(0, \sigma^2)$. In this application we will use $\sigma = 1$, but it works equally well for other values of σ as well.

The usage of the Normal error distribution enables us to use smaller or larger values of σ, without having any difference in modeling the mean (or the median) of the series, due to the fact that the new data points will be equally distributed around the mean (or median). However, it is reasonable to use a value of σ, in order to obtain new data, which will not have a larger variance than our initial data.

It is very important to point out that the Normal error distribution enables us to use MCMC algorithms which can model the mean and the median of the two new data sets, $\tilde{y}_i$, for both extreme quantiles, and provide similar results. This happens because for a Normal distribution the mean is equal to the median. If we had chosen a different error distribution, where the mean is not equal to the variance, the results would not have been similar.

Modeling the Median

We have two simulated data sets. One corresponds to $\tau = 0$ and the other one corresponds to $\tau = 1$. We apply our MCMC algorithm (described in section 2.2.3; details in Appendix A), for $\tau = 0.5$, on

both data sets (we use the simulated data $\tilde{\mathbf{y}}$ and not the initial data $\mathbf{y}$) and we obtain the results shown in table 2.6. This is very convenient as we do not need to construct a new MCMC algorithm, but use the same algorithm we used for Bayesian extreme quantile regression before. However, this time we will use it for $\tau = 0.5$ and not for $\tau \to 0$ and $\tau \to 1$, because we know that the new simulated data correspond to the extreme quantiles.

Parameters	**2.5 % Quantile**	**97.5 % Quantile**	**Mean**	**Median**
$\beta_0(0)$	-9.23	-3.46	-6.51	-6.50
$\beta_1(0)$	-0.21	0.09	-0.08	-0.082
$\beta_2(0)$	-0.003	0.006	0.003	0.0031
$\beta_3(0)$	0.00002	0.00006	0.000046	0.000045
$\beta_0(1)$	0.18	6.77	3.31	3.30
$\beta_1(1)$	-0.01	0.19	0.058	0.057
$\beta_2(1)$	0.002	0.018	0.0048	0.0049
$\beta_3(1)$	0.00004	0.00009	0.000073	0.000074

Table 2.6: Estimated (posterior) mean, estimated (posterior) median and 95% confidence intervals of the parameters, using Bayesian extreme quantile regression (combination of linear programming and MCMC algorithm), when assuming a cubic model.

Modeling the mean

An alternative way is to model the mean of the two simulated data sets, which correspond to the extreme quantiles. We construct a new MCMC algorithm without using the asymmetric Laplace distribution, but the Normal distribution of the error term. We know that the error term follows a Normal distribution, because we simulated the new data sets. Therefore, the likelihood is of the following form:

$$L(\tilde{\mathbf{y}}|\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}} exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\left(\tilde{y}_i - \beta_0 - \beta_1 x_i - \beta_2 x_i^2 - \beta_3 x_i^3\right)^2\right\}.$$

The posterior distributions of the parameters, given the data, will be obtained from:

$$\pi(\boldsymbol{\beta}|\tilde{\mathbf{y}}) \propto L(\tilde{\mathbf{y}}|\boldsymbol{\beta})\pi(\boldsymbol{\beta}).$$

We assume independent improper priors, $\pi(\boldsymbol{\beta})$, for all the components of $\boldsymbol{\beta}$ and we get the following posterior distributions:

$$\beta_1 \sim N\left(\frac{\sum_{i=1}^{n}\left[x_i(\tilde{y}_i-\beta_0-\beta_2x_i^2-\beta_3x_i^3)\right]}{\sum_{i=1}^{n}x_i^2},\frac{1}{\sum_{i=1}^{n}x_i^2}\right)$$

$$\beta_2 \sim N\left(\frac{\sum_{i=1}^{n}[x_i^2(\tilde{y}_i-\beta_0-\beta_1x_i-\beta_3x_i^3)]}{\sum_{i=1}^{n}x_i^4},\frac{1}{\sum_{i=1}^{n}x_i^4}\right)$$

$$\beta_3 \sim N\left(\frac{\sum_{i=1}^{n}[x_i^3(\tilde{y}_i-\beta_0-\beta_1x_i-\beta_2x_i^2)]}{\sum_{i=1}^{n}x_i^6},\frac{1}{\sum_{i=1}^{n}x_i^6}\right)$$

$$\beta_0 \sim N\left(\frac{\sum_{i=1}^{n}(\tilde{y}_i-\beta_1x_i-\beta_2x_i^2-\beta_3x_i^3)}{n},\frac{1}{n}\right).$$

For more details see Appendix A.

We use Gibbs sampling in order to update each of the parameters at each step, because it is simple to simulate values from the exact posterior distributions. We have a very fast convergence of the Markov chain. A burn-in period of 2000 iterations was excluded and a sample of 8000 values from the posterior distribution of each parameter was collected, in order to get our results, which are shown in table 2.7.

Parameters	**2.5 % Quantile**	**97.5 % Quantile**	**Mean**	**Median**
$\beta_0(0)$	-9.20	-3.44	-6.50	-6.50
$\beta_1(0)$	-0.22	0.09	-0.081	-0.08
$\beta_2(0)$	-0.001	0.007	0.0031	0.0031
$\beta_3(0)$	0.00002	0.00006	0.000044	0.000045
$\beta_0(1)$	0.16	6.71	3.30	3.31
$\beta_1(1)$	-0.03	0.16	0.059	0.058
$\beta_2(1)$	0.004	0.022	0.0049	0.0049
$\beta_3(1)$	0.00003	0.00009	0.000074	0.000074

Table 2.7: Estimated (posterior) mean, estimated (posterior) median and 95% confidence intervals of the parameters, using Bayesian extreme quantile regression (combination of linear programming and MCMC algorithm), when assuming a cubic model.

The algorithm of the new approach

The new general algorithm of the method which combines linear programming and MCMC algorithms is the following:

1. Use linear programming to estimate the parameters $\boldsymbol{\beta}$, which model the data, for $\tau = 0$ and $\tau = 1$.
2. Use the estimated parameters $\hat{\boldsymbol{\beta}}$ to simulate new data $\tilde{\mathbf{y}}$, which correspond to $\tau = 0$ and $\tau = 1$.
3. Use MCMC algorithm to re-estimate $\boldsymbol{\beta}$ and obtain their confidence intervals, given $\tilde{\mathbf{y}}$ (by modeling the mean or the median of the data $\tilde{\mathbf{y}}$).

Obviously, by comparing the tables 2.4, 2.5, 2.6 and 2.7 we can see that the new method of Bayesian extreme quantile regression, which combines linear programming and an MCMC algorithm, provides better results than the method which uses linear programming or MCMC algorithms alone. It is very interesting that a problem of extreme quantile regression of a data set $\mathbf{y}$ can be transformed into an equivalent problem of modeling the mean or the median of another data set $\tilde{\mathbf{y}}$. Additionally, this method provides results for $\tau = 0$ and $\tau = 1$, where the Bayesian extreme quantile regression in sections 2.2.4 and 2.2.3 provided results for $\tau = 0.001$ and $\tau = 0.999$.

This is the method we are going to use in one of the following chapters, in order to perform Bayesian extreme quantile regression for hidden Markov models. However, we are not going to use MCMC algorithms to re-estimate the parameters, as in step 3 of the general algorithm, but estimate new parameters, which describe the hidden state of the data.

Chapter 3

Hidden Markov Models (HMMs)

Hidden Markov modeling was initially introduced in the second half of the 1960s and early 1970s but, in the last several years, statistical methods for estimating hidden Markov models have become increasingly popular. Mainly, there are two reasons why this happened. First, this kind of models have a very rich mathematical structure and as a consequence they can form a strong theoretical basis in order to support a wide range of applications. Second, for these applications, hidden Markov models work very well in practice, when they are applied properly.

Hidden Markov models are best known for their use in speech recognition (Rabiner 1989; Fox et al. 2009), as evidenced by the number of published papers and talks at major speech conferences. Some other areas, which hidden Markov models have been successfully applied to and led to advances, are signal processing (Juang and Rabiner 1991; Andrieu and Doucet 1999), biology (Fredkin and Rice 1992; Leroux and Putterman 1992), genetics (Churchill 1989; Liu, Newland and Lawrence 1999), ecology (Guttorp 1995), image analysis (Romberg, Choi and Baraniuk 1999), economics (Hamilton 1989, 1990; Albert and Chib 1993), network security (Scott 1999, 2001), handwriting recognition (Koschinski, Winkler, Lang 1995; Conelli 1998, 2000), pattern recognition (Smyth 1994; Bishop 1995), fault-detection (Smyth 1994), natural language processing (Manning and Schuetze 1999), information retrieval (Teh et al. 2006), molecular dynamics (Horenko and Schütte 2008) and biochemistry (McKinney et al. 2006; Gopich and Szabo 2009).

The basic theory of hidden Markov models was published, between 1960s and 1970s, in a series of classic papers by Baum and his colleagues (Baum, Petrie, Soules and Weiss 1970). Then, it was -initially- implemented for speech processing applications at IBM (International Business Machines)

in the 1970s. However, the basic theory of hidden Markov models was published in mathematical journals and as a consequence they were not generally read by engineers working on problems concerning speech processing. Moreover, most readers were not able to apply the theory of hidden Markov models to their own research, because they couldn't understand it. This happened due to the fact that the original applications of that theory to speech processing did not provide sufficient tutorial material for the readers. In order to overcome the previous problems and difficulties, several tutorial papers were written, which provided a sufficient level of detail for a number of researchers or research labs to begin work using hidden Markov models in individual speech processing applications. That is why widespread understanding and application of the theory of hidden Markov models to speech processing has occurred only within the 1980s and not right after 1970s, when that theory was initially introduced.

A general hidden Markov model is a model where an underlying and unobserved sequence of states follows a finite state-space Markov chain and the probability distribution of the observation at any time is determined only by the current state of that Markov chain. More specifically, when we need to describe a stochastic process for which observations are made at discrete times and the observed values depend on an unobserved Markovian underlying process, we use discrete-time hidden Markov models. Such a probabilistic model includes a model for the underlying process, as well as a model for the observed process, which assumes dependence on the unobserved values of the underlying sequence of states.

3.1 Discrete-time finite state-space HMM

Let $\{X_t\}$ be a homogeneous discrete-time Markov chain on a finite state-space $S = \{1, ..., m\}$. This chain has the Markov property which states that, given the value of X_t, the values X_h , $h > t$, do not depend on the values X_s , $s < t$. That is

$$Pr(X_h|X_t, X_{t-1}, ...) = Pr(X_h|X_t).$$

Let $P = [p_{ij}]$ be the transition probability matrix of the chain with stationary distribution $\pi = (\pi_1, \pi_2, ...\pi_m)$. The general form of the matrix is

$$P = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1m} \\ p_{21} & p_{22} & \cdots & p_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mm} \end{pmatrix},$$

where the elements of P are given by

$$p_{ij} = Pr(X_t = j | X_{t-1} = i), i, j = 1, 2, ..., m, \tag{3.1}$$

i.e they are the one-step transition probabilities of the Markov chain. Therefore, p_{ij}, which is the i^{th} row j^{th} column element of P, gives the probability that, if the chain is at state i at time $t-1$, it will move to state j at time t (next step). Here, it is assumed that the next state is dependent only upon the current state (*the Markov assumption*). It is obvious that each row of the matrix P sums to one

$$\sum_{j=1}^{m} p_{ij} = 1,$$

for all i, and that $p_{ij} \geq 0$ for every (i, j), since they represent probabilities. The probabilities of the chain remaining at the current state, denoted by p_{ii}, are the diagonal elements of the matrix P. Additionally, the state transition probabilities p_{ij} are independent of the actual time at which a transition takes place (*the stationarity assumption*)

$$p_{ij} = Pr(X_{t_1} = j | X_{t_1-1} = i) = Pr(X_{t_2} = j | X_{t_2-1} = i)\ , 1 \leq t_1, t_2 \leq T.$$

Now, let $\{Y_t\}$ be the random process we are interested in and for which observations are made at discrete times $t = 1, 2, ...T$. In order to formulate a hidden Markov model for the random process $\{Y_t\}$, we assume that the distribution of Y_t depends, through a function h of known form, on the unobserved value of X_t,

$$Y_t = h(X_t) + e_t\ , t = 1, 2, ..., T\ ,$$

where e_t are additive noise terms whose distribution may depend on the value of X_t. Therefore, the variables $X_1, X_2, ..., X_T$ represent the hidden states of a mechanism/process that has generated the observed data $y_1, y_2, ..., y_T$.

Let $\mathbf{y}^T = (y_1, y_2, ...y_T)$ be a realization of the observation sequence and $\mathbf{x}^T = (x_1, x_2, ..., x_T)$ be a realization of the state sequence. The unobserved values of the state sequence x_t, depend on

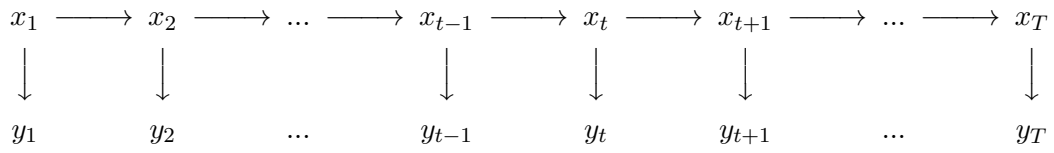

Figure 3.1: Independence structure of a discrete-time finite state-space HMM

each other through the transition matrix P, of a discrete-time Markov chain. Then, the value of the observation y_t, at time t, depends only on the state of the chain at time t. Conditionally on $\mathbf{x}^T$ and the model parameters, denoted by θ the observations $y_1, y_2, ..., y_T$ are independent of each other. The conditional independence structure of a discrete-time finite state-space hidden Markov model is depicted in figure 3.1. Due to the conditional independence property of the hidden Markov model the likelihood of the observed data $\mathbf{y}^T$ given the hidden states $\mathbf{x}^T$ is given by

$$Pr\left(\mathbf{y}^T|\mathbf{x}^T,\theta\right) = \prod_{t=1}^{T} Pr(y_t|x_t,\theta).$$

There exist hidden Markov models of different orders. For example a model of order 0 assumes that

$$Pr(X_{t_1} = i) = Pr(X_{t_2} = i)\ , i = 1, 2, ..., m,$$

for all t_1, t_2. A hidden Markov model of order 1 is every model that follows the Markov assumption and it is defined by a transition probability matrix with elements of the form (3.1). This is the class of hidden Markov models that we will use in our analysis. Generally, in a hidden Markov model of order k the next state depends on the previous k states. Despite the fact that a higher order hidden Markov model will have a higher complexity, it is possible to obtain a k^{th}-order model by defining the transition probabilities as

$$p_{i_1,i_2,...,i_k,j} = Pr(X_t = j|X_{t-1} = i_1, X_{t-2} = i_2, ..., X_{t-k} = i_k)\ , 1 \leq i_1, i_2, ...i_k, j \leq m.$$

However, hidden Markov models are usually taken to be of order 1, because every k^{th}-order hidden Markov model can be converted into an equivalent first-order hidden Markov model.

3.2 Types of HMMs

There are two basic types of hidden Markov models; ergodic and left-right models. In an ergodic model (Levinson 1986) every state can be reached from any other state in a finite number of steps.

In other words, there is a probability that we can pass from a state i of the model to a state j (figure 3.2). A special case of ergodic hidden Markov models is the one where every state of the model can be reached from every other state in a single step. This kind of model is alternatively called fully connected hidden Markov model. This special case of models has been useful in applications to some speech-modeling tasks and it has the property that every coefficient of its transition matrix is positive, $p_{ij} > 0$, for $i, j = 1, 2, ..., m$.

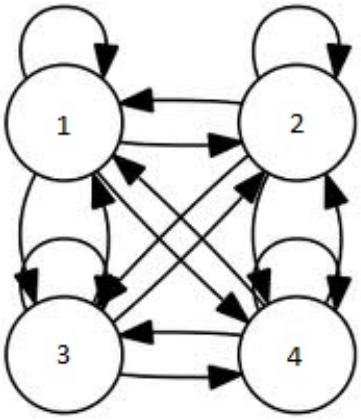

Figure 3.2: Example of a 4-state ergodic HMM.

The left-right or Bakis models (Bakis, 1976) allow the hidden Markov state either to pass from a given state to a state with larger index, or remain at the same state. In other words, no transition is allowed to a state with lower index than the current state's at any step (figure 3.3). Therefore, we have the property

$$p_{ij} = 0, \text{ for } j < i.$$

Clearly, those models are defined through upper triangular transition matrices. For a model with m states the transition coefficient $p_{mm} = 1$, as there is no other state with higher index than m and the chain must remain at the m^{th} state. Hence, the transition matrix of this model has the form

$$P= \begin{pmatrix} p_{11} & p_{12} & ... & p_{1m} \\ 0 & p_{22} & ... & p_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & 1 \end{pmatrix}.$$

Furthermore, the state sequence must begin in state 1 and end in state m. Usually, in left-right hidden Markov models large jumps in state indices are not allowed. In such *constrained* left-right hidden Markov models there are some constraints of the form $p_{ij} = 0$, $j > i + \delta$, where δ is the largest number of jumps allowed in the model.

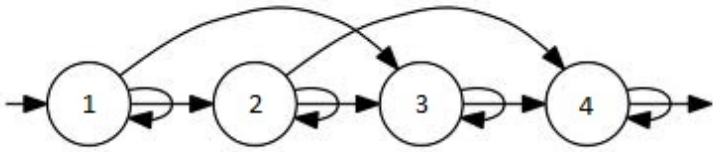

Figure 3.3: Example of a 4-state left-right HMM.

3.2.1 A special case : Break-point Models

The constraint left-right hidden Markov models, where only one jump is allowed ($\delta = 1$), are known in the literature as the class of break-point models. In these models, when the chain is leaving the current state, say a state indexed by i, then transition is made to the state indexed by $i + 1$, for $1 \leq i < m$. The transition matrix for such a model has the form

$$P= \begin{pmatrix} p_{11} & p_{12} & 0 & 0 & ... & 0 & 0 \\ 0 & p_{22} & p_{23} & 0 & ... & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & ... & p_{m-1,m-1} & p_{m-1,m} \\ 0 & 0 & 0 & 0 & ... & 0 & 1 \end{pmatrix}.$$

Therefore, the chain starts at state 1 and when leaving the i^{th} state it always moves to the state $(i+1)$, until it eventually reaches the m^{th} state. Again we have $\sum_{j=1}^{m} p_{ij} = 1$, for all $i = 1, 2, ..., m$, which

is simplified to

$$p_{ii} + p_{ii+1} = 1 \, , i = 1, 2, ..., m - 1.$$

Break-point models are considered as a special case of hidden Markov models that are widely used in financial econometric applications. The breaks usually correspond to important changes in the economic structure. They are used to model economic processes where one or more multiple structural changes (breaks) appear at discrete time points due to exogenous reasons, such as financial crises (Jeanne and Masson, 2000; Cerra, 2005; Hamilton, 2005) or abrupt changes in government policy (Hamilton, 1988; Sims and Zha, 2004, Davig, 2004). A break-point model, expressed as a special case of a HMM was used by Chib (1998) to model data on the number of coal-mining disasters by year in Britain over the period 1851-1962 (Jarrett, 1979). Break-point models have been also used to model the log-run trend in GNP (Hamilton, 1989), the behavior of foreign exchange rates and real interest rates (Garcia and Perron, 1996; Bai and Perron, 2003) and the evolution of shock returns (Kim *et al.*, 1998), among many others. In those cases, the hidden underlying process may reflect changes in monetary policies, exchange rates regimes, financial market regulation or any change in the economic environment. For instance, Garcia and Perron (1996) and later Bai and Perron (2003) considered the US ex-post real interest rate and they were interested in the presence of abrupt structural changes in the mean of the series.

When analyzing financial time-series under break-point models, it is of particular interest to infer the times of the breaks as they represent the times of structural changes in the studied economic process. Note that for a m-state break-point model the times of the breaks $t_1, t_2, ..., t_{m-1}$ are the time-points at which transitions occur in the Markovian underlying process. Therefore, inferring the times of those breaks is equivalent to inferring the hidden state sequence.

3.3 HMMs and three basic problems

In order to produce hidden Markov models useful in real-world applications there are three basic problems, which need to be solved.

- 1st problem - evaluation problem :

Supposing we have an observation sequence and a specific model, how do we efficiently compute the probability of the observation sequence, given that model? (In other words, what is the probability

that this observation sequence was produced by that model?)

- 2nd problem - decoding problem :

Supposing, again, we have an observation sequence and a specific model, how do we deduce from the observation sequence the most likely state sequence in a meaningful manner? (For instance, how do we find a corresponding state sequence that best ”explains” the observations?)

- 3rd problem - estimation problem :

How do we adjust the parameters of our model in order to maximize the probability of the observation sequence, given the model?

3.3.1 Solutions to the three basic problems of HMMs

In this section our aim is to briefly describe the solutions to the above problems and make the connections with some algorithms (Viterbi, Baum-Welch, Forward-Backward), rather than present the solutions in great detail.

The first problem (evaluation problem) can be also viewed in a different but extremely useful way; that of how well a given model matches a given observation sequence. Therefore, if we are in a situation in which we want to choose the model which best matches the observations, among several competing models, the only thing we have to do is to solve the evaluation problem. The most straightforward way of solving this problem, is to enumerate every possible state sequence of length equal to the number of observations. It is obvious that for a large number of observations or states this calculation is computationally infeasible. In fact, even for small values, the number of calculations is very large. However, there is a more efficient procedure to solve this problem which is called Forward-Backward algorithm (Baum 1972).

The second problem (decoding problem) is a way of uncovering the hidden part of the model. Finding the most likely state sequence is not always needed, because the probability measure of an hidden Markov model does not explicitly involve the state sequence. However, in many applications it is important and useful to uncover that sequence. This problem can be solved in several possible ways. One way is to maximize the probability of being in state i, at time t, given the observed sequence $\mathbf{y}^T$ and the model parameters θ ,

$$\gamma_t(i) = Pr(X_t = i|\mathbf{y}^T, \theta).$$

In order to solve this problem one can use dynamic programming methods such as the Viterbi algorithm (Forney 1973).

The third problem (estimation problem) concerns methods of optimizing the model parameters and it is the most difficult one, as there is no known way of solving it analytically. Usually, the maximum likelihood method is followed, in order to find parameters that maximize the probability of the observation sequence $\mathbf{y}^T$, given the state sequence $\mathbf{x}^T$,

$$Pr(\mathbf{y}^T|\mathbf{x}^T, \theta).$$

This maximization can be accomplished via the Baum-Welch algorithm (Baum, Petrie, Soules and Weiss 1970).

Alternatively, one may use Bayesian inference to estimate the parameters of the hidden Markov model via Markov chain Monte Carlo methods.

Forward-Backward algorithm

The Forward-Backward algorithm (FB; Baum *et al.* 1970) is a set of filtering recursions that are used to calculate the likelihood and to simulate realizations of the underlying process of a hidden Markov model given the values of the model parameters. Usually it is used within more general recursive schemes, where the parameters need to be estimated. In particular, the Forward-Backward algorithm can be used to evaluate the likelihood within the steps of an expectation-maximization (EM) algorithm for ML estimation, or to simulate realizations of the hidden chain in an MCMC algorithm for Bayesian estimation. This algorithm will be used in applications in this dissertation and, therefore, we present it in detail in the following section.

Viterbi algorithm

This algorithm was generated by Andrew Viterbi as an error-correction scheme for noisy digital communication links. Nowadays, it is commonly used in speech recognition, keyword spotting, computational linguistics and bio-informatics. It is a dynamic programming algorithm, which finds the most likely state sequence to have generated a sequence of observations. For example, a possible observed sequence could be an acoustic signal and a string of text could be the -hidden- state sequence that caused the observations. The algorithm is based on several assumptions. Both observed and hidden events must be in a sequence, which often corresponds to time. These two sequences need to be

ranged, while an observation has to correspond to exactly one hidden event. Moreover, computing the most likely -hidden- state sequence up to a certain point t must depend only on the observed event at point t, and the most likely sequence at point $t-1$. A transition from a previous state to a new one is marked by an incremental metric (number), which depends on the transition probability from the old to the new state. The aim of the algorithm is to keep a number for each state, so, when an event occurs, the Viterbi algorithm examines the new possible states and chooses the best one using these metrics.

Baum-Welch algorithm

The Baum-Welch algorithm was developed by Leonard E. Baum and his co-workers in a series of papers published between 1966 and 1972 (Baum and Petrie 1966; Baum and Egon 1967; Baum and Sell 1968; Baum, Petrie, Soules and Weiss 1970; Baum 1972). The name of Welch appears only as joint author -with Baum- of a paper listed by Baum, Petrie, Soules and Weiss (1970) as submitted for publication. It is an example of an algorithm of the Estimation-Maximization (EM) type. The Baum-Welch algorithm updates the model parameters until convergence, usually following the Forward-Backward algorithm, due to its interpretation as an extension of the forward induction procedure to the evaluation problem.

In this thesis we implement an MCMC algorithm for inference about discrete-time finite state-space hidden Markov models using the Forward-Backward algorithm. This algorithm consists of updates of the hidden sequence of states given the model parameters. Then, it updates the values of the parameters from their conditional distributions and repeats this procedure until convergence.

The reason why we use the Forward-Backward algorithm and not any of the other two (Viterbi or Baum-Welch) is that the Forward-Backward algorithm is more appropriate given our model construction (we need to calculate the likelihood and simulate realizations of the latent variables given the model parameters).

3.4 Forward-Backward algorithm

There are some problems we need to solve, in order to produce hidden Markov models useful in real-world applications. One of them is known as ”evaluation problem”. Supposing we have an

observation sequence and a specific model, how do we efficiently compute the probability of the observation sequence, given that model? (In other words, which is the probability that this observation sequence was produced by that model?). This problem can be also viewed in an alternative way; that of how well a given model matches a given observation sequence. Therefore, if we are in a situation in which we want to choose the model which best matches the observations, among several competing models, the only thing we have to do is to solve the evaluation problem. The most straightforward way of solving this problem, is to enumerate every possible state sequence of length equal to the number of observations. It is obvious that for a large number of observations or states this calculation is computationally infeasible. In fact, even for small values, the number of calculations is very large. However, there is a more efficient procedure to solve this problem which is called Forward-Backward algorithm (FB).

The Forward-Backward algorithm (Baum *et al.* 1970; Baum 1972; Rabiner 1989; Cappé *et al.* 2005) is an inference algorithm for hidden Markov models, which computes the posterior marginal distributions of all hidden state variables, given a sequence of observations. Principles of dynamic programming are used by this algorithm, in order to compute the values that are necessary for obtaining the posterior marginal distributions mentioned before. As we will show analytically in the following paragraph, the Forward-Backward algorithm first computes a set of forward variables, which enable us to obtain the probability of ending up in any particular state of the Markov chain, given the previous observations in the sequence. After that, the algorithm computes a set of backward variables, which enable us to obtain the probability of observing the remaining observations, given any starting point. Finally, the algorithm combines those sets of forward and backward variables, in order to provide the distribution over any states, at any specific time point, given the entire observation sequence.

Let $\mathbf{Y}^t = (Y_1, Y_2, ..., Y_t)$ be the history of the observation process and $\mathbf{X}^t = (X_1, X_2, ..., X_t)$ be the sequence of states, up to time t. Let f_i denote the distribution of $Y_t|X_t = i$, $i = 1, ..., m$, which is parameterized by θ_i. The joint probability of a realization $(\mathbf{Y}^T, \mathbf{X}^T)$ is given by

$$L(\mathbf{y}^T, \mathbf{x}^T|\theta) = \pi_{x_1} f_{x_1}(y_1) p_{x_1,x_2} f_{x_2}(y_2) ... p_{x_{T-1},x_T} f_{x_T}(y_T), \tag{3.2}$$

where $\theta = (\theta_1, \theta_2, ..., \theta_m, P)$ is the vector of all unknown parameters in the model and $\pi = (\pi_1, \pi_2, ..., \pi_m)$ is the stationary distribution of the matrix P.

One possible way to derive the likelihood of the observed data, $L(\mathbf{y}^T|\theta)$, since $\mathbf{x}^T$ is not observed, is by summation of the equation (3.2) over all the m^T possible sets of states. It is obvious that as T

increases, even for small values of m, it is infeasible to evaluate the likelihood function. Clearly, we need some other way of calculating the likelihood in order to overcome this difficulty. This can be done by evaluating the likelihood sequentially, using the Forward-Backward algorithm. In this way, $L(\mathbf{y}^T|\theta)$ is calculated, given the value of θ, in a sequential manner.

The terminology "forward" refers to the calculation of the likelihood function via a forward recursion, while "backward" refers to simulating realizations from the distribution of the hidden underlying sequence $\mathbf{X}^T = (X_1, X_2, ..., X_T)$ via a backward recursion. We define the forward variables as

$$a_t(i) = Pr(\mathbf{y}^t, X_t = i|\theta) = Pr(X_t = i)Pr(\mathbf{y}^t|X_t = i, \theta)\ , t = 1, 2, ..., T,$$

with $a_1(i) = \pi_i f_i(y_1)$. The forward variable at time t is the joint probability of the data (observation sequence) up to time t and the state of the hidden process at time t, given the model parameters. The forward variables for $t = 2, 3, ..., T$ can be calculated recursively by

$$a_t(i) = \sum_{j=1}^{m} Pr(\mathbf{y}^t, X_{t-1} = j, X_t = i|\theta) = \left[\sum_{j=1}^{m} a_{t-1}(j)p_{ji}\right] f_i(y_t).$$

At the final step of the recursion we calculate

$$a_T(i) = Pr(\mathbf{y}^T, X_T = i|\theta),$$

for $i = 1, 2, ...m$. Then, the likelihood can be obtained as

$$L(\mathbf{y}^T|\theta) = \sum_{i=1}^{m} a_T(i).$$

This calculation involves $O(m^2T)$ steps instead of the $O(m^T)$ needed for direct evaluation of the likelihood. Going backwards, we are able to simulate a realization $(x_1, x_2, ..., x_T)$ from the joint distribution of the hidden state variables $(X_1, X_2, ...X_T)$. The Markov property allows us to write the joint distribution of the hidden states as a product

$$Pr(\mathbf{x}^T|\mathbf{y}^T, \theta) = Pr(x_T|\mathbf{y}^T, \theta)...Pr(x_t|\mathbf{y}^T, x_{t+1}, \theta)...Pr(x_1|\mathbf{y}^T, x_2, \theta).$$

Then, we can simulate the state at time T from

$$Pr(X_T = i|\mathbf{y}^T, \theta) = \frac{Pr(\mathbf{y}^T, X_T = i|\theta)}{L(\mathbf{y}^T|\theta)} = \frac{a_T(i)}{\sum_{j=1}^{m} a_T(j)},$$

for $i = 1, 2, ...m$. Calculating backwards, for $t = T-1, T-2, ...1$, the state at time t given the state at time $t+1$ can be simulated from the distribution

$$Pr(X_t = i|\mathbf{y}^T, x_{t+1}, \theta) = \frac{Pr(X_t = i|\mathbf{y}^t, \theta)Pr(x_{t+1}|X_t = i)}{\sum_{j=1}^{m} Pr(X_t = j|\mathbf{y}^t, \theta)Pr(x_{t+1}|X_t = j)}$$

$$= \frac{a_t(i)p_{i,x_{t+1}}}{\sum_{j=1}^{m} a_t(j)p_{j,x_{t+1}}},$$

for $i = 1, 2, ..., m$. Note that the variables $a_t(i)$ have been calculated during the forward step of the algorithm.

Recursive schemes based on the Forward-Backward algorithm can be used to implement the EM algorithm or to propose realizations of the underlying Markov process within an MCMC scheme (Scott 2002).

Chapter 4

Bayesian Extreme Quantile Inference for HMMs

In this chapter we use quantile regression, Bayesian inference and hidden Markov modeling in order to analyze the highest and lowest extreme quantile of two financial time series. The data sets are the US ex-post real interest rates and the US treasury bill real interest rates. For the analysis of these data sets we use discrete-time m-state hidden Markov models and multiple break-point models. Additionally, we choose two different ways of performing our analysis. One way using the m-state Normal HMM and another one using the m-state asymmetric Laplace distribution (ALD) HMM.

The special case of break-point models can be more parsimonious than the general case of hidden Markov models, especially for modeling time series for which each state when left is almost never revisited. This phenomenon is common in financial econometrics time series, where the breaks correspond to permanent changes in the structure of the economy.

Generally, for each data set, we consider a problem of model choice first; namely we are interested in inferring the number of states m which best describe the data. In the context of break-point models this amounts to examining whether or not there exist structural changes (break-points) in our data sets and inferring their number. In the context of hidden Markov models we examine how many states the model must have to describe the data best in terms of model fit and model complexity. In order to achieve this we analyze the data under different models using linear programming and Gibbs sampling with data augmentation (in the case of the Normal HMM) and a combination of Gibbs and Metropolis-Hastings sampling (in the case of the ALD HMM). Then, for each model we compute the

associated value of the DIC based on the MCMC output. The best model is the one with the smallest value of the DIC.

Basically, for the analysis of the data, we run our MCMC algorithm for a number of Normal (or ALD) hidden Markov models, each one assuming a different number of states or break-points. The construction of the MCMC algorithm is the same for all these models, apart from a few changes concerning the prior distribution of the parameters.

4.1 The m-state Normal HMM

Let $\mathbf{y}^T = (y_1, y_2, ...y_T)$ be a sample of observations from an m-state Normal hidden Markov model. We assume that $\{X_t\}$ (the hidden underlying process) is a m-state Markov chain taking the values $1, 2, 3, ...m$, with transition matrix P and stationary distribution $\pi = (\pi_1, \pi_2, ...\pi_m)$. The distribution associated with the i^{th} state of this Markov chain is

$$f_i(y) = f_N\left(y|\mu_i, \kappa_i^{-1}\right),$$

where $f_N(y|\mu, \sigma^2)$ denotes the probability distribution function of a Normal random variable with mean μ and variance σ^2. Therefore,

$$f_i(y) = \frac{\sqrt{\kappa_i}}{\sqrt{2\pi}} exp\left\{-\frac{\kappa_i(y-\mu_i)^2}{2}\right\}.$$

Since $\mathbf{x}^T$ is a sequence of states, which is unknown, the likelihood of the observed data $\mathbf{y}^T$ can be obtained as

$$L(\mathbf{y}^T|\theta) = \sum_{\mathbf{x}^T} \pi_{x_1} f_{x_1}(y_1) p_{x_1,x_2} f_{x_2}(y_2) ... p_{x_{T-1},x_T} f_{x_T}(y_T), \tag{4.1}$$

where $\theta = (\mu, \kappa, P)$ is the totality of unknown parameters and $\mu = (\mu_1, \mu_2, ..., \mu_m)$, $\kappa = (\kappa_1, \kappa_2, ..., \kappa_m)$. Instead of summing over all the possible sets of states, we can evaluate $L(\mathbf{y}^T|\theta)$ efficiently using the Forward-Backward algorithm as following. The forward variables $a_t(i)$, $i = 1, 2, ..., m$ can be calculated as

$$a_1(i) = \pi_i f_N\left(y_1|\mu_i, \kappa_i^{-1}\right), \tag{4.2}$$

and for $t = 2, 3, ..., T$ using the recursion

$$a_t(i) = \left[a_{t-1}(1)p_{1i} + a_{t-1}(2)p_{2i} + ... + a_{t-1}(m)p_{mi}\right] f_N\left(y_t|\mu_i, \kappa_i^{-1}\right). \tag{4.3}$$

Then the likelihood of $\mathbf{y}^T$ is obtained as

$$L(\mathbf{y}^T|\theta) = a_T(1) + a_T(2) + ... + a_T(m), \tag{4.4}$$

and the state at time T can be simulated from

$$Pr(X_T = i|\mathbf{y}^T, \theta) = \frac{a_T(i)}{a_T(1) + a_T(2) + ... + a_T(m)} = \frac{a_T(i)}{L(\mathbf{y}^T|\theta)} \tag{4.5}$$

for $i = 1, 2, ..., m$. For $t = T-1, T-2, ..., 1$ the state at time t, given the state at time $t+1$ can be simulated from

$$Pr(X_t = i|\mathbf{y}^T, x_{t+1}, \theta) = \frac{a_t(i)p_{i,x_{t+1}}}{a_t(1)p_{1,x_{t+1}} + a_t(2)p_{2,x_{t+1}} + ... + a_t(m)p_{m,x_{t+1}}} \tag{4.6}$$

for $i = 1, 2, ..., m$.

4.1.1 Gibbs sampling for Normal HMMs

Let us consider again the m-state Normal hidden Markov model. Suppose we have observed data $\mathbf{y}^T = (y_1, y_2, ..., y_T)$, where $\mathbf{y}^T$ is a sample of T consecutive observations from a time series of interest, modeled by a Normal discrete-time m-state hidden Markov model. As mentioned above, Bayesian inference for a hidden Markov model is an example of inference for missing data problems using data augmentation. In our setting of the Normal hidden Markov model we can construct a Gibbs sampler which, at each iteration, first updates the latent data $\mathbf{x}^T$ of the hidden Markov model given the model parameters $\theta = (\mu, \kappa, P)$, and then updates θ given $\mathbf{x}^T$. Realizations of the hidden sequence of states $\mathbf{x}^T$ given θ are simulated using the Forward-Backward algorithm of section 4.1 (more specifically, equations 4.5 and 4.6). Then the parameters $\mu_1, \mu_2, ..., \mu_m, \kappa_1, \kappa_2, ..., \kappa_m, p_{11}, p_{12}, ..., p_{mm}$ can be simulated from their full conditional posterior distributions via Gibbs steps.

Under the Bayesian approach, first we need to specify priors for the parameters $\theta = (\mu, \kappa, P)$. We begin from the matrix P and denote as $p_{i.}$ the i^{th} row of the matrix, for $i = 1, 2, ..., m$. It is assumed that each row $p_{i.}$ follows a Dirichlet distribution with parameter $\omega = (\omega_1, \omega_2, ..., \omega_m)$.

$$p_{i.} \sim Dir(w)\ ,\ i = 1, 2, ..., m.$$

Then, we assume a conjugate Normal prior distribution for each mean μ_i with mean ξ and variance λ^{-1}, that is

$$\mu_i \sim N(\xi, \lambda^{-1})\ ,\ i = 1, 2, ..., m,$$

and a Gamma prior for each precision κ_i with parameters a and b, that is

$$\kappa_i \sim Gamma(a,b)\ , i = 1,2,...,m.$$

We have chosen to parameterize the Normal distribution in terms of the precision rather than in terms of the variance in order to make the calculations simpler. Therefore, for $i = 1,2,...,m$ we have

$$\pi(p_{i.}) = \frac{1}{B(\omega)} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} \propto \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} \tag{4.7}$$

$$\pi(\mu_i) = \frac{\sqrt{\lambda}}{\sqrt{2\pi}} exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\} \propto exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\}$$

$$\pi(\kappa_i) = \frac{b^a}{\Gamma(a)} \kappa_i^{a-1} exp\left\{-b\kappa_i\right\} \propto \kappa_i^{a-1} exp\left\{-b\kappa_i\right\}$$

where the normalizing constant $B(\omega)$ is the multinomial Beta function, which is expressed in terms of the Gamma function, as

$$B(\omega) = \frac{\prod_{i=1}^{m} \Gamma(\omega_i)}{\Gamma\left(\sum_{i=1}^{m} \omega_i\right)}.$$

The likelihood of the observed data $\mathbf{y}^T$ given the hidden sequence of states $\mathbf{x}^T$, under an m-state Normal hidden Markov model is given by

$$\begin{aligned} L(\mathbf{y}^T|\mathbf{x}^T, \mu, \sigma^2, P) &= \pi(x_1) f_{x_1}(y_1) p_{x_1,x_2} f_{x_2}(y_2) ... p_{x_{T-1},x_T} f_{x_T}(y_T) \\ &= \pi(x_1) \underbrace{\prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}}}_{A} \underbrace{\prod_{i=1}^{m} \prod_{t:x_t=i} f_i(y_t)}_{B}, \end{aligned} \tag{4.8}$$

where $f_i(y_t)$, $i = 1,2,...,m$, is the probability distribution function of the Normal distribution associated with the i^{th} state and n_{ij} is the number of times the chain passes from state i to state j:

$$n_{ij} = \sum_{t=1}^{T} I(x_t = i, x_{t+1} = j).$$

Clearly, the above formula is written as a product of two terms, each involving a subset of the model parameters. From factor A we can make inference about the parameters (elements) of the matrix P, while from B we can make inference about the Normal parameters μ and κ_i.

The joint posterior distribution of the model parameters given the observed and the unobserved data of the hidden Markov model is given by

$$f(\mu, \kappa, P|\mathbf{y}^T, \mathbf{x}^T) \propto L(\mathbf{y}^T|\mathbf{x}^T, \mu, \kappa, P)\pi(P)\pi(\mu)\pi(\kappa)$$

$$\propto \prod_{i=1}^{m}\prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m}\prod_{t:x_t=i} \sqrt{\kappa_i} exp\left\{-\frac{\kappa_i}{2}(y_t-\mu_i)^2\right\} \times \prod_{i=1}^{m}\prod_{j=1}^{m} p_{ij}^{\omega_j-1} \times$$

$$exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i-\xi)^2\right\} \times \prod_{i=1}^{m} \kappa_i^{a-1} exp\left\{-b\kappa_i\right\}.$$

Then, we are able to obtain the full conditional posterior distributions of the parameters $p_{i.}$, μ_i and κ_i, each being proportional to $f(\mu, \kappa, P|\mathbf{y}^T, \mathbf{x}^T)$ regarded as a function of the respective parameter only. All of the full conditionals are of known form.

The full conditional of $p_{i.}$, the i^{th} row of the matrix P, $i = 1, 2, ..., m$, is a Dirichlet distribution with parameters $(n_{i.} + \omega) = (n_{i1} + \omega_1, n_{i2} + \omega_2, ..., n_{im} + \omega_m)$ where $n_{i.} = (n_{i1}, n_{i2}, ..., n_{im})$:

$$\pi(p_{i.}|\mathbf{y}^T, \mathbf{x}^T, \mu, \kappa) \equiv Dir(n_{i.} + \omega). \tag{4.9}$$

The full conditional of μ_i, $i = 1, 2, ..., m$, is a Normal distribution with mean $\xi_2 = \frac{\kappa_i \sum_{t:x_t=i} y_t + \lambda\xi}{n_i\kappa_i + \lambda}$ and variance $\lambda_2^{-1} = \frac{1}{n_i\kappa_i + \lambda}$:

$$\pi(\mu_i|\mathbf{y}^T, \mathbf{x}^T, P, \kappa) \equiv N\left(\frac{\kappa_i \sum_{t:x_t=i} y_t + \lambda\xi}{n_i\kappa_i + \lambda}, \frac{1}{n_i\kappa_i + \lambda}\right),$$

where $n_i = \sum_{t=1}^{T} I(x_t = i)$. Finally, the full conditional of κ_i, $i = 1, 2, ..., m$, is a Gamma distribution with parameters $a_2 = a + \frac{n_i}{2}$ and $b_2 = b + \sum_{t:x_t=i} \frac{(y_t-\mu_i)^2}{2}$,

$$\pi(\kappa_i|\mathbf{y}^T, \mathbf{x}^T, P, \mu) \equiv Gamma\left(a + \frac{n}{2}, b + \sum_{t:x_t=i} \frac{(y_t-\mu_i)^2}{2}\right).$$

Details for the MCMC algorithm and the posterior distribution calculations, for the Normal HMM, can be found in Appendix B.

4.1.2 Prior Specification

The parameters of the prior distributions in the MCMC algorithm for an m-state Normal hidden Markov model are $a = 0.1, b = 0.1$, which are the parameters of a Gamma prior distribution of each precision κ_i, $i = 1, 2, ..., m$; $\xi = 0, \lambda = 0.1$, which are the parameters of a Normal prior distribution for each mean μ_i, $i = 1, 2, ..., m$; $\omega = (1, ..., 1)$, a vector of 1s with length m (the specified number of states), which is the parameter of a Dirichlet prior distribution for each line $p_{i.}$ of the model's transition matrix P.

Therefore, the parameters of the m-state Normal hidden Markov model and their priors are

$$p_{i\cdot} \sim Dir((1, ..., 1)) \, , i = 1, 2, ..., m,$$

$$\mu_i \sim N(0, 10) \, , i = 1, 2, ..., m,$$

$$\kappa_i \sim Gamma(0.1, 0.1) \, , i = 1, 2, ..., m.$$

4.2 US ex-post real interest rates

The US ex-post real interest rates data set was considered by Garcia and Perron (1996). The series represents the three-month treasury bill rate deflated by the CPI inflation rate taken from the Citibase data bank. In this chapter we are interested in inferring the number of states in the Normal HMM (section 4.2.1 and 4.2.2) and the ALD HMM (section 4.5.3), for the highest and lowest extreme quantiles of the series ($\tau = 1$ and $\tau = 0$, respectively). We model these extreme quantiles of the series in three different ways. We consider a Normal HMM with a quadratic model fit of the extreme quantiles, then a Normal HMM with a cubic model fit of the extreme quantiles and an ALD HMM. Note that it is known from section 2.2.4 that a cubic model fits the extreme quantiles of the series in a better way than the quadratic model. However, we are interested in exploring how a different assumption on the extreme quantile model fit can affect the estimation of the hidden state, via HMMs. Finally, in a following chapter, we also consider a continuous state-space HMM for this data set (section 5.3.1).

4.2.1 Normal HMM with a quadratic model fit of the extreme quantiles

Let us start by considering a Normal HMM as described in section 4.1. Based on the shape of our data, we assume that those quantiles can be described by a quadratic model of the form:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i,$$

$i = 1, 2, ..., n = 103$, where y_i are the observations, x_i are the times of the observations and ε_i is the error term. We define $x_i = i - 52$, where i is the number of the observation. Using linear programming we estimate the values of β_j, $j = 0, 1, 2$, and we obtain the two following models, as in section 2.2.4:

$$\tilde{y}_i = -6.31 - 0.02 x_i + 0.002 x_i^2 + \varepsilon_i,$$

$$\tilde{y}_i = 5.9 + 0.13 x_i + 0.002 x_i^2 + \varepsilon_i,$$

$i = 1, 2, ..., n$, for the lowest ($\tau = 0$) and highest ($\tau = 1$) extreme quantile, respectively. We choose $\varepsilon_i \sim N(0, 1)$. Based on the first model, we simulate data which, obviously, correspond to the lowest extreme quantile of our data set and then we apply our MCMC algorithm, as described in section 4.1.1. We follow the same method for the second model (highest extreme quantile).

We run our MCMC algorithm assuming 2,3,4 and 5 hidden states for both extreme quantiles and we choose the best model based on the DIC. Table 4.1 shows that the best model is a 4-state Normal HMM for the highest extreme quantile and a 3-state Normal HMM for the lowest extreme quantile (figure 4.1). That means that we need one more state, or two more parameters (μ_4 and κ_4), in order to model the highest extreme quantile, compared to the lowest extreme quantile.

Models	**DIC**
2-states HMM	5.07
3-states HMM	4.81
4-states HMM	**4.24**
5-states HMM	4.76
2-states HMM	5.32
3-states HMM	**4.34**
4-states HMM	4.97
5-states HMM	4.92

Table 4.1: Values of DIC for different Normal HMMs for the US ex-post real interest rates, for the highest and lowest extreme quantiles, respectively. A quadratic model was used to fit the extreme quantiles. The best models are indicated with bold characters.

The estimated Normal parameters (mean μ and precision κ) for the best models are shown in table 4.2.

4.2.2 Normal HMM with a cubic model fit of the extreme quantiles

Let us now consider again a Normal HMM, but this time assume that the extreme quantiles of the series are described by a cubic model of the form:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \varepsilon_i,$$

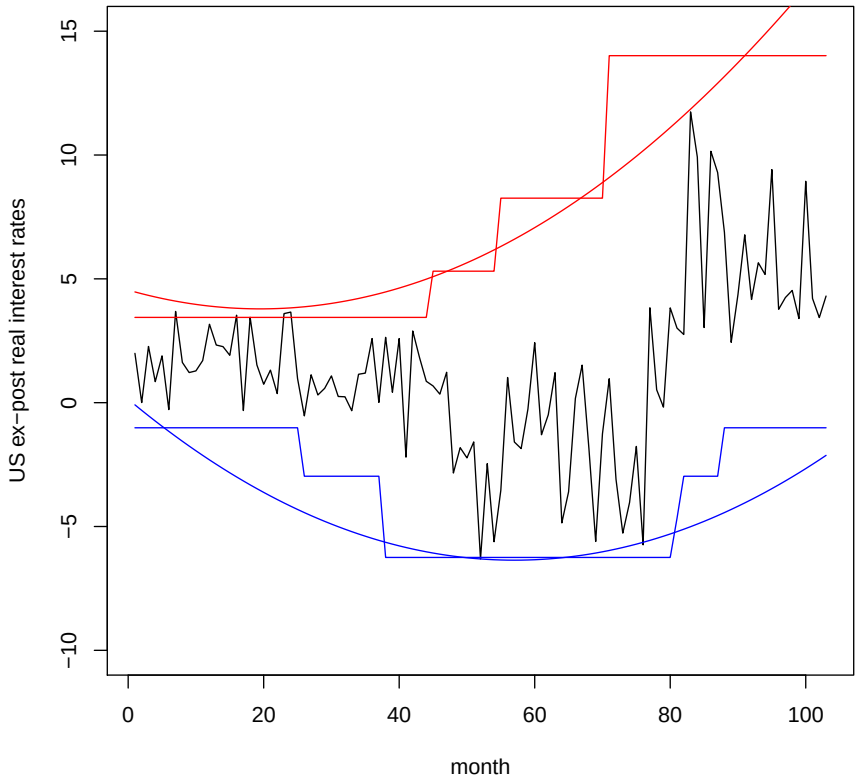

Figure 4.1: Model fit of the Normal HMMs for the US ex-post real interest rates, when using a quadratic model fit for the extreme quantiles. The red lines correspond to the highest extreme quantile (4-state Normal HMM) and the blue lines correspond to the lowest extreme quantile (3-state Normal HMM). The smooth lines represent the extreme quantile model fit and the stepwise lines represent the hidden states.

$i = 1, 2, ..., n$, where y_i are the observations, x_i are the times of the observations and ε_i is the error term. We define $x_i = i - 52$, where i is the number of the observation. Using linear programming we estimate the values of β_j, $j = 0, 1, 2, 3$, and we obtain the two following models, as in sections 2.2.4 and 2.4.1:

$$\tilde{y}_i = -6.5 - 0.081x_i + 0.003x_i^2 + 0.000045x_i^3 + \varepsilon_i,$$

$$\tilde{y}_i = 3.3 + 0.058x_i + 0.0049x_i^2 + 0.000073x_i^3 + \varepsilon_i,$$

4-state Normal HMM							
Highest extreme quantile							
μ_1	κ_1	μ_2	κ_2	μ_3	κ_3	μ_4	κ_4
3.44	4.20	5.31	1.01	8.26	0.54	14.01	0.20

3-state Normal HMM					
Lowest extreme quantile					
μ_1	κ_1	μ_2	κ_2	μ_3	κ_3
-6.25	1.04	-4.67	1.63	-1.01	1.05

Table 4.2: Normal parameter estimates for US ex-post real interest rates, using a quadratic model fit for the extreme quantiles.

$i = 1, 2, ..., n$, for the lowest ($\tau = 0$) and highest ($\tau = 1$) extreme quantile, respectively. We choose $\varepsilon_i \sim N(0, 1)$. Based on the first model, we simulate data which, obviously, correspond to the lowest extreme quantile of our data set and then we apply our MCMC algorithm. We follow the same method for the second model (highest extreme quantile).

We run our MCMC algorithm assuming 2,3,4 and 5 hidden states for both extreme quantiles and we choose the best model based on the DIC. Table 4.3 shows that the best model is a 2-state Normal HMM for the highest extreme quantile and a 3-state Normal HMM for the lowest extreme quantile (figure 4.2). That means that we need one more state, or two more parameters (μ_3 and κ_3), in order to model the lowest extreme quantile, compared to the highest extreme quantile.

The estimated Normal parameters (mean μ and precision κ) for the best models are shown in table 4.4.

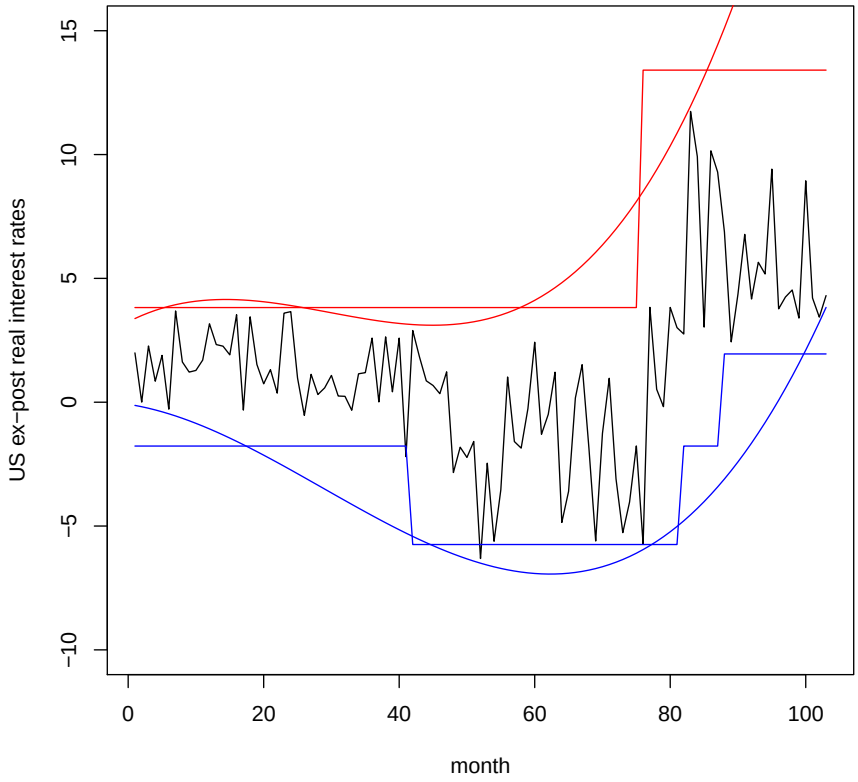

Figure 4.2: Model fit of the Normal HMMs for the US ex-post real interest rates, when using a cubic model fit for the extreme quantiles. The red line corresponds to the highest extreme quantile (2-state Normal HMM) and the blue line corresponds to the lowest extreme quantile (3-state Normal HMM). The smooth lines represent the extreme quantile model fit and the stepwise lines represent the hidden states.

Models	**DIC**
2-states HMM	**4.09**
3-states HMM	4.92
4-states HMM	5.04
5-states HMM	4.88
2-states HMM	5.02
3-states HMM	**4.29**
4-states HMM	4.81
5-states HMM	5.06

Table 4.3: Values of DIC for different Normal HMMs for the US ex-post real interest rates, for the highest and lowest extreme quantiles, respectively. A cubic model was used to fit the extreme quantiles. The best models are indicated with bold characters.

2-state Normal HMM			
Highest extreme quantile			
μ_1	κ_1	μ_2	κ_2
3.82	0.71	13.41	0.02

3-state Normal HMM					
Lowest extreme quantile					
μ_1	κ_1	μ_2	κ_2	μ_3	κ_3
-5.75	0.55	-1.77	0.74	1.95	0.41

Table 4.4: Normal HMM parameter estimates for US ex-post real interest rates, using a cubic model fit for the extreme quantiles

4.3 US treasury bill real interest rates

The US real interest rates data set consists of monthly observations from 1959 to 1998 that represent the US Treasury constant maturity interest rates. In this chapter we are interested in inferring the number of the states in the Normal HMM (sections 4.3.1, 4.3.2 and 4.3.3) and ALD HMM (section 4.5.4), for the extreme quantiles of the series (highest and lowest). In our analysis we consider up to 7 hidden states and, again, we use the best model based on DIC. We are also interested in exploring how a different assumption on the extreme quantile model fit can affect the estimation of the hidden state, via HMMs. First, we consider a Normal HMM with a quadratic model fit of the extreme quantiles. We observe that we do not have a very good fit for the highest extreme quantile ($\tau = 1$). Therefore, we use a cubic model fit for that quantile, in order to investigate whether we can obtain a better fit. The results clearly show that this model has a very bad fitting for half of the data points. However, it gives us the idea of modeling the highest extreme quantile by using two different quadratic models for our Normal HMM. Then we use an ALD HMM and, finally, we also consider a continuous state-space HMM, in a following chapter (section 5.3.2).

4.3.1 Normal HMM with a quadratic model fit of the extreme quantiles

We start by considering a Normal HMM as described in section 4.1. Based on the shape of our data, we assume that a very good idea is to fit the extreme quantiles by using a quadratic model of the form:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i,$$

$i = 1, 2, ..., n = 527$, where y_i are the observations, x_i are the times of the observations and ε_i is the error term. This data set consists of 527 observations. We define $x_i = i - 263$, where i is the number of the observation. As we mentioned in a previous chapter, this method enable us to avoid using large numbers for our calculations, especially when we need to obtain x_i^2 and x_i^3.

Using linear programming we estimate the values of β_j, $j = 0, 1, 2$, and we obtain the two following models :

$$\tilde{y}_i = 3.52 - 0.00029 x_i - 0.000032 x_i^2 + \varepsilon_i,$$

$$\tilde{y}_i = 16.19 + 0.0014 x_i - 0.00018 x_i^2 + \varepsilon_i,$$

$i = 1, 2, ..., n$, for the lowest ($\tau = 0$) and highest ($\tau = 1$) extreme quantile, respectively. We choose $\varepsilon_i \sim N(0, 1)$. Based on the first model, we simulate data which, obviously, correspond to the lowest

extreme quantile of our data set and then we apply our MCMC algorithm, as described in section 4.1.1. We follow the same method for the second model (highest extreme quantile).

We run our MCMC algorithm assuming $2, 3, 4, 5, 6$ and 7 hidden states for both extreme quantiles and we choose the best model based on the DIC. Table 4.5 shows that the best model is a 3-state Normal HMM for the both extreme quantiles (figure 4.3).

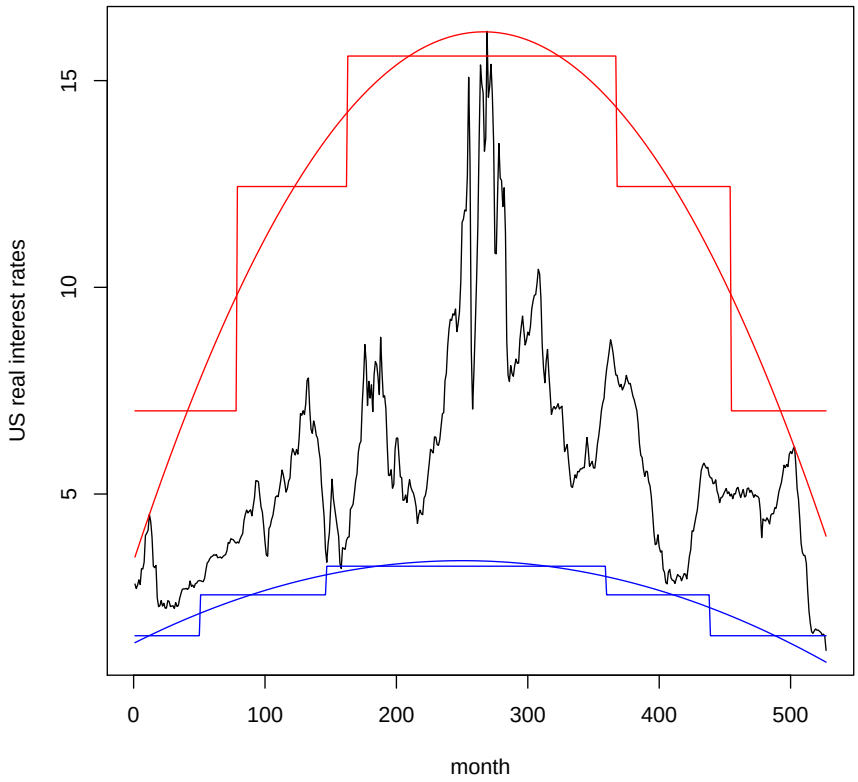

Figure 4.3: Model fit of the 3-state Normal HMM for the US real interest rates. The red lines correspond to the highest extreme quantile and the blue lines correspond to the lowest extreme quantile. The smooth lines represent the extreme quantile model fit and the stepwise lines represent the hidden states.

The estimated Normal parameters (mean μ and precision κ) for the best models (3-state Normal HMM) are shown in table 4.6. The HMM fitting for the lowest extreme quantile of the series looks very good. However, the HMM fitting for the highest extreme quantile does not look so good, because

Models	**DIC**
2 states	8.22
3 states	**7.17**
4 states	8.38
5 states	7.98
6 states	8.13
7 states	7.86
2 states	8.33
3 states	**7.28**
4 states	8.31
5 states	7.87
6 states	7.98
7 states	7.77

Table 4.5: Values of DIC for different Normal HMMs for the US real interest rates, for the highest and lowest extreme quantiles, respectively. The best models are indicated with bold characters.

the fitting lines of the hidden states of the HMM are not very close to the data. This suggests that maybe a different model could fit the highest extreme quantile of the series in a more appropriate way.

3-state Normal HMM					
Highest extreme quantile					
μ_1	κ_1	μ_2	κ_2	μ_3	κ_3
7.02	0.28	12.46	0.58	15.06	2.59
Lowest extreme quantile					
μ_1	κ_1	μ_2	κ_2	μ_3	κ_3
1.59	4.47	2.56	5.25	3.25	5.67

Table 4.6: Normal parameter estimates for US real interest rates.

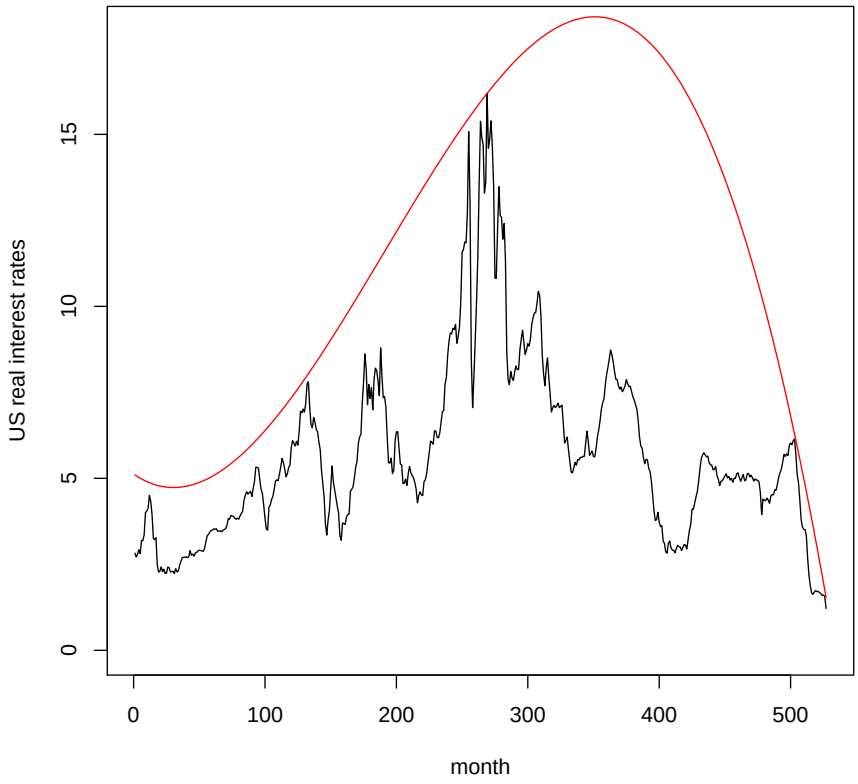

Figure 4.4: The cubic model fit for the highest extreme quantile of the US real interest rates.

4.3.2 Normal HMM with a cubic model fit of the highest extreme quantile

We are looking for a different model to fit the highest extreme quantile of the series. After the quadratic model of the previous section, our next choice is a cubic model of the form:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \varepsilon_i,$$

$i = 1, 2, ..., n = 527$, where y_i are the observations, x_i are the times of the observations and ε_i is the error term. We define $x_i = i - 263$, where i is the number of the observation. Using linear programming, for the highest extreme quantile ($\tau = 1$), we estimate the values of β_j, $j = 0, 1, 2, 3$, and we obtain the following model :

$$\hat{y}_i = 15.89 + 0.051 x_i - 0.00018 x_i^2 - 0.00000083 x_i^3,$$

$i = 1, 2, ..., n$. From figure 4.4, it is clear that the fit is not appropriate. However, we can see that this model is very good for the first 269 months. After the 269^{th} month (maximum observation) the fit is very bad. In spite of being a not appropriate fit, this gives us the idea of modeling the highest extreme quantile by using two different models; one model for the first 269 months and another model for the rest of the months.

4.3.3 Normal HMM with two quadratic model fits of the highest extreme quantile

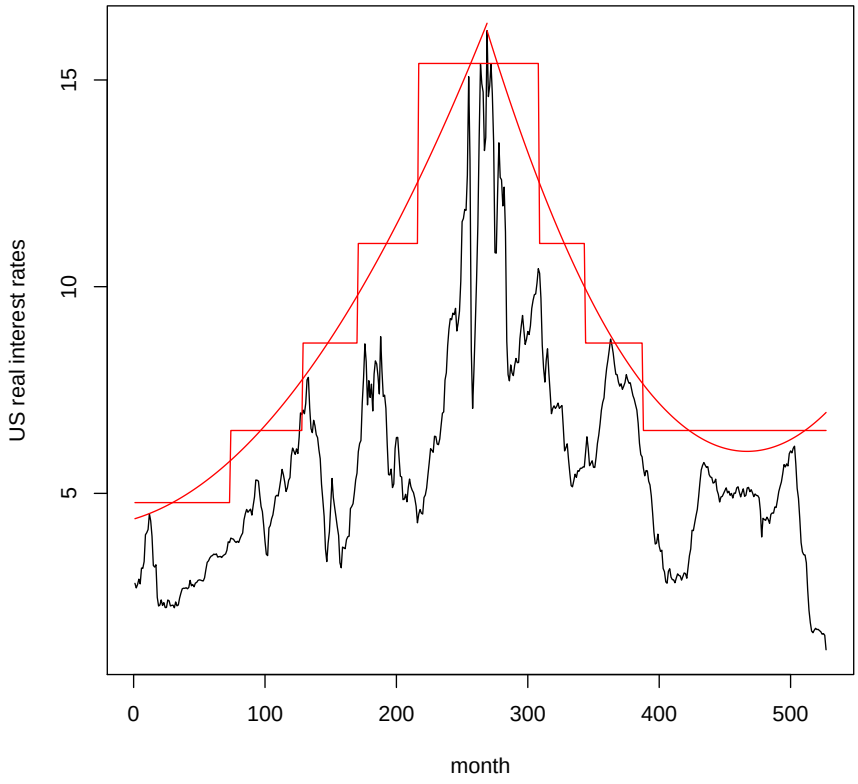

Figure 4.5: The two quadratic model fit (red curves) and the 5-state Normal HMM for the highest extreme quantile of the US real interest rates. The smooth line represents the extreme quantile model fit and the stepwise line represents the hidden states.

We think that it would be a good idea to model the US real interest rates by using two quadratic models to fit the highest extreme quantile. The first one will be for months $1 - 269$ and the second

one for months $270 - 527$. We are able to use this technique, as we are not interested in obtaining a model which fits the highest extreme quantile, but we are interested in obtaining new simulated data that represent the highest extreme quantile of the series. If we can achieve the best representation by using more than one model to fit the highest extreme quantile, then those are the models we should use.

Using linear programming, for the highest extreme quantile ($\tau = 1$), we estimate the parameters of the quadratic model, and we obtain the following models :

$$\tilde{y}_i = 15.9 + 0.078x_i + 0.00013x_i^2 + \varepsilon_i, \quad i = 1, 2, ..., 269,$$

$$\tilde{y}_i = 16.82 - 0.106x_i + 0.00026x_i^2 + \varepsilon_i, \quad i = 270, 271, ..., 527.$$

In figure 4.5 we can see the red curves, as a proof that the above quadratic models fit the highest extreme quantile very well. Based on these two models we simulate new data, which represent the highest extreme quantile of the series and then we apply our MCMC algorithm, for $2, 3, 4, 5, 6$ and 7 hidden states. Then, we choose the best model based on the DIC. Table 4.7 shows that the best HMM for the highest extreme quantile of the series is a 5-state Normal HMM (figure 4.5).

Models	**DIC**
2 states	9.03
3 states	8.92
4 states	8.11
5 states	**7.82**
6 states	8.18
7 states	8.66

Table 4.7: Values of DIC for different Normal HMMs for the US real interest rates, for the highest extreme quantile. The best model is indicated with bold characters.

The estimated Normal parameters (mean μ and precision κ) for the best model (5-state Normal HMM) are shown in table 4.8.

5-state Normal HMM				
Highest extreme quantile				
μ_1	μ_2	μ_3	μ_4	μ_5
4.78	6.52	8.64	11.05	15.40
κ_1	κ_2	κ_3	κ_4	κ_5
1.76	2.18	0.93	3.02	1.84

Table 4.8: Normal parameter estimates for US real interest rates.

4.4 The Normal Break-Point HMM

For the special case of the break-point models only a few changes are needed. Recall that a model with $m-1$ structural breaks can be expressed as an m-state hidden Markov model with transition matrix P given in section 3.2.1. The Forward-Backward algorithm of section 4.1 can still be used in order to simulate realizations of the hidden sequence of states. However, here the chain always starts at state 1 and the specific form of the transition matrix P makes calculations easier. We remind that here only one jump is allowed (by the definition of the break-point model) so, in each row of the transition matrix P there are only two non-zero probabilities, p_{ii} and p_{ii+1}, for $i = 1, 2, ..., m$. Additionally, these two probabilities sum to one for all i which means that for each i, there is a single unknown parameter p_{ii}, while $p_{ii+1} = 1 - p_{ii}$. For p_{ii} we assume a Beta prior distribution with parameters p and q, that is

$$p_{ii} \sim Beta(p, q)\ , i = 1, 2, ..., m,$$

with probability distribution function

$$\pi(p_{ii}) = \frac{\Gamma(a+b)}{\Gamma(a)+\Gamma(b)} p_{ii}^{p-1}(1-p_{ii})^{q-1} \propto p_{ii}^{p-1}(1-p_{ii})^{q-1}.$$

The assumptions concerning the prior distributions of the other parameters remain the same. Therefore, we have

$$\pi(\mu) = \prod_{i=1}^{m} \pi(\mu_i) \propto exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}$$

$$\pi(\kappa) = \prod_{i=1}^{m} \pi(\kappa_i) \propto \prod_{i=1}^{m} \kappa_i^{a-1} exp\left\{-b\kappa_i\right\}$$

Moreover, the form of the conditional likelihood of $\mathbf{y}^T$ given $\mathbf{x}^T$ slightly changes since, under the break-point model, the chain starts at state 1 with probability 1. Therefore,

$$L(\mathbf{y}^T|\mathbf{x}^T,\mu,\sigma^2,P) = f_1(y_1)p_{x_1,x_2}f_{x_2}(y_2)...p_{x_{T-1},x_T}f_{x_T}(y_T) \tag{4.10}$$
$$= \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m}\prod_{t:x_t=i} f_i(y_t),$$

where n_{ii} is the number of times the chain remains at the same state i. Note that the chain moves from state i to state $i+1$ just once. The joint posterior distribution of the model parameters is given by

$$f(\mu,\kappa,P|\mathbf{y}^T,\mathbf{x}^T) \propto L(\mathbf{y}^T|\mathbf{x}^T,\mu,\kappa,P)\pi(P)\pi(\mu)\pi(\kappa)$$
$$\propto \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m}\prod_{t:x_t=i} \sqrt{\kappa_i} exp\left\{-\frac{\kappa_i}{2}(y_t-\mu_i)^2\right\} \prod_{i=1}^{m} p_{ii}^{p-1}(1-p_{ii})^{q-1} \times$$
$$\times exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i-\xi)^2\right\} \prod_{i=1}^{m}\prod_{i=1}^{m} \kappa_i^{a-1} exp\left\{-b\kappa_i\right\}.$$

The full conditionals of the Normal parameters are obtained as before. For p_{ii} we now have a Beta full conditional posterior distribution with parameters $p_2 = p + n_{ii}$ and $q_2 = q + 1$:

$$\pi(p_{ii}|\mathbf{y}^T,\mathbf{x}^T,\mu,\kappa) \equiv Beta(p+n_{ii},q+1)\ , i = 1,2,...,m.$$

Note that, in the case of a break-point model, it is of particular interest to infer the times when the structural breaks occurred. Having simulated realizations of the hidden sequence of states $\mathbf{x}^T$ this is straightforward, since we just need to record the times when the transitions from state i to state $i+1$ occurred.

Details for the MCMC algorithm and the posterior distribution calculations, for the Normal break-point HMM, can be found in Appendix B.

4.4.1 Prior Specification

For the Normal break-point hidden Markov model the prior specification of the model parameters is similar to section 4.1.2. The only change is the prior distribution of each line of the matrix P. Here we have 2 elements in each line and as a consequence we need to assume a prior distribution only for one of them as they sum to 1. The prior we choose in this case is Beta with parameters $p = 0.5, q = 0.1$. Therefore, we have

$$p_{ii} \sim Beta(0.5,0.1)\ , i = 1,2,...,m.$$

4.4.2 Normal Break-Point HMM with a cubic model fit of the extreme quantiles

Let us consider a Normal break-point HMM, as described in section 4.4, and assume a cubic model fit for the extreme quantiles. We work exactly as in section 4.2.2 and by using linear programming we obtain an estimation of the parameters of the cubic model. Obviously, we get the same models as in section 4.2.2 :

$$\tilde{y}_i = -6.5 - 0.081x_i + 0.003x_i^2 + 0.000045x_i^3 + \varepsilon_i,$$

$$\tilde{y}_i = 3.3 + 0.058x_i + 0.0049x_i^2 + 0.000073x_i^3 + \varepsilon_i,$$

$i = 1, 2, ..., n$, for the lowest ($\tau = 0$) and highest ($\tau = 1$) extreme quantile, respectively. We choose $\varepsilon_i \sim N(0, 1)$. Based on the first model, we simulate data which, obviously, correspond to the lowest extreme quantile of our data set and then we apply our MCMC algorithm. We follow the same method for the second model (highest extreme quantile).

We run our MCMC algorithm, as described in section 4.4, for 0, 1, 2 and 3 structural breaks (these breaks correspond to 1, 2, 3 and 4 hidden states, respectively). This happens because we are more interested in finding the date and the number of those structural breaks, rather than the number of the hidden states. We choose the best model based on the DIC (table 4.9). It is found that a single break-point Normal HMM models better the highest extreme quantile and a 2 break-point Normal HMM models better the lowest extreme quantile.

In figure 4.6 we can see the estimated dates of the break-points (for both extreme quantiles) and their histograms, based on our simulated sample values of the break-points. Additionally, we can see that both extreme quantiles are affected by a structural break, which is around 1980. The estimated parameters for the best break-point HMMs, for both extreme quantiles, are shown in table 4.10.

Models	**DIC**
0 break-points	2.45
1 break-point	**2.07**
2 break-points	2.55
3 break-points	3.22
0 break-points	2.71
1 break-point	2.66
2 break-points	**1.88**
3 break-points	2.92

Table 4.9: Values of DIC for different Normal break-point HMMs for the US ex-post real interest rates, for the highest and lowest extreme quantile, respectively. The best modes are indicated with bold characters.

1 break-point Normal HMM			
Highest extreme quantile			
μ_1	κ_1	μ_2	κ_2
4.03	1.61	13.17	1.18

2 break-points Normal HMM					
Lowest extreme quantile					
μ_1	κ_1	μ_2	κ_2	μ_3	κ_3
-6.22	1.31	-1.87	0.91	0.26	0.55

Table 4.10: Normal break-point HMM parameter estimates for US ex-post real interest rates, using a cubic model fit for the extreme quantiles.

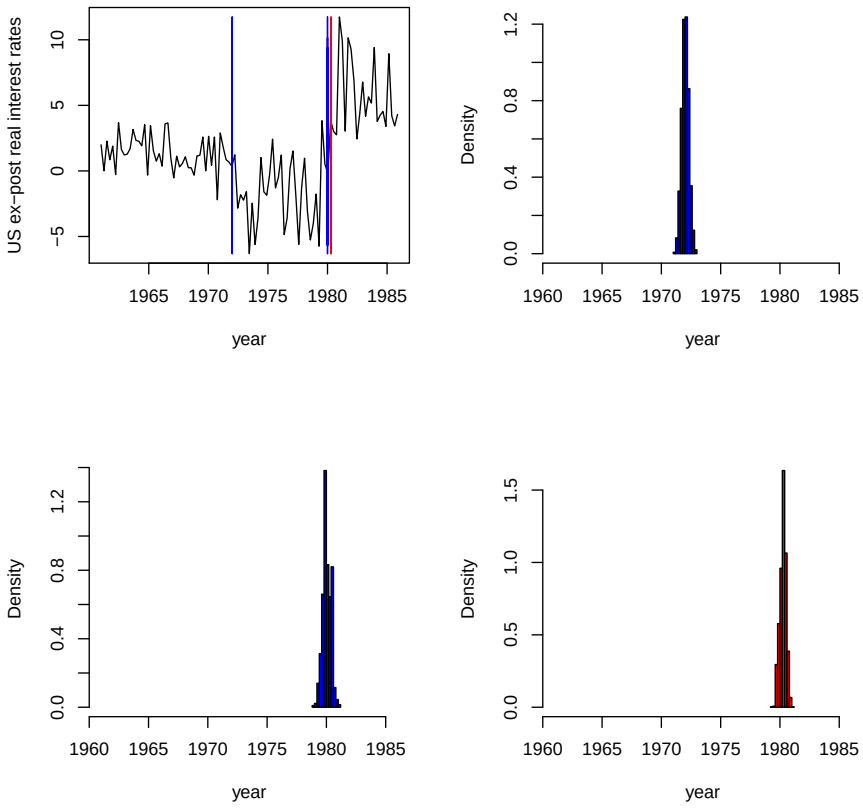

Figure 4.6: Dates and histograms of the break points of the US ex-post real interest rates (for the Normal break-point HMMs). The blue line corresponds to the break-points of the lowest extreme quantile and the red line corresponds to the break-point of the highest extreme quantile.

4.5 The m-state ALD hidden Markov model

Another way to perform Bayesian extreme quantile regression on hidden Markov models is by using the asymmetric Laplace distribution (ALD) instead of the Normal distribution. Let $\mathbf{y}^T = (y_1, y_2, ...y_T)$ be a sample of observations from an m-state ALD hidden Markov model. We assume that $\{X_t\}$ (the hidden underlying process) is a m-state Markov chain taking the values $1, 2, 3, ...m$, with transition matrix P and stationary distribution $\pi = (\pi_1, \pi_2, ...\pi_m)$. The distribution associated with the i^{th} state of this Markov chain is the asymmetric Laplace distribution

$$f_i(y) = f_L(y|\mu_i).$$

Therefore,

$$f_i(y) = \tau(1-\tau)exp\{p_\tau(y-\mu_i)\},$$

where τ is the quantile. Following the same procedures as in the Normal hidden Markov model, we can obtain the likelihood of the observed data $\mathbf{y}^T$ from 4.1 and we can also implement the Forward-Backward algorithm, by using f_L instead of f_N. The forward variables are given by 4.2 and 4.3. The likelihood within the Forward-Backward algorithm is given by 4.4 and the state at time t is given by 4.5 and 4.6. However, in this case we have less parameters to estimate, because by using f_L our parameters are $\theta = (P, \mu)$ and not $\theta = (P, \mu, \kappa)$ like before.

4.5.1 Gibbs and Metropolis-Hastings sampling for ALD HMMs

In the case of the ALD hidden Markov model we can construct another algorithm, which, at each iteration, updates the latent data $\mathbf{x}^T$ of the hidden Markov model given the model parameters $\theta = (\mu, P)$ and then updates θ given $\mathbf{x}^T$. Again, realizations of the hidden sequence of states $\mathbf{x}^T$ given θ are simulated using the Forward-Backward algorithm of section 4.1 (using f_L instead of f_N). However, there is a small change in updating the model parameters. This time only P is updated via Gibbs sampling. The parameter μ is updated via Metropolis-Hastings sampling, because its posterior distribution is of unknown form.

We start from the prior specification for the parameters $\theta = (\mu, P)$, as we did with the Normal HMM. We begin from the matrix P and denote as $p_{i.}$ the i^{th} row of the matrix, for $i = 1, 2, ..., m$. It is assumed that each row $p_{i.}$ follows a Dirichlet distribution with parameter $\omega = (\omega_1, \omega_2, ..., \omega_m)$.

$$p_{i.} \sim Dir(w)\ , i = 1, 2, ..., m.$$

Then, we assume a conjugate Normal prior distribution for each mean μ_i with mean ξ and variance λ^{-1}, that is

$$\mu_i \sim N(\xi, \lambda^{-1}) \, , i = 1, 2, ..., m.$$

Therefore, for $i = 1, 2, ..., m$ we have

$$\pi(p_{i.}) = \frac{1}{B(\omega)} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} \propto \prod_{j=1}^{m} p_{ij}^{\omega_j - 1}, \tag{4.11}$$

$$\pi(\mu_i) = \frac{\sqrt{\lambda}}{\sqrt{2\pi}} exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\} \propto exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\}.$$

The likelihood of the observed data $\mathbf{y}^T$ given the hidden sequence of states $\mathbf{x}^T$, under an m-state ALD hidden Markov model is given by

$$\begin{aligned} L(\mathbf{y}^T|\mathbf{x}^T, \mu, P) &= \pi(x_1) f_{x_1}(y_1) p_{x_1,x_2} f_{x_2}(y_2) ... p_{x_{T-1},x_T} f_{x_T}(y_T) \\ &= \pi(x_1) \underbrace{\prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}}}_{A} \underbrace{\prod_{i=1}^{m} \prod_{t:x_t=i} f_i(y_t)}_{B}, \end{aligned} \tag{4.12}$$

where $f_i(y_t)$, $i = 1, 2, ..., m$, is the probability distribution function of the asymmetric Laplace distribution associated with the i^{th} state and n_{ij} is the number of times the chain passes from state i to state j:

$$n_{ij} = \sum_{t=1}^{T} I(x_t = i, x_{t+1} = j).$$

Again, the likelihood is written as a product of two terms, each involving a subset of the model parameters. From factor A we can make inference about the parameters (elements) of the matrix P, while from B we can make inference about the parameter μ of the ALD.

The joint posterior distribution of the model parameters given the observed and the unobserved data of the hidden Markov model is given by

$$\begin{aligned} f(\mu, P|\mathbf{y}^T, \mathbf{x}^T) &\propto L(\mathbf{y}^T|\mathbf{x}^T, \mu, P)\pi(P)\pi(\mu) \\ &\propto \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} exp\left\{-p_\tau(y_t - \mu_i)\right\} \times \\ &\times \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} exp\left\{-\frac{\lambda}{2} \sum_{i=1}^{m} (\mu_i - \xi)^2\right\}. \end{aligned}$$

Then, we are able to obtain the full conditional posterior distributions of the parameters $p_{i.}$ and μ_i , each being proportional to $f(\mu, P|\mathbf{y}^T, \mathbf{x}^T)$ regarded as a function of the respective parameter only.

Similarly, the full conditional of $p_{i.}$, the i^{th} row of the matrix P, $i = 1, 2, ..., m$, is a Dirichlet distribution with parameters $(n_{i.} + \omega)$, where $n_{i.} = (n_{i1}, n_{i2}, ..., n_{im})$:

$$\pi(p_{i.}|\mathbf{y}^T, \mathbf{x}^T, \mu) \equiv Dir(n_{i.} + \omega). \tag{4.13}$$

The full conditional of μ_i, $i = 1, 2, ..., m$, is given by the following formula

$$\pi(\mu_i|\mathbf{y}^T, \mathbf{x}^T, P) \propto exp\left\{\sum_{t:x_t=i}\left[(y_t - \mu_i)(\tau - I_{(-\infty,0)}(y_t - \mu_i))\right] + \frac{\lambda}{2}(\mu_i - \xi)^2\right\}.$$

This posterior distribution is clearly of unknown form. Therefore, Metropolis-Hastings sampling is used for the parameter μ.

Details for the MCMC algorithm and the posterior distribution calculations, for the ALD HMM, can be found in Appendix C.

4.5.2 Prior Specification

In the case of the ALD hidden Markov model the prior distributions for $p_{i.}$, μ_i are exactly the same with the case of the Normal hidden Markov model (section 4.1.2). However, this time we do not have to estimate the precision κ_i.

4.5.3 ALD HMM for the US ex-post real interest rates

Now, let us consider an ALD hidden Markov model as described in section 4.5. In this case, there is no need of using linear programming to fit the extreme quantiles of the series, because we can specify the quantile we are interested in, within the MCMC algorithm, by fixing τ, the parameter of the ALD. However, τ has to be between 0 and 1, as defined in the asymmetric Laplace distribution. Therefore, we can approximate the extreme quantiles by using $\tau = 0.001 \approx 0$ and $\tau = 0.999 \approx 1$. Again, we run our MCMC algorithm, as described in section 4.5.1, assuming 2,3,4 and 5 states for both extreme quantiles and the best model is chosen based on the DIC. Table 4.11 shows that the best model is a 3-state ALD hidden Markov model (figure 4.7) for both quantiles.

The estimated parameters of the ALD, for the best models (3-state ALD HMM, for both quantiles) are shown in table 4.12.

Models	**DIC**
2-states HMM	5.45
3-states HMM	**4.09**
4-states HMM	4.88
5-states HMM	5.04
2-states HMM	5.81
3-states HMM	**4.22**
4-states HMM	5.12
5-states HMM	5.67

Table 4.11: Values of DIC for different ALD HMMs for the US ex-post real interest rates, for the highest and lowest extreme quantiles, respectively. The best models are indicated with bold characters.

3-state ALD HMM		
Highest extreme quantile		
μ_1	μ_2	μ_3
6.96	10.12	14.93
Lowest extreme quantile		
μ_1	μ_2	μ_3
-7.25	-4.34	-0.87

Table 4.12: ALD parameter estimates for US ex-post real interest rates.

4.5.4 ALD HMM for the US real interest rates

Let us consider an ALD hidden Markov model. We need to specify the extreme quantiles of the series, by fixing the parameter τ of the asymmetric Laplace distribution. The distribution needs $0 < \tau < 1$, so we choose $\tau = 0.001$ for the lowest extreme quantile and $\tau = 0.999$ for the highest extreme quantile. We consider up to 7 hidden states and we use the DIC to determine which model is the best. Table 4.13 shows that for both extreme quantiles, the series is better described by a 4-state ALD hidden Markov model (figure 4.8).

The estimated ALD parameters for the best models (4-state ALD HMM) are shown in table 4.14.

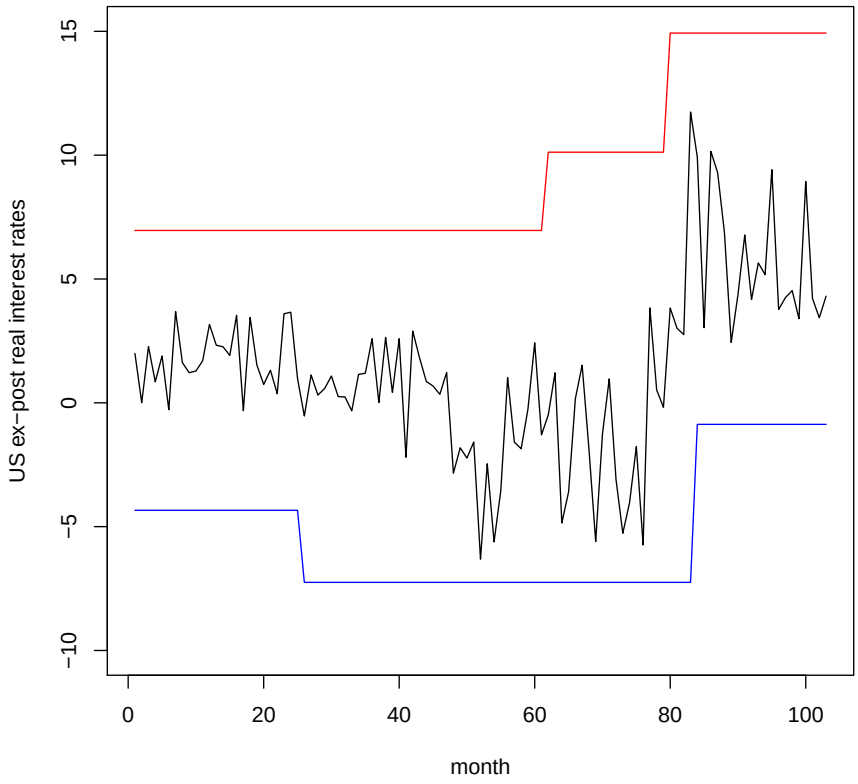

Figure 4.7: Model fit of the 3-state ALD HMM for the US ex-post real interest rates, for both extreme quantiles. The red line corresponds to the highest extreme quantile and the blue line corresponds to the lowest extreme quantile.

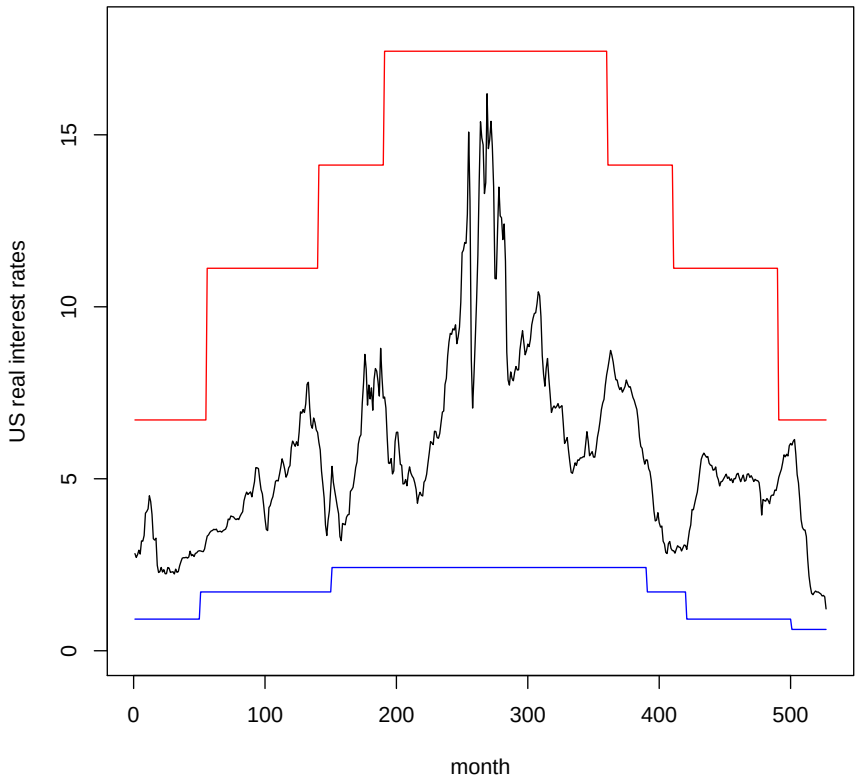

Figure 4.8: Model fit of the 4-state ALD HMM for the US real interest rates. The red line corresponds to the highest extreme quantile and the blue line corresponds to the lowest extreme quantile.

Models	**DIC**
2 states	8.54
3 states	8.16
4 states	**8.02**
5 states	8.23
6 states	8.87
7 states	9.22
2 states	8.89
3 states	8.41
4 states	**8.34**
5 states	9.12
6 states	9.05
7 states	8.97

Table 4.13: Values of DIC for different ALD HMMs for the US real interest rates, for the highest and lowest extreme quantiles, respectively. The best models are indicated with bold characters.

4-state ALD HMM			
Highest extreme quantile			
μ_1	μ_2	μ_3	μ_4
6.71	11.12	14.12	17.43
Lowest extreme quantile			
μ_1	μ_2	μ_3	μ_4
0.62	0.92	1.71	2.42

Table 4.14: ALD parameter estimates for US real interest rates.

4.6 The ALD Break-Point HMM

Again, we work similarly to the Normal break-point hidden Markov model and for p_{ii} we assume a Beta prior distribution with parameters p and q. That is

$$p_{ii} \sim Beta(p, q)\ , i = 1, 2, ..., m,$$

and for μ we assume a Normal distribution with parameters τ and ξ. That is

$$\mu_i \sim N\left(\xi, \lambda^{-1}\right)\ , i = 1, 2, ..., m.$$

Therefore, we have

$$\pi(p_{ii}) = \frac{\Gamma(a+b)}{\Gamma(a)+\Gamma(b)} p_{ii}^{p-1}(1-p_{ii})^{q-1} \propto p_{ii}^{p-1}(1-p_{ii})^{q-1},$$

$$\pi(\mu_i) \propto exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\}.$$

The likelihood is obtained from

$$L(\mathbf{y}^T|\mathbf{x}^T, \mu, P) = f_1(y_1)p_{x_1,x_2}f_{x_2}(y_2)...p_{x_{T-1},x_T}f_{x_T}(y_T) \tag{4.14}$$

$$= \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m} \prod_{t:x_t=i} f_i(y_t),$$

where n_{ii} is the number of times the chain remains at the same state i. Note that the chain moves from state i to state $i+1$ just once.

The joint posterior distribution of the model parameters is given by

$$f(\mu, P|\mathbf{y}^T, \mathbf{x}^T) \propto L(\mathbf{y}^T|\mathbf{x}^T, \mu, \kappa, P)\pi(P)\pi(\mu)$$

$$\propto \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m} \prod_{t:x_t=i} exp\left\{p_\tau(y_t - \mu_i)\right\} \times$$

$$\times \prod_{i=1}^{m} p_{ii}^{p-1}(1-p_{ii})^{q-1} exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}.$$

For the full conditional of p_{ii} we have

$$p_{ii} \sim Beta(p + n_{ii}, q + 1),$$

and the full conditional of μ_i is given by

$$\pi(\mu_i|\mathbf{y}^T, \mathbf{x}^T, P) \propto exp\left\{ \sum_{t:x_t=i} \left[(y_t - \mu_i)(\tau - I_{(-\infty,0)}(y_t - \mu_i))\right] + \frac{\lambda}{2}(\mu_i - \xi)^2 \right\}.$$

Clearly, like in the previous section, Gibbs and Metropolis-Hastings sampling are required for updating P and μ, respectively.

Details for the MCMC algorithm and the posterior distribution calculations, for the ALD break-point HMM, can be found in Appendix C.

4.6.1 Prior Specification

In the case of the ALD break-point hidden Markov model, the prior distributions for the parameters are exactly the same with the ones described in section 4.4.1. However, we do not need to estimate the precision κ_i.

4.6.2 ALD Break-Point HMM for the US ex-post real interest rates

Let us consider an ALD break-point HMM, as described in section 4.6. This time we do not have to model the extreme quantiles, as we can fix τ, the parameter of the asymmetric Laplace distribution, within our MCMC algorithm. However, we are not able to work for $\tau = 0$ and $\tau = 1$, because the asymmetric Laplace distribution needs $0 < \tau < 1$. Therefore, we try to approximate those (extreme) values by using $\tau = 0.001 \approx 0$ and $\tau = 0.999 \approx 1$. We run our MCMC algorithm, for both extreme quantiles, for $0, 1, 2$ and 3 structural breaks. We choose the best break-point model based on the DIC (table 4.15). The most appropriate models are: a single break-point ALD HMM for $\tau = 0.999$ and a 2 break-point ALD HMM for $\tau = 0.001$.

Figure 4.9 shows the estimated dates of those break-points (for both extreme quantiles) and their histograms, based on our simulated sample values of the break-points. Additionally, we can see that both extreme quantiles are affected by a structural break, which is probably the same. Our evidence for this assumption is that the dates of that structural break are very close. The estimated date using a single break-point ALD HMM, for the highest extreme quantile, is 1979 and the estimated date using a 2 break-point ALD HMM, for the lowest extreme quantile, is 1981. The estimated parameters for the best models, for both extreme quantiles, are shown in table 4.16.

Models	**DIC**
0 break-points	3.21
1 break-point	**2.13**
2 break-points	2.72
3 break-points	3.01
0 break-points	2.87
1 break-point	2.98
2 break-points	**1.93**
3 break-points	3.11

Table 4.15: Values of DIC for different ALD break-point HMMs for the US ex-post real interest rates, for the highest and lowest extreme quantile, respectively. The best modes are indicated with bold characters.

1 break-point ALD HMM	
Highest extreme quantile	
μ_1	μ_2
6.76	10.08

2 break-points ALD HMM		
Lowest extreme quantile		
μ_1	μ_2	μ_3
-7.36	-3.59	-0.21

Table 4.16: ALD break-point HMM parameter estimates for US ex-post real interest rates.

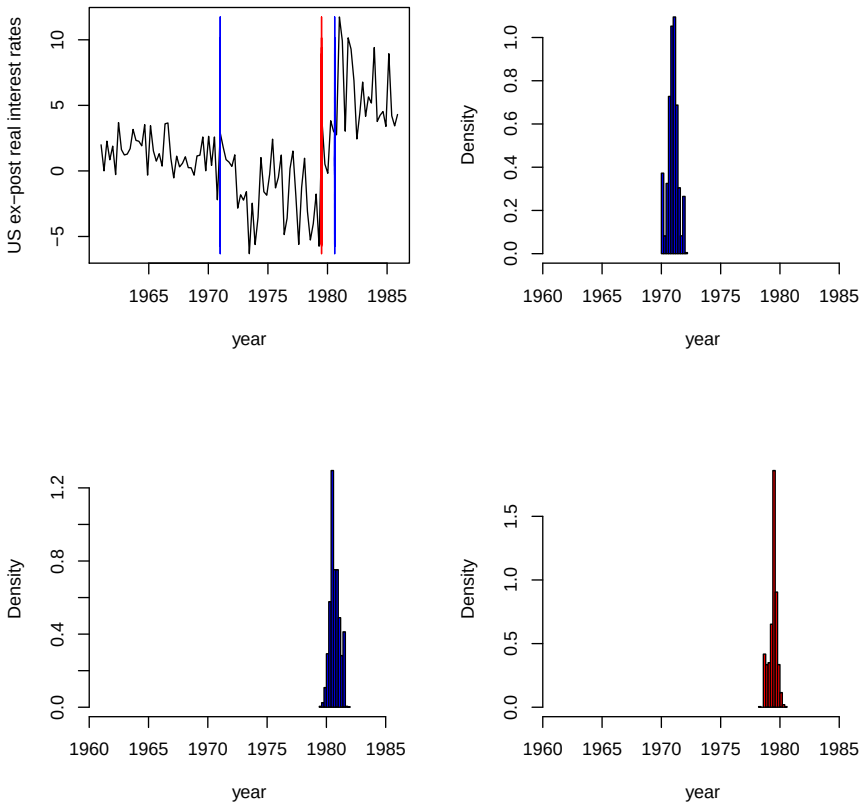

Figure 4.9: Dates and histograms of the break points of the US ex-post real interest rates (for the ALD break-point HMMs). The blue line corresponds to the break-points of the lowest extreme quantile and the red line corresponds to the break-point of the highest extreme quantile.

4.7 Deviance Information Criterion

The deviance information criterion (DIC) was introduced by Spiegelhalter et al. (2002) as a Bayesian model comparison criterion and it is directly inspired by linear and generalized linear models. However, it can be extended to models like mixtures of distributions, even if, in these cases, there are some inconsistencies in the definition of DIC, which were described by Delorio and Robert (2002). It is a hierarchical modeling generalization of the AIC (Akaike information criterion) and BIC (Bayesian information criterion) and it is an asymptotic approximation as the sample size becomes large. Additionally, it is particularly useful in Bayesian model selection problems where the posterior distributions of the model parameters have been obtained by Markov chain Monte Carlo (MCMC) simulations.

First let us define the deviance of the likelihood of the data $\mathbf{y}$ given the parameters θ as

$$D(\theta) = -2logL(\mathbf{y}|\theta). \tag{4.15}$$

Note that the likelihood $L(\mathbf{y}|\theta)$ includes all the normalizing constants. Then, the mean deviance is

$$\bar{D} = \overline{D(\theta)} = E_{\theta|\mathbf{y}}(D) = E(D(\theta)|\mathbf{y}).$$

The mean deviance can be regarded as a Bayesian measure of fit. Then, we define the effective dimension (or effective number of parameters) p_D as

$$p_D = \overline{D(\theta)} - D(\tilde{\theta}),$$

where $\tilde{\theta}$ is an estimate of θ depending on the data $\mathbf{y}$. The posterior mean $\bar{\theta}$ is often a natural choice for $\tilde{\theta}$ so,

$$\tilde{\theta} = \bar{\theta} = E(\theta|\mathbf{y}),$$

but the posterior mode or median can also be justified as an alternative choice. DIC takes into account the fit of the data to the model and the complexity of the model. The fit is measured by the mean deviance $\bar{D(\theta)}$ and the complexity is measured by the effective dimension p_D. Therefore, DIC takes the form

$$DIC = \overline{D(\theta)} + p_D = 2\overline{D(\theta)} - D(\bar{\theta}).$$

Clearly, using the equation (4.15) we have

$$\overline{D(\theta)} = -2log\overline{L(\mathbf{y}|\theta)},$$

$$D(\bar{\theta}) = -2logL(\mathbf{y}|\bar{\theta}).$$

Models with smaller DIC are better supported by the data.

In complicated models, where the expectations required for the calculation of the DIC are not available in closed form, it is possible to compute its value using the output of an MCMC algorithm for the estimation of the model parameters (see Spiegelhalter et al. 2002). In this thesis we use DIC to justify the choice of the best models. In the MCMC algorithms we used to analyze the data, we have simulated values for $D(\theta)$ and θ. As a consequence, the mean deviance $\overline{D(\theta)}$ was easily approximated by taking the sample mean of the simulated values of $D(\theta)$ and $D(\bar{\theta})$ was approximated by plugging in the sample mean of the simulated values of θ.

Chapter 5

Kalman Filter for Continuous State-Space HMMs

5.1 Hidden Markov models with continuous latent variables

In a previous chapter we showed how to deal with discrete-time finite (discrete) state-space hidden Markov models. In other words, hidden Markov models where the latent variables are discrete. Now, let us consider the case of a discrete-time continuous state-space hidden Markov model. This means that for this hidden Markov model the latent variables are continuous. We assume that the state of the chain can be described at any time by a m-vector of state variables **x**, which cannot be observed directly and at each time step a n-vector of observations **y** is produced by the system. Additionally, the hidden state is assumed to change according to a Markov chain of order 1. That means that a state at a time point t depends only on the state at a time point $t-1$, exactly as we described in the case of a discrete-time finite state-space hidden Markov model. In other words, the Markov model has a memory of size 1. The observed vector $\mathbf{y}^T = (y_1, y_2, ..., y_T)$ is generated from the current state by a simple linear observation process.

We can write:

$$x_{t+1} = Ax_t + w_t, \tag{5.1}$$

$$y_t = Cx_t + v_t, \tag{5.2}$$

for $t = 1, 2, ..., T$, where w_t and v_t are random variables, which represent the process and measurement noise, respectively. They are assumed to be independent of each other and of the values of **x** and

y. Both of them are assumed to be white (uncorrelated from time step to time step) and spatially gaussian distributed with zero mean and covariance matrices, which are denoted as Q and R, respectively. A is the $m \times m$ state transition matrix, which relates the state at the previous time t to the current step $t+1$. B is the $n \times m$ observation (generative) matrix, which relates the state at the current time step t to the measurement y_t.

Notes on model assumptions

Despite the fact that the random variables w and v are indexed with t, they do not have any knowledge of the time index. X_t is considered to be a Gauss-Markov random process of order 1, since the process noise is Gaussian and its dynamics are linear.

The noise processes w_t and v_t are very important and essential elements of the model because: a) without w_t, the state x_t would always either converge exponentially to zero, or converge exponentially in the direction of the leading eigenvector of the matrix A. b) without v_t the state would not be hidden.

All of the structure in matrix Q can be moved into the matrices A and C, which means that we can work with models where Q is the identity matrix, without loss of generality. In other words, for any model where Q is not the identity matrix, we can generate an exactly equivalent model, such that the new covariance matrix is the identity matrix. This happens due to the fact that Q is symmetric positive semi-definite, since it is a covariate matrix, and can be diagonalized to the form PDP^T (where P is a rotation matrix of eigenvectors and D is a diagonal matrix of eigenvalues).

The components of the state vector can be arbitrarily reordered just by swapping the columns of the matrices C and A. An ordering choice based on the norms of the columns of the matrix C can resolve the existing degeneracy of the model.

The matrices A and C and the covariance matrices Q and R might change with each time step measurement. However, we assume they are constant.

The main idea of the model described in this section is that the hidden state sequence should be an informative explanation of the observation sequence. Using dynamical and noise models, the states should summarize the underlying causes of the data in a more clear way than the observations themselves. For this reason, state dimensions much smaller than the number of observations are preferred to work with.

5.2 Kalman filter

The Kalman filter is a mathematical method that provides an efficient computational means to estimate the state of a process, in a way that minimizes the mean of the squared errors. Specifically, the state of a process has to be estimated even though it cannot be directly measured. Instead, the available measurements (observations) are used in order to achieve this estimation. Obviously, the Kalman filter is used when we deal with models described in the previous section (equations 5.1 and 5.2).

There are two basic requirements when using this method. First, we want the average value of our state estimate to be equal to the average value of the time state. And second, we want a state estimate which varies from the true state as little as possible. Mathematically, we can rephrase these requirements as follows: we want the expected value of the estimate to be equal to the expected value of the state and we also want to find an estimator with the smallest possible error variance.

We can also describe the Kalman filter as an algorithm for efficiently performing exact inference in Bayesian models, like hidden Markov models, where the state space of the hidden states (latent variables) is continuous and where both latent and observed variables have a Gaussian distribution. Practically, the Kalman filter produces estimates of the true values of the measurements and their associated calculated values by predicting a value, estimating its uncertainty and computing a weighted average of the predicted value and the measured value. The value with the least uncertainty is given the most weight. The weights are calculated from the covariance, which is a measure of the estimated uncertainty of the prediction of the system's state. The weighted average results to a new state estimate that lies in between the predicted and the measured state, and has a better estimated uncertainty than either of those alone. This process is repeatedly performed in each time step, when the new estimate and its covariance "informs" the prediction used in the following iteration. This shows that the Kalman filter is a recursive algorithm and does not require the entire history of the system's states, but only the last estimate, in order to calculate a new state at a specific time step. This method produces estimates which tend to be closer to the true values than the original measurements, because the weighted average has a better estimated uncertainty than either of the values included in the weighted average.

The Kalman filter is also theoretically attractive, because it is the one, of all possible filters, that minimizes the variance of the estimation error. It is named after Rudolf E. Kalman, when he described a recursive solution to the discrete-data linear filtering problem in 1960, despite the fact that

Peter Swerling (1958) had developed a very similar algorithm. Sorenson (1970) and Maybeck (1979) provide us with an introduction to this method. Also, Gelb (1974), Grewal (1993), Lewis (1986), Brown (1992) and Jacobs (1993) provide us with useful information. In spite of having its roots on Karl Gauss's method of least squares (1795), Kalman filter was developed to solve the problem of spacecraft navigation for the Apollo space program. Kalman filter has found applications in space and military technology, including all forms of navigation (aerospace, land and marine), nuclear power plant instrumentation, demographic modeling, manufacturing, the detection of underground radioactivity and fuzzy logic and network training.

5.2.1 Computing the Kalman Filter

Let us start by using a hidden Markov model given by the equations 5.1 and 5.2. We define $\hat{\tilde{x}}_t$ as our a priori estimate of the state x_t, at time t, which takes into consideration all observations until y_{t-1} (all observations without y_t). Then, we define $\hat{x}_t$ as our a posteriori estimate of the state x_t, at time t, which takes into consideration the observation y_t as well (all observations until y_t). The a priori state estimate $\hat{\tilde{x}}_t$ is given by:

$$\hat{\tilde{x}}_t = A\hat{x}_{t-1}. \tag{5.3}$$

We can then define

$$\tilde{e}_t = x_t - \hat{\tilde{x}}_t \quad \text{(a priori estimate error)}$$

$$e_t = x_t - \hat{x}_t \quad \text{(a posteriori estimate error)}$$

and

$$\tilde{P}_t = E[\tilde{e}_t \tilde{e}_t^T] \quad \text{(a priori estimate error covariance)} \tag{5.4}$$

$$P_t = E[e_t e_t^T] \quad \text{(a posteriori estimate error covariance).} \tag{5.5}$$

The quantity that describes the discrepancy between the actual measurement and the predicted measurement is called residual and it is given by:

$$\text{Residual} = y_t - \hat{y}_t = y_t - C\hat{\tilde{x}}_t.$$

We need to find a way to obtain the a posteriori state estimate as a linear combination of the a priori state estimate and the residual. This is a way to correct our a priori state estimate and is given by:

$$\hat{x}_t = \hat{\tilde{x}}_t + K_t(y_t - C\hat{\tilde{x}}_t). \tag{5.6}$$

The matrix K_t is chosen in a way that minimizes the a posteriori error covariance. So, let us start by its definition:

$$P_t = E[(x_t - \hat{x}_t)(x_t - \hat{x}_t)^T]$$

$$= E(x_t\hat{x}_t^T - x_t x_t^T - \hat{x}_t\hat{x}_t^T - \hat{x}_t x_t^T)$$

$$= E[(x_t - (\hat{\tilde{x}}_t + K_t(Cx_t + v_t - C\hat{\tilde{x}}_t)))(x_t - (\hat{\tilde{x}}_t + K_t(Cx_t + v_t - C\hat{\tilde{x}}_t)))^T].$$

We expand the quantity inside $E[.]$ and we have

$$P_t = E[((x_t - \hat{\tilde{x}}_t)(I - K_tC) - K_tv_t)((x_t - \hat{\tilde{x}}_t)^T(I - K_tC)^T - v_t^T K_t^T)].$$

The measurement noise v_t is uncorrelated to the hidden state x_t, therefore uncorrelated to its transpose x_t^T, so we have $E(x_tv_t) = E(x_t^T v_t) = 0$. Using this property we have

$$P_t = (I - K_tC)E[(x_t - \hat{\tilde{x}}_t)(x_t - \hat{\tilde{x}}_t)^T](I - K_tC)^T + K_tE(v_tv_t^T)K_t^T$$

$$= (I - K_tC)\tilde{P}_t(I - K_tC)^T + K_tRK_t^T$$

$$= \tilde{P}_t - K_tC\tilde{P}_t - \tilde{P}_tC^TK_t^T + K_t(C\tilde{P}_tC^T + R)K_t^T. \tag{5.7}$$

Now, we need to minimize the a posteriori estimate covariance P_t. This is equivalent to minimizing the trace of P_t.

$$tr(P_t) = tr(\tilde{P}_t) - 2tr(K_tC\tilde{P}_t) + tr(K_t(C\tilde{P}_tC^T + R)K_t^T).$$

We take the derivative of $tr(P_t)$ with respect to K_t and set it equal to zero.

$$0 = \frac{\partial tr(P_t)}{\partial K_t} = -2(C\tilde{P}_t)^T + 2K_t(C\tilde{P}_tC^T + R) \Rightarrow$$

$$\Rightarrow K_t = \tilde{P}_tC^T(C\tilde{P}_tC^T + R)^{-1} \tag{5.8}$$

We multiply with $(C\tilde{P}_tC^T+R)K_t^T$ both left and right part of the above equation and we get:

$$K_t(C\tilde{P}_tC^T + R)K_t^T = \tilde{P}_tC^TK_t^T.$$

We replace the above equation to equation 5.7 and we get a simpler form for P_t:

$$P_t = \tilde{P}_t - K_tC\tilde{P}_t = (I - K_tC)\tilde{P}_t. \tag{5.9}$$

Now, we will try to find a form to express $\tilde{P}_t$, starting by its definition:

$$\tilde{P}_t = E[(x_t - \hat{\tilde{x}}_t)(x_t - \hat{\tilde{x}}_t)^T]$$

$$= E[(x_t - A\hat{x}_{t-1})(x_t - A\hat{x}_{t-1})^T]$$

$$= E[(Ax_{t-1} + w_{t-1} - A\hat{x}_{t-1})(Ax_{t-1} + w_{t-1} - A\hat{x}_{t-1})^T].$$

We expand the quantity inside $E[.]$ and we use the fact that the process noise w_t is uncorrelated to the state x_t and its transpose. That is $E(x_t w_t) = E(x_t^T w_t) = 0$. Therefore, we have:

$$\tilde{P}_t = E[A(x_{t-1} - \hat{x}_{t-1})(x_{t-1} - \hat{x}_{t-1})^T A^T + w_{t-1}w_{t-1}^T]$$

$$= AE[(x_{t-1} - \hat{x}_{t-1})(x_{t-1} - \hat{x}_{t-1})^T]A^T + E(w_{t-1}w_{t-1}^T)$$

$$= AP_{t-1}A^T + Q. \tag{5.10}$$

The equations 5.3, 5.6, 5.8, 5.9, and 5.10 are used for creating the algorithm. The computational origins of the Kalman filter can be found in Appendix D.

The Kalman filter estimates the hidden state by using a form of a predictor-corrector algorithm. First, it uses (time update) equations in order to project forward in time the current state estimate to obtain the a priori estimates (for hidden state and covariance) for the next time step. Then, it uses (estimation update) equations in order to combine the new observation and the a priori estimates to obtain the a posteriori estimates (for hidden state and covariance).

Prediction (Time Update)
$\hat{\tilde{x}}_t$ = $A\ \hat{x}_{t-1}$ $\tilde{P}_t$= $A\ P_{t-1}\ A^T$ + Q

Kalman Gain
$K_t = \tilde{P}_t C^T (C\tilde{P}_t C^T + R)^{-1}$

Correction (Estimation Update)
$\hat{x}_t$= $\hat{\tilde{x}}_t$+ K_t (y_t-C $\hat{\tilde{x}}_t$) P_t= (I-K_t C) $\tilde{P}_t$

Table 5.1: Kalman filter equations used to create a predictor-corrector algorithm.

Figure 5.1 shows how the Kalman filter performs the parameter update, for each time step t, by using the above formulas. The actual steps can be found in Appendix D.

$$\hat{x}_0, P_0 \longrightarrow \hat{\tilde{x}}_1, \tilde{P}_1 \longrightarrow K_1, y_1 \longrightarrow \hat{x}_1, P_1 \longrightarrow \quad ...$$
$$\longrightarrow \hat{\tilde{x}}_t, P_t \longrightarrow \hat{\tilde{x}}_{t+1}, \tilde{P}_{t+1} \longrightarrow K_{t+1}, y_{t+1} \longrightarrow \hat{x}_{t+1}, P_{t+1}$$

Figure 5.1: Kalman filter parameter update.

5.2.2 Kalman Smoothing

After applying the Kalman filter there is also another procedure, which enables us improve our estimates of the state variables x_t and the estimates of the covariance P_t. This is the Kalman smoothing and it takes all the measurements $\mathbf{y}^T = (y_1, y_2, ..., y_T)$ into consideration to improve the estimates, at each time step t. After Kalman smoothing the parameter estimates are less noisy (smoother) than before (after the Kalman filter). Kalman filtering and Kalman smoothing can be considered to be a kind of Forward-Backward algorithm. The equations used for Kalman filtering are the forward estimation equations and the equations of Kalman smoothing are the backward estimations equations. Those equations are:

$$\hat{x}'_t = \hat{x}_t + (P_t A^T \tilde{P}^{-1}_{t+1})(\hat{x}'_{t+1} - A\hat{x}_t)$$

$$P'_t = P_t + (P_t A^T \tilde{P}^{-1}_{t+1})(P'_{t+1} - \tilde{P}_{t+1})(P_t A^T \tilde{P}^{-1}_{t+1})^T,$$

for $t = T, T-1, ..., 1$, where $\hat{x}'_t$ is the smoothed state estimate at time t, using all observations and not only the observations until time t. Similarly, P'_t is the smoothed covariance estimate at time t, using all observations and not only the observations until time t. Kalman smoothing needs (initial) values for $\hat{x}'_{t+1}$ and P'_{t+1}. So, it uses the values of $\hat{x}_t$ and P_t, respectively, which were computed at the last time step T of the Kalman filter. Table 5.2 shows how the equations of Kalman filter and Kalman smoothing can be combined in order to create a Forward-Backward algorithm.

Forward Estimation
Prediction (Time Update)
$\hat{\tilde{x}}_t = A\hat{x}_{t-1}$ $\tilde{P}_t = AP_{t-1}A^T + Q$
Kalman Gain
$K_t = \tilde{P}_t C^T (C\tilde{P}_t C^T + R)^{-1}$
Correction (Estimation Update)
$\hat{x}_t = \hat{\tilde{x}}_t + K_t(y_t - C\hat{\tilde{x}}_t)$ $P_t = (I - K_t C)\tilde{P}_t$
Backward Estimation
Parameter Smoothing
$\hat{x}'_t = \hat{x}_t + (P_t A^T \tilde{P}_{t+1}^{-1})(\hat{x}'_{t+1} - A\hat{x}_t)$ $P'_t = P_t + (P_t A^T \tilde{P}_{t+1}^{-1})(P'_{t+1} - \tilde{P}_{t+1})(P_t A^T \tilde{P}_{t+1}^{-1})^T$

Table 5.2: Forward-Backward algorithm created by combining Kalman filtering and Kalman smoothing.

5.3 Applications

We also want to model these two data sets assuming a continuous state-space hidden Markov model. In order to do that we use Kalman filtering for the parameter estimations and then Kalman smoothing to improve those estimations.

Our observation sequence, $\mathbf{y}^T = (y_1, y_2, ...y_T)$, represents the observed real interest rates. The hidden underlying process $\{X_t\}$ is a m-state Markov chain taking the values $1, 2, 3, ...m$, with transition matrix P and stationary distribution $\pi = (\pi_1, \pi_2, ...\pi_m)$. The latent variables, $\mathbf{x}^T = (x_1, x_2, ...x_T)$, represent the financial regimes in the following way :

An observed real interest rate y_t $(1 \leq t \leq T)$ belongs to the financial segment m_1 $(1 \leq m_1 \leq m)$, if the hidden state $x_t = m_1$ occurs and y_t was generated according to the distribution associated with

the state x_t.

5.3.1 Discrete-time Continuous state-space HMM for the US ex-post real interest rates

In this section we model the US ex-post real interest rates by considering a discrete-time continuous state-space HMM. We create a new algorithm, which combines linear programming and Kalman filtering (and smoothing). As in sections 4.2.2 and 4.4.2, we model the extreme quantiles of the series using a cubic model. Using linear programming we obtain the parameters of the cubic model, for both extreme quantiles ($\tau = 0$ and $\tau = 1$). As a result, we get the following models:

$$\tilde{y}_i = -6.5 - 0.081x_i + 0.003x_i^2 + 0.000045x_i^3 + \varepsilon_i,$$

$$\tilde{y}_i = 3.3 + 0.058x_i + 0.0049x_i^2 + 0.000073x_i^3 + \varepsilon_i,$$

$i = 1, 2, ..., n$, for the lowest ($\tau = 0$) and highest ($\tau = 1$) extreme quantile, respectively. We choose $\varepsilon_i \sim N(0, 1)$.

Based on the first model, we simulate data which, obviously, correspond to the lowest extreme quantile of our data set and then we apply the Kalman filtering algorithm. We obtain estimates for the continuous hidden state and its variance and, then, we apply the Kalman smoothing algorithm, in order to improve our estimations. We follow the same method for the second model (highest extreme quantile). In figure 5.2 we can see this fitting.

5.3.2 Discrete-time Continuous state-space HMM for the US real interest rates

Now let us model the extreme quantiles of the US treasury bill real interest rates by considering a discrete-time continuous state space HMM. First, we need to obtain new simulated data, which correspond to the extreme quantiles, using a quadratic model fit. Using linear programming, as in section 4.3.1, we obtain the following quadratic models:

$$\tilde{y}_i = 3.52 - 0.00029x_i - 0.000032x_i^2 + \varepsilon_i,$$

$$\tilde{y}_i = 16.19 + 0.0014x_i - 0.00018x_i^2 + \varepsilon_i,$$

$i = 1, 2, ..., n$, for the lowest ($\tau = 0$) and highest ($\tau = 1$) extreme quantile, respectively. We choose $\varepsilon_i \sim N(0, 1)$. Then we apply Kalman filtering and Kalman smoothing to the new simulated data. In figure 5.3 we can see this fitting.

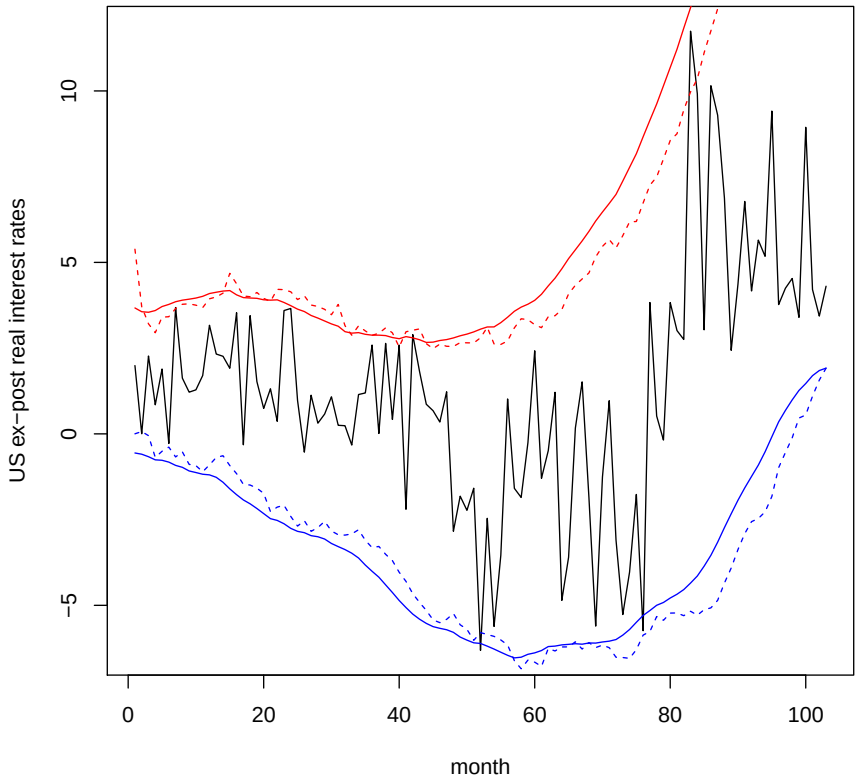

Figure 5.2: Model fit using Kalman filter (dashed line) and Kalman smoothing (solid line) for the lowest (blue lines) and highest (red lines) extreme quantiles, for the US ex-post real interest rates.

As in previous sections, we see that our model fits the lowest extreme quantile in a very good way, but it fails to model the shape of the highest extreme quantile. Therefore, we can follow the method we used in section 4.3.3 and simulate new data which correspond to the highest extreme quantile, by using two quadratic models. Using linear programming we obtain the following quadratic models:

$$\tilde{y}_i = 15.9 + 0.078x_i + 0.00013x_i^2 + \varepsilon_i, \quad i = 1, 2, ..., 269,$$

$$\tilde{y}_i = 16.82 - 0.106x_i + 0.00026x_i^2 + \varepsilon_i, \quad i = 270, 271, ..., 527.$$

We simulate new data based on these models and we apply Kalman filtering and Kalman smoothing. This fitting is much better, as we can see in figure 5.4.

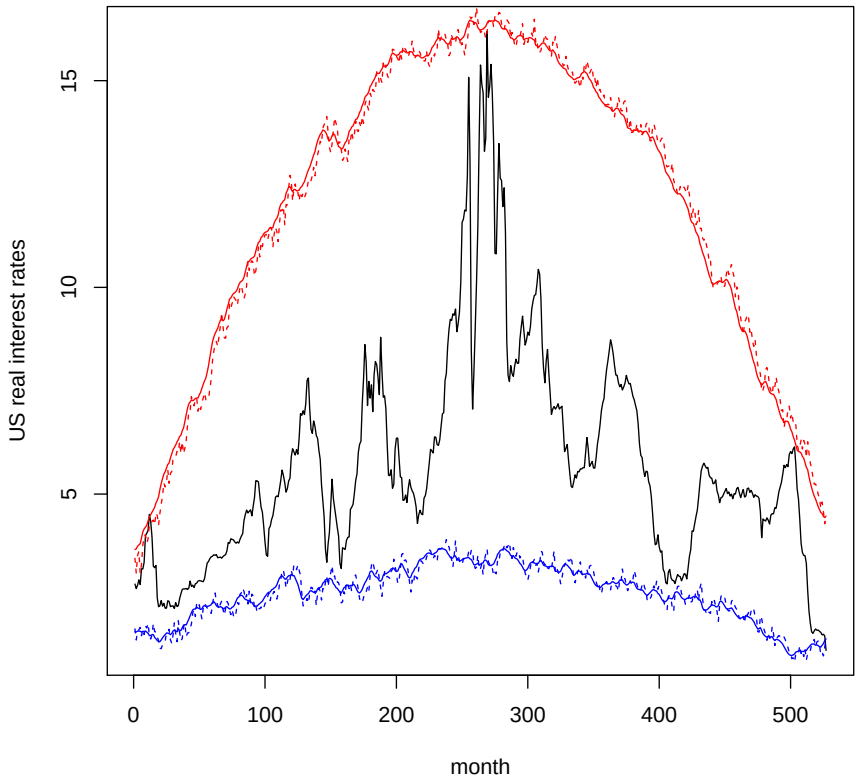

Figure 5.3: Model fit using Kalman filter (dashed line) and Kalman smoother (solid line) for the lowest (blue lines) and highest (red lines) extreme quantiles, for the US real interest rates. A quadratic model fit was assumed for both extreme quantiles.

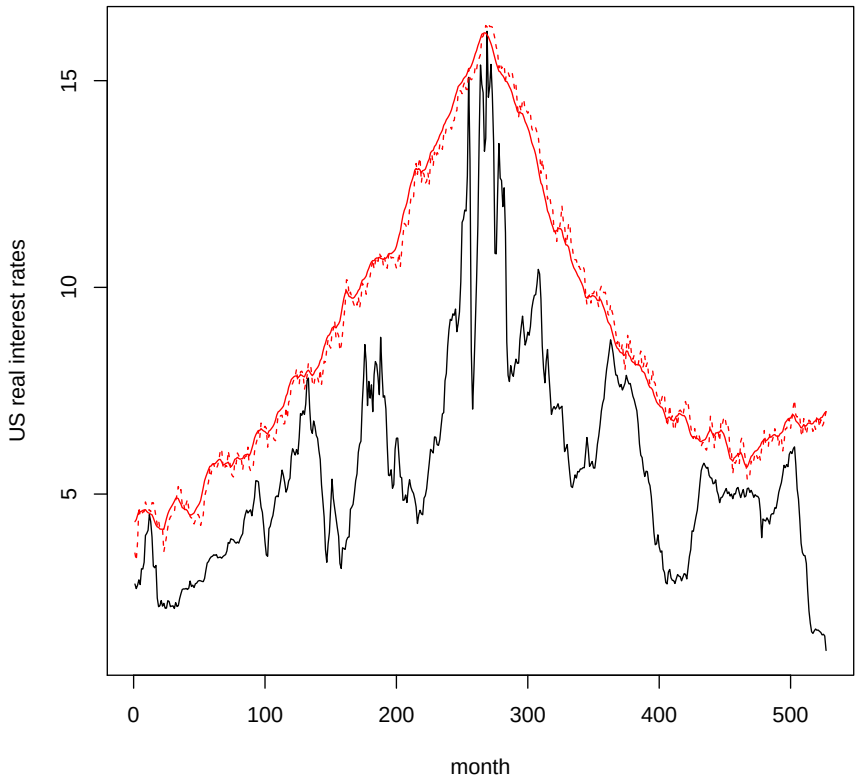

Figure 5.4: Model fit using Kalman filter (dashed line) and Kalman smoother (solid line) for the highest extreme quantile, for the US real interest rates. Two quadratic models were used to fit the highest extreme quantile.

Chapter 6

Comparison of the Proposed Methods for HMMs

In this chapter we present the findings of the different methods we applied on our data sets and we also make a comparison between those methods. Specifically, we present differences and similarities in terms of estimating the number of hidden states, the number and location of the break-points and the total number of parameters of the models. We also describe the general algorithms we used in order to implement Bayesian extreme quantile regression using Normal HMMs, Normal break-point HMMs and discrete-time continuous state-space HMMs. The reason why we concentrate in those methods is because their algorithms are more complex, due to the fact that they combine linear programming and MCMC methods.

6.1 Comparison of the HMM methods for the US ex-post real interest rates

For the analysis of the US ex-post real interest rates we used the following three models:
a) Normal HMM with a quadratic extreme quantile fit.
b) Normal HMM with a cubic extreme quantile fit.
c) Asymmetric Laplace Distribution (ALD) HMM.
For all models we have a good and fast convergence of the Markov chain. As an example of this convergence see traceplots 7.10 and 7.11, in Appendix F. All models can describe the lowest extreme

quantile of the data set by using 3 states. However, those states are different for every model. The highest extreme quantile of the data set is described by a different number of states, depending on the model. The Normal HMM with a quadratic extreme quantile fit uses 4 states, the Normal HMM with a cubic extreme quantile fit uses 2 states and the ALD HMM uses 3 states. Figure 6.1 shows that the Normal HMM with a cubic extreme quantile fit models the extreme quantiles of the series in the best way, because the fitting lines (hidden states) are closer to the data. However, we have to say that the first 25 data points for the lowest extreme quantiles and the first 45 data points for the highest extreme quantile are modeled in a better way by the Normal HMM with a quadratic extreme quantile fit. The ALD HMM is not as good as the other two models, but we have to point out that it has less parameters for estimation, as it does not contain the precision κ, which is contained in the Normal HMMs.

The fact that we need to obtain new simulated data, when we use Normal HMMs, instead of using the initial data, in the case of ALD HMMs, does not allow us to use the DIC criterion, because we use different kinds of data in order to calculate the likelihood. However, we can check the DIC only in order to compare the Normal HMM with a quadratic extreme quatile fit and the Normal HMM with a cubic extreme quantile fit. Comparing tables 4.1 and 4.3 we see that a cubic extreme quantile fit is more appropriate for our data set.

We have to point out, again, that a Normal HMM works for $\tau = 0$ and $\tau = 1$ as the lowest and highest extreme quantiles, respectively, when an ALD HMM works for the approximations $\tau = 0.001$ and $\tau = 0.999$.

The general algorithm for the Normal HMMs, which combines linear programming and MCMC methods, is the following:

1. Use linear programming to estimate the parameters $\boldsymbol{\beta}$, which model the data, for $\tau = 0$ and $\tau = 1$.
2. Use the estimated parameters $\hat{\boldsymbol{\beta}}$ to simulate new data $\tilde{\mathbf{y}}$, which correspond to $\tau = 0$ and $\tau = 1$.
3. Use MCMC algorithm to estimate (μ, κ, $\mathbf{P}$), given $\tilde{\mathbf{y}}$; (Appendix B).

The MCMC algorithm for the ALD HMM is described in Appendix C.

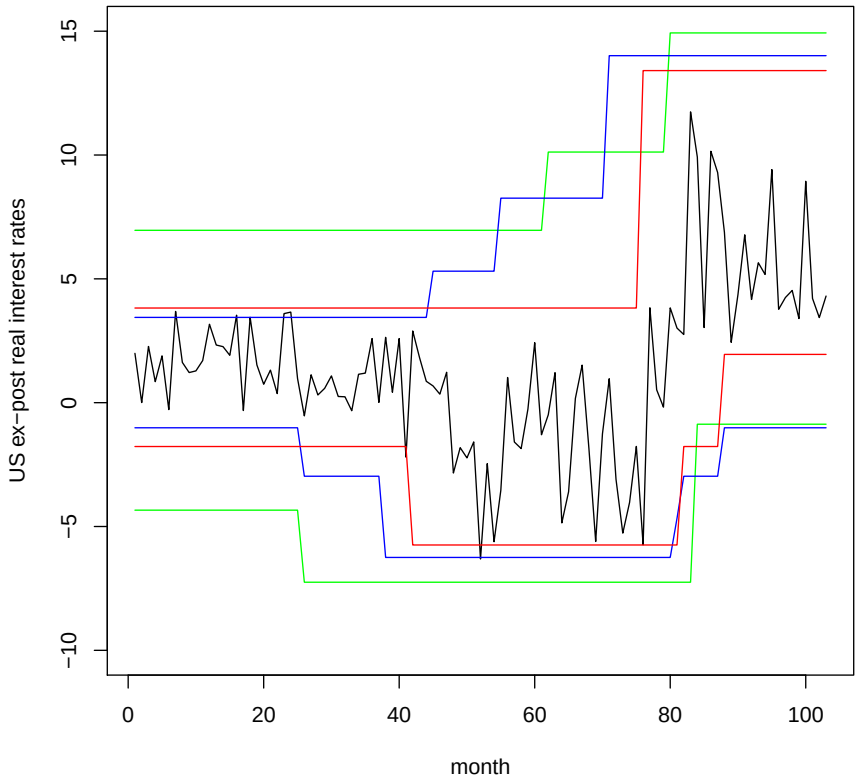

Figure 6.1: Model comparison for the US ex-post real interest rates. The red lines represent the Normal HMM with a cubic extreme quantile fit. The blue lines represent the Normal HMM with a quadratic extreme quantile fit. The green lines represent the ALD HMM.

6.2 Comparison of the Break-Point HMM methods

We considered two break-point HMMs for the US ex-post real interest rates, in order to estimate the number of possible structural changes and their dates. We started with a Normal break-point HMM (with a cubic model fit of the extreme quantiles) and then we used an ALD break-point HMM. The Normal break-point HMM enables us to work for $\tau = 0$ and $\tau = 1$, due to the cubic model fit of the extreme quantiles, using linear programming. On the other hand, the ALD break-point HMM works for $0 < \tau < 1$, due to the asymmetric Laplace distribution. As a consequence, we approximated the extreme quantiles using $\tau = 0.001 \approx 0$ and $\tau = 0.999 \approx 1$. All HMMs have a good and fast

convergence of the Markov chain and as an example see traceplots 7.12 and 7.13, in Appendix F.

Both break-point HMMs were found to have 1 break-point for the highest extreme quantile and 2 break-points for the lowest extreme quantile. Additionally, the dates of those break-points were very close (graphs 4.6 and 4.9). However, it is very important, especially for computational reasons, to say that the ALD break-point HMM has less parameters for estimation compared to the Normal break-point HMM. Specifically, the ALD break-point HMM estimates (μ_1, μ_2) for the highest extreme quantile and (μ_1, μ_2, μ_3) for the lowest extreme quantile. On the other hand, the Normal break-point HMM estimates $(\mu_1, \mu_2, \kappa_1, \kappa_2)$ for the highest extreme quantile and $(\mu_1, \mu_2, \mu_3, \kappa_1, \kappa_2, \kappa_3)$ for the lowest extreme quantile.

Again, in the case of the Normal break-point HMM the likelihood was calculated using new simulated data (which correspond to the highest and lowest extreme quantiles), obtained via linear programming, where as in the case of the ALD break-point HMM the likelihood was calculated using the initial data. As a consequence, we cannot perform a model choice based on the DIC criterion. Given that both methods provide us with similar results, it would be reasonable to say that the ALD break-point HMM is preferred to the Normal break-point HMM as it needs a less number of parameters to be estimated.

The general algorithm for the Normal break-points HMMs, which combines linear programming and MCMC methods, is the following:

1. Use linear programming to estimate the parameters $\boldsymbol{\beta}$, which model the data, for $\tau = 0$ and $\tau = 1$.
2. Use the estimated parameters $\hat{\boldsymbol{\beta}}$ to simulate new data $\tilde{\mathbf{y}}$, which correspond to $\tau = 0$ and $\tau = 1$.
3. Use MCMC algorithm to estimate (μ, κ, **P**), given $\tilde{\mathbf{y}}$; (Appendix B).

The MCMC algorithm for the ALD break-point HMM is described in Appendix C.

6.2.1 Discrete-time Continuous state-space HMM

We also analyzed both US ex-post real interest rates and US real interest rates using discrete-time continuous state-space HMMs. The general algorithm for these models, which combines linear programming and Kalman filtering (and smoothing) methods, is the following:

1. Use linear programming to estimate the parameters $\boldsymbol{\beta}$, which model the data, for $\tau = 0$ and $\tau = 1$.

2. Use the estimated parameters $\hat{\boldsymbol{\beta}}$ to simulate new data $\tilde{\mathbf{y}}$, which correspond to $\tau = 0$ and $\tau = 1$.

3. Use Kalman filter to estimate the values of the continuous hidden state and its covariance, ($\mathbf{x}_t$, $\mathbf{P}_t$), for the new data $\tilde{\mathbf{y}}$.

4. Use Kalman smoother to correct (improve) the previous estimations.

The Kalman filtering and smoothing algorithm for the discrete-time continuous state-space HMM is described in Appendix D.

6.2.2 Comparison of the HMM methods for the US real interest rates

In order to analyze the extreme quantiles of the US treasury bill real interest rates we used the following models:
a) Normal HMM with a quadratic extreme quantile fit.
b) Normal HMM with two quadratic highest extreme quantile fit.
c) ALD HMM.
For all HMMs we have a very good and fast convergence of the Markov chain. As an example of this convergence see traceplots 7.14, 7.15, 7.16 and 7.17, in Appendix F. Both extreme quantiles of the series are described by a different number of hidden states, depending on the model. The Normal HMM with a quadratic extreme quantile fit uses 3 states for both extreme quantiles, but the ALD HMM uses 4 states for both extreme quantiles. This means that the ALD HMM needs one more state, but it has only 4 parameters for estimation, $(\mu_1, \mu_2, \mu_3, \mu_4)$, for each extreme quantile, when the Normal HMM needs to estimate 6 parameters, $(\mu_1, \mu_2, \mu_3, \kappa_1, \kappa_2, \kappa_3)$, for each extreme quantile. The Normal HMM with two quadratic highest extreme quantile fit uses 5 states. That means estimating 10 parameters, (μ and κ, for 5 states), but it fits the highest extreme quantile of the series in the best possible way, compared to all the other HMMs (figure 4.5).

We also tried to model the highest extreme quantile of the data by a cubic model fit (figure 4.4), but that fit was very bad, as it could not model the shape of the data.

We have to point out, again, that a Normal HMM works for $\tau = 0$ and $\tau = 1$ as the lowest and highest extreme quantiles, respectively, when an ALD HMM works for the approximations $\tau = 0.001$

and $\tau = 0.999$.

The general algorithm which combines linear programming and MCMC methods, used for this data set, is similar to the algorithm of section 6.1.

Chapter 7

Discussion and Conclusion

In the first part of this thesis we performed Bayesian extreme quantile regression, in order to analyze various data sets. Particularly, we wanted to model the extreme quantiles of the data sets. Then, we were interested in comparing our method with the classical quantile regression approach, which uses linear programming. What makes this comparison even more interesting, is the fact that when the asymmetric Laplace distribution is used, in order to perform Bayesian extreme quantile regression, this corresponds to solving some minimization problems. And one of the most common and easy ways to do that is linear programming.

In the beginning we used three simulated data sets, which were obtained from three different models. Those models differ in the distribution of the error term. We assumed Uniform, Beta and Weibull distributions. Therefore, apart from comparing Bayesian extreme quantile regression and the classical approach, we were interested in finding out how those error terms' distributions affect our estimation, for both methods and for both extreme quantiles. For these simulated data sets Bayesian extreme quantile regression was performed via a MCMC algorithm, which uses independent improper uniform priors and a Metropolis-Hastings sampling step for all parameters.

After that, we used one real data set, which was considered by Garcia and Perron (1996) and consists of the US ex-post real interest rates. In order to perform Bayesian extreme quantile regression we used, again, a MCMC algorithm which assumes independent improper uniform prior distributions. The classical quantile regression was performed by using linear programming. We assumed a linear, a quadratic and a cubic model to fit the extreme quantiles of the data. We found very interesting to use our methods in order to model non-extreme quantiles as well and check possible similarities, or

differences, in our approaches. From the results we got evidence that a possible combination of linear programming and MCMC methods would lead to better results. As a consequence, we managed to create an algorithm, which combines linear programming and MCMC methods, in order to perform a more accurate Bayesian extreme quantile regression, in terms of parameter estimation and confidence intervals estimation.

Then, our aim was to analyze some financial data sets using Bayesian extreme quantile regression and hidden Markov models. Specifically, we wanted to estimate the number of the underlying hidden states and possible structural changes (break points) of our data sets, for both extreme quantiles. The data we used were the US ex-post real interest rates and the US treasury bill real interest rates. For the analysis of these data sets we used discrete-time m-state hidden Markov models and break-point hidden Markov models. Two different kinds of hidden Markov models were used. Those which associate the hidden state with a Normal distribution (Normal HMMs) and those which associate the hidden state with an asymmetric Laplace distribution (ALD HMMs). Bayesian extreme quantile regression was performed via MCMC algorithms, using Gibbs sampling (in the case of the Normal HMM) and a mixture of Gibbs and Metropolis-Hastings sampling (in the case of the ALD HMM). In order to estimate the number of the hidden states and the number of break-points, we considered a problem of model choice, which was performed based on the DIC value of every model.

7.1 Simulated Data

Our aim, first, was to see how the distribution of the error term affects the parameter estimation via a Bayesian extreme quantile regression approach and, second, to compare this approach with the classical quantile regression method. We found that we have a good parameter estimation under the Bayesian extreme quantile regression approach, for both quantiles, no matter what the error term's distribution is. However, the parameter estimation provided by the classical extreme quantile regression method was slightly better. Additionally, the classical method works very well for a small number of simulated data as well, but the Bayesian extreme quantile regression provides better results as the number of the simulated data gets larger. However, the confidence intervals for the parameters obtained by Bayesian extreme quantile regression were better than those obtained by the classical approach.

7.2 Real Data Set

Our aim was to model the extreme quantiles of the series via a Bayesian extreme quantile regression and the classical quantile regression approach and then compare these methods. We assumed a linear, a quadratic and a cubic model to fit those quantiles. In order to estimate the parameters of those models we used Bayesian extreme quantile regression (via an MCMC algorithm) and the classical extreme quantile regression (via linear programming). We found that the linear model fails to model the shape of the data and that the cubic model is better than the quadratic one, therefore it provides the best possible fit for the extreme quantiles. Concerning the parameter estimation, the classical approach is slightly better, but Bayesian extreme quantile regression provides better confidence intervals for the estimated parameters. However, the parameter estimation, and as a consequence the quantile fitting, is the same for both methods, when we deal with non-extreme quantiles. Finally, a new algorithm, which combines MCMC methods and linear programming, was used in order to perform Bayesian extreme quantile regression and it was found to provide a very good parameter estimation and very good confidence intervals for those parameters.

7.3 Real Data Sets and Hidden Markov Models

Our aim was to explore the underlying hidden states of the highest and lowest extreme quantiles of our financial data sets, by using two different HMMs (Normal HMM and ALD HMM). Additionally, we wanted to check if and how the two different HMMs we assumed affect the extreme quantile modeling. Then, we applied two different break-point HMMs to the first real data set, in order to check the existence of any structural changes in our financial series.

7.3.1 US ex-post Real Interest Rates

Using a Normal HMM which assumes a quadratic fit for both extreme quantiles of the series, we found that the highest extreme quantile can be modeled by 4 hidden states and the lowest extreme quantile can be modeled by 3 hidden states. If we use an ALD HMM we need 3 hidden states for both extreme quantiles. However, the best fit is provided by a Normal HMM which assumes a cubic fit for both extreme quantiles. This model uses 2 hidden states for the highest extreme quantile and 3 hidden states for the lowest extreme quantile.

Then, we applied a break-point Normal HMM, which assumes a cubic fit for both extreme quantiles and a break-point ALD HMM. Our results showed that both HMMs estimate 1 break-point for the highest extreme quantile and 2 break-points for the lowest extreme quantile. Additionally, they provided similar dates for those break-points.

Finally, we used a continuous state space HMM, which assumes a cubic fit for both extreme quantiles and we saw that we had a very good extreme quantile fit.

7.3.2 US Treasury Bill Real Interest Rates

Using an ALD HMM we found that both extreme quantiles of the series can be modeled by 4 hidden states. A slightly better fit was obtained by a Normal HMM, which assumes a quadratic fit for both extreme quantiles. This model uses 3 hidden states for both extreme quantiles. However, an even better fit for the highest extreme quantile of the series was obtained by another Normal HMM, which assumes two quadratic model fits for that quantile and estimates 5 hidden states. On the other hand, a cubic model fit for the highest extreme quantile was found to be a very bad choice.

After that, we modeled both extreme quantiles of the series by using continuous state space HMMs. The lowest extreme quantile was modeled very well by a continuous state space HMM, which assumes a quadratic fit of the extreme quantile. However, the highest extreme quantile was modeled very well by a continuous state space HMM, which assumes two quadratic model fits of the extreme quantile.

7.4 General Comments

We managed to perform Bayesian extreme quantile regression using Normal HMMs and ALD HMMs. Both methods provided us with very good estimations. However, in some cases they slightly differ on the number of hidden states. Using ALD HMMs we have a more straight-forward estimation as we can define the quantile of interest, where as by using Normal HMMs we need to perform a quantile fit first and simulate new data, which correspond to the quantiles of interest. This is why we cannot compare those methods using the DIC criterion. The first method computes the likelihood using the initial data and the second one uses new simulated data. However, we have evidence, from various graphs and plots, that a Normal HMM tends to better model the shape of the series, than a ALD HMM.

Moreover, if someone is interested in estimating the possible break-points, rather than the hidden states, it is more appropriate to use break-point HMMs. In that case we found that both Normal and ALD HMMs provide us with very good and similar results.

Finally, the fact that we used both discrete and continuous state-space HMMs enable us, based on our experience, to choose which one to use in order to have the most appropriate construction of our model and the most appropriate assumption for the hidden states of the model.

7.5 Further Research

We can extend our research by employing recent advance MCMC methods, as in Yu *et al*. (2011). It would be possible to use a Mixture of Dirichlet Process (MDP) model for the likelihood and a computationally efficient data augmentation scheme to aid inference. Like Yu *et al*. (2011), we can perform Bayesian semi-parametric time-series analysis and use MCMC methods, which combine recent retrospective sampling techniques with the use of slice sampler variables.

We can also extend our research by developing HMMs for Bayesian spatial quantile regression. Following Reich *et al*. (2010), we can perform a Bayesian spatial quantile regression method using a non-Gaussian response. This allows for complicated relationships between the response and the covariates. Additionally, by modeling the conditional distribution as a spatial process, our model will account for spatial variability.

Moreover, following Hughes *et al*. (1999), we can use non-homogeneous HMMs for precipitation occurences. We can also investigate whether a possible combination of the classical extreme quantile regression (for parameter estimation) and bootstrapping methods (for confidence intervals estimation) could be an alternative method for extreme quantile regression modeling that can be compared to our proposed methodology.

Appendix A

Linear Programming (LP)

Linear programming, known also as linear optimization, is a specific case of mathematical programming (mathematical optimization). It is a method for determining a way to obtain the best outcome (such as maximum profit or minimum cost), in a given mathematical model, under some conditions represented as linear relationships.

It deals with problems, which can be expressed in the following form (canonical form):

$$\text{maximize} \quad \mathbf{c}^T\mathbf{x}$$

$$\text{subject to} \quad A\mathbf{x} \leq \mathbf{b}, \quad \mathbf{x} \geq \mathbf{0}$$

where $\mathbf{x}$ represents the vector of variables to be determined, $\mathbf{c}$ and $\mathbf{b}$ are vectors of known coefficients. They are also called decision variables and their values are unknown in the beginning of the problem. They usually represent things that can be controlled or adjusted. $\mathbf{c}^T\mathbf{x}$ is called the objective function and it is the expression to be maximized. The equations $A\mathbf{x} \leq \mathbf{b}$ are the constraints, which specify the area over which the objective function is to be optimized, by combining the variables to express limits on the possible solutions. The objective function is defined on its feasible region, which is a convex polyhedron. This convex polyhedron is a set defined as the intersection of the spaces, which are created by the linear inequalities.

The general process for solving linear programming exercises is to form the feasible region by graphing the inequalities (constraints). Then, we can get the coordinates of the corners of our feasible region and check which gives the optimal (highest or lowest) value through our objective function. However, when there are many variables, graphing the inequalities is impossible due to the fact that we need to work in high-dimensional regions. This is why there are various algorithms, which enable us to tackle this problem, such as simplex method (Wood and Dantzig, 1949) and polynomial time algorithm (Khachian, 1979; Karmarkar, 1984).

Notes on the assumptions of LP

The usage of linear relationships and linear models in order to describe real applications has

certain limitations. The most obvious one is that those applications can be poorly modeled by lines. This problem is addressed by using nonlinear relationships like curves or step-functions. Therefore, a technique, different than LP, should be used.

The usage of cross-product terms, such as the product x_1x_2, where x_1 and x_2 are two different variables, is not permitted in LP.

LP assumes that the variables are real valued (they can take on fractional values). Sometimes, though, fractional values are not appropriate, such as determining the number of people that have to work in a company. In that case the variable that describes this number must take on an integer value. Therefore, the usage of integer programming is needed. Treating those kind of problems as linear problems and then round the results to the closest integer is not acceptable, as it may lead to very bad results. This happens because the optimal solution in integer programming can be very different from the approximate solution (round the optimal solution of LP to the closest integer) obtained by LP.

All mathematical programming (and therefore LP) have a common weakness, which is the assumption of the accuracy of the input data. This is the assumption that the objective function coefficients and the constraint coefficients are all correct. In fact, these values are seldom known with accuracy in real world. Therefore, companies use data to estimate those values. For example, they use the average price paid for the materials used, the average worker wages and the average selling prices, in order to estimate the profit per product sold.

Sometimes we do not know how useful the optimal solution is, especially when the input data is of poor quality, because quite different result can be obtained by a slight change in the input parameters. That means that if our estimation of the input parameters is poor, the optimal solution is not the best for our real life problem. Therefore, sensitivity analysis is applied, which explores how sensitive the optimal solution is to slight changes in the values of the input parameters, by using various tests.

Properties of Extreme Quantile Regression

Following Portnoy and Jureckova (1999), we consider the model (1) of the previous section and we assume that the error term follows a Weibull-like distribution. Let $\hat{\beta} = \hat{\beta}(1)$ be the maximal extreme regression quantile. Then, the joint density for that extreme quantile is

$$f_{\hat{\beta}(1)} = \sum_{h \in \boldsymbol{H}} d(h, \bar{\mathbf{x}}) \prod_{i \in h} f(\mathbf{x}_i'(\mathbf{b} - \beta)) \prod_{i \notin h} F(\mathbf{x}_i'(\mathbf{b} - \beta)),$$

where H is the set of all p-element subsets of the indices $\{1, 2, ..., n\}$ and $d(h, \bar{\mathbf{x}}) \equiv det(\ ((\mathbf{x}_i))\ :\ i \in h)\ I(u \in co(h))$, where $co(h)$ denotes the convex hull of vectors $\{\mathbf{x}_i\ :\ i \in h\}$.

Following Smith (1994), and based on the same procedures for the minimal extreme regression quantile $\hat{\beta} = \hat{\beta}(0)$, we can obtain its joint density from

$$f_{\hat{\beta}(0)} = \sum_{h \in \boldsymbol{H}} d(h, \bar{\mathbf{x}}) \prod_{i \in h} f(\mathbf{x}_i'(\mathbf{b} - \beta)) \prod_{i \notin h} \left[1 - F(\mathbf{x}_i'(\mathbf{b} - \beta))\right],$$

For the initial model (1), the distribution of the quantity

$$\mathbf{V}_n = a(logn)^{\frac{a-1}{a}} \left[(\hat{\beta} - \beta) - (logn)^{1/a}\mathbf{e}\right],$$

where $\mathbf{e} = (1, 0, ..., 0)$, converges to the density function

$$f^*(\mathbf{v}) = (p!)^{-1} g_2(\mathbf{v}; a) e^{-g_1(\mathbf{v})},$$

as $n \to \infty$. We can calculate $g_1(\mathbf{v})$ and $g_2(\mathbf{v}; a)$ from

$$g_1(\mathbf{v}) = E[e^{-\mathbf{x}_i'\mathbf{v}}]$$

or

$$\frac{1}{n} \sum_{i=1}^{n} e^{-\mathbf{x}_i'\mathbf{v}} \to g_1(\mathbf{v})$$

and

$$g_2(\mathbf{v}; a) = E[d(\{1, ..., p\})] \prod_{i=1}^{p} I(\mathbf{x}_i'\mathbf{v} \geq 0) exp\left\{-\sum_{i=1}^{p} \mathbf{x}_i'\mathbf{v}\right\}$$

or

$$\binom{n}{p}^{-1} \sum_{h \in H} d(h, \bar{\mathbf{x}}) \left\{\prod_{i \in h} I(\mathbf{x}_i'\mathbf{v} \geq \epsilon_n) \left(1 + \frac{(a-1)\mathbf{x}_i'\mathbf{v}}{a\ logn} + \epsilon_n\right)\ exp\left\{-\sum_{i \in h} \mathbf{x}_i'\mathbf{v}\right\}\right\} \to g_2(\mathbf{v}; a).$$

Then, we can obtain $F^{*-1}(a/2)$ and $F^{*-1}(1 - a/2)$ from

$$\int_{-\infty}^{F^{*-1}(a/2)} f^*(\mathbf{v})\ d\mathbf{v} = a/2$$

and

$$\int_{-\infty}^{F^{*-1}(1-a/2)} f^*(\mathbf{v})\ d\mathbf{v} = 1 - a/2.$$

We know that

$$1 - a = Pr\left[F^{*-1}(a/2) \leq \mathbf{V}_n \leq F^{*-1}(1 - a/2)\right],$$

where $F^{*-1}(p)$, $0 \leq p \leq 1$, is the inverse cumulative distribution function of $f^*(\mathbf{v})$. So,

$$1 - a =$$

$$Pr\left[F^{*-1}(a/2) \leq a(logn)^{\frac{a-1}{a}}\left[(\hat{\beta} - \beta) - (logn)^{1/a}\mathbf{e}\right] \leq F^{*-1}(1-a/2)\right] =$$

$$Pr\left[\hat{\beta} - F^{*-1}(1-a/2)\,\frac{(logn)^{-(\frac{a-1}{a})}}{a} - (logn)^{1/a}\,\mathbf{e} \leq \beta \leq \hat{\beta} - F^{*-1}(a/2)\,\frac{(logn)^{-(\frac{a-1}{a})}}{a} - (logn)^{1/a}\,\mathbf{e}\right].$$

Using the last equation we can approximately obtain the confidence intervals for the parameters β.

Calculations of the true values of the parameters, for all distributions of the error term.

For Uniform distribution : $F^{-1}(\tau; a, b) = (b-a)\tau + a$.

So, for Uniform(0,1) we have :

$$\beta_0(0) = 1 + 0 = 1$$

$$\beta_0(1) = 1 + 1 = 2.$$

For Beta distribution : $F^{-1}(\tau; a, b) = \tau^{1/a},\ b > 0$.

So, for Beta(1,1) we have :

$$\beta_0(0) = 1 + 0 = 1$$

$$\beta_0(1) = 1 + 1 = 2.$$

For Weibull distribution : $F^{-1}(\tau; \lambda, \kappa) = \lambda \sqrt[\kappa]{ln\left(\frac{1}{1-\tau}\right)}$.

So, for Weibull(2,1) we have :

$$\beta_0(0) = 1 + \sqrt{ln(1)} = 1$$

$$\beta_0(1) = 1 + \sqrt{ln(1/0)} = +\infty.$$

Metropolis-Hastings Algorithm (for real data set and quadratic model fit)

1. Start the chain at some value $\beta^{(0)} = \left(\beta_0^{(0)}, \beta_1^{(0)}, \beta_2^{(0)}\right)$.
2. Given that the chain is currently at $\beta^{(j)} = \left(\beta_0^{(j)}, \beta_1^{(j)}, \beta_2^{(j)}\right)$:

Propose a candidate value $\beta_0^{can} \sim N\left(\beta_0^{(j)}, v_0\right)$, for some suitably chosen variance v_0. Take as new value of the chain

$$\beta_0^{(j+1)} = \begin{cases} \beta_0^{can}, & \text{with probability p} \\ \beta_0^{(j)}, & \text{with probability 1-p} \end{cases}$$

where

$$p = min\left\{1, \frac{exp\left\{-\sum_{i=1}^{n}\left(y_i - \beta_0^{can} - \beta_1^{(j)}x_{1i} - \beta_2^{(j)}x_{2i}\right) I\left(y_i - \beta_0^{can} - \beta_1^{(j)}x_{1i} - \beta_2^{(j)}x_{2i} < 0\right)\right\}}{exp\left\{-\sum_{i=1}^{n}\left(y_i - \beta_0^{(j)} - \beta_1^{(j)}x_{1i} - \beta_2^{(j)}x_{2i}\right) I\left(y_i - \beta_0^{(j)} - \beta_1^{(j)}x_{1i} - \beta_2^{(j)}x_{2i} < 0\right)\right\}}\right\}.$$

(This is implemented by drawing $q \sim Uniform(0,1)$ and taking $\beta_0^{(j+1)} = \beta_0^{can}$ if and only if $q < p$).

Propose a candidate value $\beta_1^{can} \sim N\left(\beta_1^{(j)}, v_1\right)$, for some suitably chosen variance v_1. Take as new value of the chain

$$\beta_1^{(j+1)} = \begin{cases} \beta_1^{can}, & \text{with probability p} \\ \beta_1^{(j)}, & \text{with probability 1-p} \end{cases}$$

where

$$p = min\left\{1, \frac{exp\left\{-\sum_{i=1}^{n}\left(y_i - \beta_0^{(j+1)} - \beta_1^{can}x_{1i} - \beta_2^{(j)}x_{2i}\right) I\left(y_i - \beta_0^{(j+1)} - \beta_1^{can}x_{1i} - \beta_2^{(j)}x_{2i} < 0\right)\right\}}{exp\left\{-\sum_{i=1}^{n}\left(y_i - \beta_0^{(j+1)} - \beta_1^{(j)}x_{1i} - \beta_2^{(j)}x_{2i}\right) I\left(y_i - \beta_0^{(j+1)} - \beta_1^{(j)}x_{1i} - \beta_2^{(j)}x_{2i} < 0\right)\right\}}\right\}.$$

(This is implemented by drawing $q \sim Uniform\,(0,1)$ and taking $\beta_1^{(j+1)} = \beta_1^{can}$ if and only if $q < p$).

Propose a candidate value $\beta_2^{can} \sim N\left(\beta_2^{(j)}, v_2\right)$, for some suitably chosen variance v_2. Take as new value of the chain

$$\beta_2^{(j+1)} = \begin{cases} \beta_2^{can}, & \text{with probability p} \\ \beta_2^{(j)}, & \text{with probability 1-p} \end{cases}$$

where

$$p = min\left\{1, \frac{exp\left\{-\sum_{i=1}^{n}\left(y_i - \beta_0^{(j+1)} - \beta_1^{(j+1)}x_{1i} - \beta_2^{can}x_{2i}\right) I\left(y_i - \beta_0^{(j+1)} - \beta_1^{(j+1)}x_{1i} - \beta_2^{can}x_{2i} < 0\right)\right\}}{exp\left\{-\sum_{i=1}^{n}\left(y_i - \beta_0^{(j+1)} - \beta_1^{(j+1)}x_{1i} - \beta_2^{(j)}x_{2i}\right) I\left(y_i - \beta_0^{(j+1)} - \beta_1^{(j+1)}x_{1i} - \beta_2^{(j)}x_{2i} < 0\right)\right\}}\right\}.$$

(This is implemented by drawing $q \sim Uniform(0,1)$ and taking $\beta_2^{(j+1)} = \beta_2^{can}$ if and only if $q < p$).

3. Iterate step 2 a large number of times. Discard an initial number of draws and base inference on subsequent draws.

Note that: a) for the cubic model fit we follow the same procedure, by adding one extra parameter β_3; b) for the simulated data set we follow the same procedure, however, in step 2 we have $\beta^{(j)} = (\beta_0^{(j)}, \beta_1^{(j)})$.

Calculations of the MCMC algorithm, for simulating the mean of the new data set.

Our model is

$$y_i = \beta_0 + \beta_1\, x_i + \beta_2\, x_i^2 + \beta_3\, x_i^3, \;\; i = 1, 2, ..., n,$$

with $\varepsilon_i \sim N(0, 1)$. Therefore, we have

$$f(\varepsilon_i) = f(y_i - \beta_0 - \beta_1\, x_i - \beta_2\, x_i^2 - \beta_3\, x_i^3)$$

and

$$f(\boldsymbol{\varepsilon}) = L(\mathbf{y}|\boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}} exp\left\{-\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1\, x_i - \beta_2\, x_i^2 - \beta_3\, x_i^3)^2\right\}.$$

We can obtain the posterior distributions of the parameters from

$$\pi(\boldsymbol{\beta}|\mathbf{y}) \propto L(\mathbf{y}|\boldsymbol{\beta})\pi(\boldsymbol{\beta}).$$

For β_1 we have

$$\pi(\beta_1|\mathbf{y}) \propto L(\mathbf{y}|\boldsymbol{\beta})\pi(\beta_1).$$

$$\propto exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\left[(\beta_1 x_i)^2 - 2(y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)(\beta_1 x_i) + (y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)^2\right]\right\}$$

$$= exp\left\{-\frac{1}{2}\left[\beta_1^2\sum_{i=1}^{n} x_i^2 - 2\beta_1\sum_{i=1}^{n}\left[x_i(y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)\right] + \sum_{i=1}^{n}(y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)^2\right]\right\}$$

$$\propto exp\left\{-\frac{\sum_{i=1}^{n} x_i^2}{2}\left[\beta_1^2 - \frac{-2\beta_1\sum_{i=1}^{n}\left[x_i(y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)\right]}{\sum_{i=1}^{n} x_i^2}\right]\right\}$$

$$\propto exp\left\{-\frac{\sum_{i=1}^{n} x_{1i}^2}{2}\left[\beta_1 - \frac{\sum_{i=1}^{n}\left[x_i(y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)\right]}{\sum_{i=1}^{n} x_i^2}\right]^2\right\}$$

Therefore, we get

$$\beta_1 \sim N\left(\frac{\sum_{i=1}^n \left[x_i(y_i - \beta_0 - \beta_2 x_i^2 - \beta_3 x_i^3)\right]}{\sum_{i=1}^n x_i^2}, \frac{1}{\sum_{i=1}^n x_i^2}\right).$$

In a similar way, we get

$$\beta_2 \sim N\left(\frac{\sum_{i=1}^n \left[x_i^2(y_i - \beta_0 - \beta_1 x_i - \beta_3 x_i^3)\right]}{\sum_{i=1}^n x_i^4}, \frac{1}{\sum_{i=1}^n x_i^4}\right),$$

$$\beta_3 \sim N\left(\frac{\sum_{i=1}^n \left[x_i^3(y_i - \beta_0 - \beta_1 x_i - \beta_2 x_i^2)\right]}{\sum_{i=1}^n x_i^6}, \frac{1}{\sum_{i=1}^n x_i^6}\right)$$

and

$$\beta_0 \sim N\left(\frac{\sum_{i=1}^n \left[(y_i - \beta_1 x_i - \beta_2 x_i^2 - \beta_3 x_i^3)\right]}{n}, \frac{1}{n}\right).$$

Appendix B

A. Calculation of the Full Conditional Posterior Distributions

Calculations for Normal HMM

Priors

$$\pi(P) = \prod_{i=1}^{m} \pi(p_{i.}) \propto \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1}$$

$$\pi(\mu) = \prod_{i=1}^{m} \pi(\mu_i) \propto \prod_{i=1}^{m} exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\} = exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}$$

$$\pi(\kappa) = \prod_{i=1}^{m} \pi(\kappa_i) \propto \prod_{i=1}^{m} \kappa_i^{a-1} exp\left\{-b\kappa_i\right\}$$

Likelihood

$$L(\mathbf{y}^T|\mathbf{x}^T, \mu, \sigma^2, P) = \pi_{x_1} f_{x_1}(y_1) p_{x_1,x_2} f_{x_2}(y_2) ... p_{x_{T-1},x_T} f_{x_T}(y_T)$$

$$= \pi(x_1) p_{x_1,x_2} ... p_{x_{T-1},x_T} f_{x_1}(y_1) ... f_{x_T}(y_T)$$

$$= \pi(x_1) \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} f_i(y_t)$$

$$= \pi(x_1) \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} \sqrt{\frac{\kappa_i}{2\pi}} exp\left\{-\frac{\kappa_i}{2}(y_t - \mu_i)^2\right\}$$

Joint posterior

$$f(\mu, \kappa, P|\mathbf{y}^T, \mathbf{x}^T) \propto \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} \sqrt{\kappa_i} exp\left\{-\frac{\kappa_i}{2}(y_t - \mu_i)^2\right\}$$

$$\times \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\} \times \prod_{i=1}^{m} \kappa_i^{a-1} exp\left\{-b\kappa_i\right\}$$

Calculation of full conditionals

$$\pi(p_{i.}|\mathbf{y}^T, \mathbf{x}^T\, \mu, \kappa) \propto \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} \propto \prod_{j=1}^{m} p_{ij}^{n_{ij} + \omega_j - 1}$$

$$\equiv Dir(n_{i.} + \omega)$$

$$\pi(\kappa_i|\mathbf{y}^T, \mathbf{x}^T\, \mu, P) \propto \kappa_i^{a-1} exp\left\{-b\kappa_i\right\} \prod_{t:x_t=i} \sqrt{\kappa_i} exp\left\{-\frac{\kappa_i}{2}(y_t - \mu_i)^2\right\}$$

$$= \kappa_i^{a-1} exp\left\{-b\kappa_i\right\} \kappa_i^{n_i/2} exp\left\{-\frac{\kappa_i}{2}\sum_{t:x_t=i}(y_t - \mu_i)^2\right\}$$

$$= \kappa_i^{a+n_i/2-1} exp\left\{-\kappa_i\left(b + \sum_{t:x_t=i}\frac{(y_t-\mu_i)^2}{2}\right)\right\}$$

$$\equiv Gamma\left(a + \frac{n_i}{2}, b + \sum_{t:x_t=i}\frac{(y_t-\mu_i)^2}{2}\right)$$

$$\pi(\mu_i|\mathbf{y}^T, \mathbf{x}^T\,\kappa, P) \propto \prod_{t:x_t=i} exp\left\{-\frac{\kappa_i}{2}(y_t-\mu_i)^2\right\} exp\left\{-\frac{\lambda}{2}(\mu_i-\xi)^2\right\}$$

$$= exp\left\{-\frac{\kappa_i}{2}\sum_{t:x_t=i}(y_t-\mu_i)^2\right\} exp\left\{-\frac{\lambda}{2}(\mu_i-\xi)^2\right\}$$

$$= exp\left\{-\frac{\kappa_i}{2}\sum_{t:x_t=i}(y_t-\mu_i)^2 - \frac{\lambda}{2}(\mu_i-\xi)^2\right\}$$

$$= exp\left\{-\frac{\kappa_i}{2}\sum_{t:x_t=i}\left(y_t^2 - 2y_t\mu_i + \mu_i^2\right) - \frac{\lambda}{2}\left(\mu_i^2 - 2\xi\mu_i + \xi^2\right)\right\}$$

$$= exp\left\{-\frac{\kappa_i}{2}\left(\sum_{t:x_t=i}y_t^2 - 2\mu_i\sum_{t:x_t=i}y_t + n_i\mu_i^2\right) - \frac{\lambda}{2}\left(\mu_i^2 - 2\xi\mu_i + \xi^2\right)\right\}$$

$$= exp\left\{-\frac{1}{2}\left((\kappa_i n_i + \lambda)\mu_i^2 - 2\left(\kappa_i\sum_{t:x_t=i}y_t + \lambda\xi\right)\mu_i + \left(\kappa_i\sum_{t:x_t=i}y_t^2 + \lambda\xi\right)\right)\right\}$$

$$\propto exp\left\{-\frac{\kappa_i n_i + \lambda}{2}\left[\mu_i^2 - 2\frac{\kappa_i\sum_{t:x_t=i}y_t + \lambda\xi}{\kappa_i n_i + \lambda}\mu_i + \left(\frac{\kappa_i\sum_{t:x_t=i}y_t + \lambda\xi}{\kappa_i n_i + \lambda}\right)^2\right]\right\}$$

$$= exp\left\{-\frac{\kappa_i n_i + \lambda}{2}\left(\mu_i - \frac{\kappa_i\sum_{t:x_t=i}y_t + \lambda\xi}{\kappa_i n_i + \lambda}\right)^2\right\}$$

$$\equiv Normal\left(\frac{\kappa_i\sum_{t:x_t=i}y_t + \lambda\xi}{n_i\kappa_i + \lambda}, \frac{1}{n_i\kappa_i + \lambda}\right)$$

Calculation for Normal Break-point HMM

Priors

$$\pi(P) = \prod_{i=1}^{m}\pi(p_{ii}) \propto \prod_{i=1}^{m} p_{ii}^{p-1}(1-p_{ii})^{q-1}$$

$$\pi(\mu) = \prod_{i=1}^{m}\pi(\mu_i) \propto \prod_{i=1}^{m} exp\left\{-\frac{\lambda}{2}(\mu_i-\xi)^2\right\} = exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i-\xi)^2\right\}$$

$$\pi(\kappa) = \prod_{i=1}^{m}\pi(\kappa_i) \propto \prod_{i=1}^{m}\kappa_i^{a-1}exp\left\{-b\kappa_i\right\}$$

Likelihood

$$L(\mathbf{y}^T|\mathbf{x}^T,\mu,\sigma^2,P) = \pi(x_1)\times f_{x_1}(y_1)p_{x_1,x_2}f_{x_2}(y_2)...p_{x_{T-1},x_T}f_{x_T}(y_T)$$

$$= 1\times p_{x_1,x_2}...p_{x_{T-1},x_T}f_{x_1}(y_1)...f_{x_T}(y_T)$$

$$= \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii})\prod_{i=1}^{m}\prod_{t:x_t=i} f_i(y_t)$$

$$= \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii})\prod_{i=1}^{m}\prod_{t:x_t=i}\sqrt{\frac{\kappa_i}{2\pi}}exp\left\{-\frac{\kappa_i}{2}(y_t-\mu_i)^2\right\}$$

Joint posterior

$$f(\mu,\kappa,P|\mathbf{y}^T,\mathbf{x}^T) \propto \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii})\prod_{i=1}^{m}\prod_{t:x_t=i}\sqrt{\kappa_i}exp\left\{-\frac{\kappa_i}{2}(y_t-\mu_i)^2\right\}$$

$$\times\prod_{i=1}^{m} p_{ii}^{p-1}(1-p_{ii})^{q-1}exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i-\xi)^2\right\}\times\prod_{i=1}^{m}\kappa_i^{a-1}exp\left\{-b\kappa_i\right\}$$

Calculation of conditionals

$$\pi(p_{ii}|\mathbf{y}^T,\mathbf{x}^T\,\mu,\kappa) \propto p_{ii}^{n_{ii}}(1-p_{ii})p_{ii}^{p-1}(1-p_{ii})^{q-1}$$

$$= p_{ii}^{p+n_{ii}-1}(1-p_{ii})^{q}$$

$$\equiv Beta(p+n_{ii},q+1)$$

(using the same calculations as in the Normal HMM we obtain the next two conditionals)

$$\pi(\mu_i|\mathbf{y}^T,\mathbf{x}^T\,\kappa,P) \equiv Normal\left(\frac{\kappa_i\sum_{t:x_t=i}y_t+\lambda\xi}{n_i\kappa_i+\lambda},\frac{1}{n_i\kappa_i+\lambda}\right)$$

$$\pi(\kappa_i|\mathbf{y}^T,\mathbf{x}^T\,\mu,P) \equiv Gamma\left(a+\frac{n_i}{2},b+\sum_{t:x_t=i}\frac{(y_t-\mu_i)^2}{2}\right)$$

Dirichlet distribution

The Dirichlet distribution (after Johann Peter Gustav Lejeune Dirichlet) is a family of continuous multivariate probability distributions. It is parameterized by the vector **w** of nonnegative reals and denoted as Dir(**w**). It is the multivariate generalization of the Beta distribution and conjugate prior of the parameters of the multinomial distribution.

The probability density of the Dirichlet distribution for variables $\mathbf{p} = (p_1, p_2, ..., p_m)$, with parameters $\mathbf{w} = (w_1, w_2, ..., w_m)$ is defined by

$$\pi(\mathbf{p}) = Dir(\mathbf{p}; \mathbf{w}) = \frac{1}{B(\mathbf{w})} \prod_{i=1}^{m} p_i^{w_i - 1},$$

where $w_i > 0$, $p_i \geq 0$, $i = 1, 2, ..., m$ and $\sum_{i=1}^{m} p_i = 1$. The normalizing constant $B(\mathbf{w})$ is the multinomial Beta function, which is expressed in terms of the Gamma function

$$B(\mathbf{w}) = \frac{\prod_{i=1}^{m} \Gamma(w_i)}{\Gamma\left(\sum_{i=1}^{m} w_i\right)}.$$

The mean and the variance of the Dirichlet distribution are

$$E(p_i) = \frac{w_i}{w_0},$$

$$Var(p_i) = \frac{w_i(w_0 - w_i)}{w_0^2(w_0 + 1)},$$

where $w_0 = \sum_{i=1}^{m} w_i$.

B. MCMC algorithms

(Gibbs Sampling with Data Augmentation)

Normal HMM

1. Initialize with $\theta^{(0)} = \left(\mu^{(0)}, \kappa^{(0)}, P^{(0)}\right)$ from their priors.
2. Augment the data by simulating latent variables $\mathbf{x}^{(1)}$, using Forward-Backward algorithm, given $\theta^{(0)}$.
3. For $i = 1, ..., m$, simulate $\mu^{(1)}$ from its full conditional posterior distribution $\pi\left(\mu_i | \kappa^{(0)}, P^{(0)}, \mathbf{x}^{(0)}\right)$.
4. For $i = 1, ..., m$, simulate $\kappa^{(1)}$ from its full conditional posterior distribution $\pi\left(\kappa_i | \mu^{(1)}, P^{(0)}, \mathbf{x}^{(0)}\right)$.
5. For $i = 1, ..., m$, simulate the ith row of the transition matrix $p_{i\cdot}^{(1)}$ from its full conditional posterior distribution $\pi\left(p_{i\cdot} | \mu^{(1)}, \kappa^{(1)}, \mathbf{x}^{(0)}\right)$.
6. Iterate this procedure.

In order to avoid label switching problems, when implementing the above MCMC algorithm, we have labeled the hidden states using the constraint $\mu_1 < \mu_2 < ... < \mu_m$.

Normal Break-point HMM

1. Initialize with $\theta^{(0)} = \left(\mu^{(0)}, \kappa^{(0)}, P^{(0)}\right)$ from their priors.
2. Augment the data by simulating latent variables $\mathbf{x}^{(0)}$, using Forward-Backward algorithm, given $\theta^{(0)}$.
3. For $i = 1, ..., m$, simulate $\mu^{(1)}$ from its full conditional posterior distribution $\pi\left(\mu_i|\kappa^{(0)}, P^{(0)}, \mathbf{x}^{(0)}\right)$.
4. For $i = 1, ..., m$, simulate $\kappa^{(1)}$ from its full conditional posterior distribution $\pi\left(\kappa_i|\mu^{(1)}, P^{(0)}, \mathbf{x}^{(0)}\right)$.
5. For $i = 1, ..., m$, simulate the element of the transition matrix $p_{ii}^{(1)}$ from its full conditional posterior distribution $\pi\left(p_{ii}|\mu^{(1)}, \kappa^{(1)}, \mathbf{x}^{(0)}\right)$.
6. Check for break-points based on $\mathbf{x}^{(0)}$.
7. Iterate this procedure.

Note that the simulated times of the breaks can be obtained from the draws of $\mathbf{x}$.

Appendix C

A. Calculation of the Full Conditional Posterior Distributions

Calculations for ALD HMM

Priors

$$\pi(P) = \prod_{i=1}^{m} \pi(p_{i.}) \propto \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1}$$

$$\pi(\mu) = \prod_{i=1}^{m} \pi(\mu_i) \propto \prod_{i=1}^{m} exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\} = exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}$$

Likelihood

$$L(\mathbf{y}^T|\mathbf{x}^T, \mu, P) = \pi_{x_1} f_{x_1}(y_1) p_{x_1,x_2} f_{x_2}(y_2) ... p_{x_{T-1},x_T} f_{x_T}(y_T)$$

$$= \pi(x_1) p_{x_1,x_2} ... p_{x_{T-1},x_T} f_{x_1}(y_1) ... f_{x_T}(y_T)$$

$$= \pi(x_1) \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} f_i(y_t)$$

$$= \pi(x_1) \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} exp\left\{-p_\tau(y_t - \mu_i)\right\}$$

Joint posterior

$$f\left(\mu, P|\mathbf{y}^T, \mathbf{x}^T\right) \propto \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{i=1}^{m} \prod_{t:x_t=i} exp\{-p_q(y_t - \mu_i)\}$$

$$\times \prod_{i=1}^{m} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}$$

Calculation of full conditionals

$$\pi(p_{i.}|\mathbf{y}^T, \mathbf{x}^T\, \mu) \propto \prod_{j=1}^{m} p_{ij}^{n_{ij}} \prod_{j=1}^{m} p_{ij}^{\omega_j - 1} \propto \prod_{j=1}^{m} p_{ij}^{n_{ij} + \omega_j - 1}$$

$$\equiv Dir(n_{i.} + \omega)$$

$$\pi(\mu_i|\mathbf{y}^T, \mathbf{x}^T, P) \propto \prod_{t:x_t=i} exp\{-p_\tau(y_t - \mu_i)\} exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\}$$

$$\propto exp\left\{\sum_{t:x_t=i} \left[(y_t - \mu_i)(\tau - I_{(-\infty,0)}(y_t - \mu_i))\right] + \frac{\lambda}{2}(\mu_i - \xi)^2\right\}$$

Calculation for ALD Break-point HMM

Priors

$$\pi(P) = \prod_{i=1}^{m} \pi(p_{ii}) \propto \prod_{i=1}^{m} p_{ii}^{p-1}(1-p_{ii})^{q-1}$$

$$\pi(\mu) = \prod_{i=1}^{m} \pi(\mu_i) \propto \prod_{i=1}^{m} exp\left\{-\frac{\lambda}{2}(\mu_i - \xi)^2\right\} = exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}$$

Likelihood

$$L(\mathbf{y}^T|\mathbf{x}^T, \mu, P) = \pi(x_1) \times f_{x_1}(y_1)p_{x_1,x_2}f_{x_2}(y_2)...p_{x_{T-1},x_T}f_{x_T}(y_T)$$

$$= 1 \times p_{x_1,x_2}...p_{x_{T-1},x_T}f_{x_1}(y_1)...f_{x_T}(y_T)$$

$$= \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m} \prod_{t:x_t=i} f_i(y_t)$$

$$= \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m} \prod_{t:x_t=i} exp\left\{p_\tau(y_t - \mu_i)\right\}$$

Joint posterior

$$f(\mu, P|\mathbf{y}^T, \mathbf{x}^T) \propto \prod_{i=1}^{m} p_{ii}^{n_{ii}}(1-p_{ii}) \prod_{i=1}^{m} \prod_{t:x_t=i} exp\left\{p_\tau(y_t - \mu_i)\right\}$$

$$\times \prod_{i=1}^{m} p_{ii}^{p-1}(1-p_{ii})^{q-1} exp\left\{-\frac{\lambda}{2}\sum_{i=1}^{m}(\mu_i - \xi)^2\right\}$$

Calculation of conditionals

$$\pi(p_{ii}|\mathbf{y}^T, \mathbf{x}^T\,\mu) \propto p_{ii}^{n_{ii}}(1-p_{ii})p_{ii}^{p-1}(1-p_{ii})^{q-1}$$

$$= p_{ii}^{p+n_{ii}-1}(1-p_{ii})^q$$

$$\equiv Beta(p + n_{ii}, q + 1)$$

(using the same calculations as in the Normal HMM we obtain the following)

$$\pi(\mu_i|\mathbf{y}^T, \mathbf{x}^T\,P) \propto exp\left\{\sum_{t:x_t=i}\left[(y_t - \mu_i)(\tau - I_{(-\infty,0)}(y_t - \mu_i))\right] + \frac{\lambda}{2}(\mu_i - \xi)^2\right\}.$$

B. MCMC algorithms

(Gibbs and Metropolis-Hastings Sampling with Data Augmentation)

ALD HMM

1. Initialize with $\theta^{(0)} = \left(\mu^{(0)}, P^{(0)}\right)$ from their priors.

2. Given that the chain is currently at $\theta^{(j)} = \left(\mu^{(j)}, P^{(j)}\right)$:

   Augment the data by simulating latent variables $\mathbf{x}^{(j)}$, using Forward-Backward algorithm, given $\theta^{(j)}$.

3. For $i = 1, ..., m$, propose a candidate value $\mu_i^{can} \sim N\left(\mu_i^{(j)}, v_0\right)$, for some suitably chosen variance v_0. Take as new value of the chain

$$\mu_i^{(j+1)} = \begin{cases} \mu_i^{can}, & \text{with probability p} \\ \mu_i^{(j)}, & \text{with probability 1-p} \end{cases}$$

   where

$$p = min\left\{1, \frac{exp\left\{\sum_{t:x_t=i}\left(y_t - \mu_i^{can}\right)\left(q - I(y_t - \mu_i^{can} < 0)\right) + \frac{\tau}{2}(\mu_i^{can} - \xi)^2\right\}}{exp\left\{\sum_{t:x_t=i}\left(y_t - \mu_i^{(j)}\right)\left(q - I\left(y_t - \mu_i^{(j)} < 0\right)\right) + \frac{\tau}{2}(\mu_i^{(j)} - \xi)^2\right\}}\right\}.$$

   (This is implemented by drawing $z \sim Uniform(0, 1)$ and taking $\mu_i^{(j+1)} = \mu_i^{can}$, if and only if $z < p$).

4. For $i = 1, ..., m$, simulate the ith row of the transition matrix $p_{i.}^{(j+1)}$ from its full conditional posterior distribution $\pi\left(p_{i.}^{(j)} | \mu^{(j+1)}, \mathbf{x}^{(j)}\right)$.

5. Iterate this procedure.

In order to avoid label switching problems, when implementing the above MCMC algorithm, we have labeled the hidden states using the constraint $\mu_1 < \mu_2 < ... < \mu_m$.

ALD Break-point HMM

1. Initialize with $\theta^{(0)} = \left(\mu^{(0)}, P^{(0)}\right)$ from their priors.

2. Given that the chain is currently at $\theta^{(j)} = \left(\mu^{(j)}, P^{(j)}\right)$:

   Augment the data by simulating latent variables $\mathbf{x}^{(j)}$, using Forward-Backward algorithm, given $\theta^{(j)}$.

3. For $i = 1, ..., m$, propose a candidate value $\mu_i^{can} \sim N\left(\mu_i^{(j)}, v_0\right)$, for some suitably chosen variance v_0. Take as new value of the chain

$$\mu_i^{(j+1)} = \begin{cases} \mu_i^{can}, & \text{with probability p} \\ \mu_i^{(j)}, & \text{with probability 1-p} \end{cases}$$

where

$$p = min\left\{1, \frac{exp\left\{\sum_{t:x_t=i}(y_t - \mu_i^{can})(q - I(y_t - \mu_i^{can} < 0)) + \frac{\tau}{2}(\mu_i^{can} - \xi)^2\right\}}{exp\left\{\sum_{t:x_t=i}\left(y_t - \mu_i^{(j)}\right)\left(q - I\left(y_t - \mu_i^{(j)} < 0\right)\right) + \frac{\tau}{2}(\mu_i^{(j)} - \xi)^2\right\}}\right\}.$$

(This is implemented by drawing $z \sim Uniform(0, 1)$ and taking $\mu_i^{(j+1)} = \mu_i^{can}$, if and only if $z < p$).

4. For $i = 1, ..., m$, simulate the element of the transition matrix $p_{ii}^{(j+1)}$ from its full conditional posterior distribution $\pi\left(p_{ii}^{(j)} | \mu^{(j+1)}, \mathbf{x}^{(j)}\right)$.

5. Check for break-points based on $\mathbf{x}^{(j)}$.

6. Iterate this procedure.

Note that the times of the breaks can be obtained from the draws of $\mathbf{x}$.

The Metropolis-Hastings Algorithm

This algorithm is a Markov chain Monte Carlo (MCMC) method, for obtaining a sequence of random samples from a probability distribution, which is difficult to sample from. The algorithm was named after Nicholas Metropolis, who first proposed the algorithm for the specific case of the Boltzmann distribution in the paper *Equation of State Calculations by Fast Computing Machines* in 1953, and W. Keith Hastings, who extended it to the more general case in 1970.

Suppose that the current state of our Markov chain is $\theta_1^{(j)}, ..., \theta_d^{(j)}$ and that we want to simulate $\theta_1^{(j+1)}$, which is the next value of θ_1. In other words, we need to update $\theta_1^{(j)}$ to $\theta_1^{(j+1)}$, based on the conditional distribution $\pi(\theta_1|\theta_2^{(j)}, ..., \theta_d^{(j)})$. This is performed as follows:

1. Propose a candidate value θ_1^{can}, which is a draw from an arbitrary distribution with density $q(\theta_1^{can}|\theta_1^{(j)}, \theta_2^{(j)}, ..., \theta_d^{(j)})$.

2. Take as the next value of θ_1 in the chain

$$\theta_1^{(j+1)} = \begin{cases} \theta_1^{can}, & \text{with probability p} \\ \theta_1^{(j)}, & \text{with probability 1-p} \end{cases}$$

where

$$p = min\left\{1, \frac{\pi(\theta_1^{can}|\theta_2^{(j)}, ..., \theta_d^{(j)})}{\pi(\theta_1^{j}|\theta_2^{(j)}, ..., \theta_d^{(j)})} \frac{q(\theta_1^{j}|\theta_1^{can}, \theta_2^{(j)}, ..., \theta_d^{(j)})}{q(\theta_1^{can}|\theta_1^{j}, \theta_2^{(j)}, ..., \theta_d^{(j)})}\right\}, \tag{7.1}$$

with $\pi(\theta_1^{can}|\theta_2^{(j)}, ..., \theta_d^{(j)})$ denoting the density corresponding to the conditional posterior distribution of θ_1, evaluated at $\theta_1 = \theta_1^{can}$ and similarly for $\pi(\theta_1^{j}|\theta_2^{(j)}, ..., \theta_d^{(j)})$.

Comments:

- In practice, the way to implement the second part of the Metropolis-Hastings algorithm described above is by drawing a value u from a Uniform(0,1) distribution and taking $\theta_1^{(j+1)} = \theta_1^{can}$ if $u < p$ and $\theta_1^{(j+1)} = \theta_1^{j}$ otherwise.
- It is not necessary to be able to simulate from all the conditional posterior distributions, but only from the candidate generator $q(.)$, which can be chosen arbitrarily. Moreover, we only need to know the conditional posterior densities up to proportionality, since any constants of proportionality cancel in the numerator and denominator of the calculation of p in 7.1. However, if $q(.)$ is poorly chosen, then the number of rejections can be high, so the efficiency of the procedure can be very low.

- Another common choice of the candidate generator is to take $q(\theta_1^{can}|\theta_1^j, \theta_2^{(j)}, ..., \theta_d^{(j)})$ to be the density of Normal distribution for θ_1^{can}, with mean θ_1^j and a suitably chosen variance v. This is known as "Random Walk Metropolis algorithm with Normal increments" and it is very popular due to its simplicity. Therefore, the terms involving $q(.)$ cancel in equation 7.1, due to the symmetry of the candidate generator, so the acceptance probability is simplified to

$$p = min\left\{1, \frac{\pi(\theta_1^{can}|\theta_2^{(j)}, ..., \theta_d^{(j)})}{\pi(\theta_1^j|\theta_2^{(j)}, ..., \theta_d^{(j)})}\right\}. \tag{7.2}$$

Gibbs sampling

This is an algorithm used to generate a sequence of samples from the joint posterior distribution (this is often called the target distribution). The algorithm is named after the physicist J. W. Gibbs, in reference to an analogy between the sampling algorithm and statistical physics. It was described by brothers Stuart and Donald Geman (1984).

Gibbs sampling is a special case of the Metropolis-Hastings algorithm and it is applicable when the joint distribution is not known explicitly or is difficult to sample from directly, but the conditional distribution of each variable is known and is easy to sample from. It obtains a sample from the multivariate distribution $\pi(\theta_1, ..., \theta_d)$ by successively and repeatedly simulating from the conditional distributions of each component, given the other components. This is done as follows:

1. Initialize with $\theta = (\theta_1^{(0)}, ..., \theta_d^{(0)})$.
2. Simulate $\theta_1^{(1)}$ from the conditional distribution $\pi(\theta_1|\theta_2^{(0)}, \theta_3^{(0)}, ..., \theta_d^{(0)})$.
3. Simulate $\theta_2^{(1)}$ from the conditional distribution $\pi(\theta_2|\theta_1^{(1)}, \theta_3^{(0)}, ..., \theta_d^{(0)})$.
4. ...
5. Simulate $\theta_d^{(1)}$ from the conditional distribution $\pi(\theta_d|\theta_1^{(1)}, \theta_2^{(1)}, ..., \theta_{d-1}^{(1)})$.
6. Iterate this procedure.

Comments:

- Under conditional conjugacy, the simulation step is usually straightforward.

- Under mild regularity conditions, convergence of the Markov chain to the stationary distribution $\pi(\theta_1, ..., \theta_d)$ is guaranteed. Therefore, after a burn-in period, the subsequent draws can be regarded as realizations from this distribution.

Appendix D

Computational Origins of the Kalman Filter

Based on the model described in section 5.2, we can construct a simpler model as follows:

$$x_t = ax_{t-1} + w_{t-1}$$

$$y_t = cx_t + v_t.$$

We can get an estimate of x_t, which is called $\hat{x}_t$, by reproducing the system. However, if the quantities a, c or the initial value x_0 are unknown the estimate $\hat{x}_t$ is unable to track the exact value of x_t. Additionally, we do not balance the addition of w and v. Therefore, we follow a slightly different procedure.

First, we define the a priori estimate of x_t as:

$$\hat{\hat{x}}_t = a\hat{x}_{t-1}.$$

Then, we use $\hat{\hat{x}}_t$ to predict an estimate for the observed value y_t, which we call $\hat{y}_t$, from:

$$\hat{y}_t = c\hat{\hat{x}}_t.$$

Second, we define the residual as the difference between the actual and the estimated observed value:

$$\text{Residual} = y_t - \hat{y}_t = y_t - c\hat{\hat{x}}_t.$$

There is a good estimate for y_t is the residual is small. Finally, we define the a posteriori estimate of x_t as:

$$\hat{x}_t = \hat{\hat{x}}_t + \kappa(y_t - c\hat{\hat{x}}_t). \tag{7.3}$$

Clearly, this means that if the residual is small (or large) there is a small (or large) correction to our estimate. However, it is needed to find a way to work out the quantity κ (Kalman gain), which enables us to refine our estimate.

Initially, we have to define the errors of our estimates. These are the differences between the true value of x_t and the two estimates (a priori state estimate and a posteriori state estimate). Therefore, we have:

$$\tilde{e}_t = x_t - \hat{\hat{x}}_t \quad \text{(a priori error)} \tag{7.4}$$

$$e_t = x_t - \hat{x}_t \quad \text{(a posteriori error)}. \tag{7.5}$$

We have the following covariances (mean squared errors) associated with the previous errors:

$$\tilde{p}_t = E\{(\tilde{e}_t)^2\} \quad \text{(a priori covariance)} \tag{7.6}$$

$$p_t = E\{(e_t)^2\} \quad \text{(a posteriori covariance)}. \tag{7.7}$$

The Kalman filter chooses suitably the value of κ, so as to minimize the a posteriori covariance p_t. Combining the equations 7.5, 7.7 and 7.3 we have:

$$p_t = E\{(x_t - \hat{x}_t)^2\} = E\{(x_t - \hat{\tilde{x}}_t + \kappa(y_t - c\hat{\tilde{x}}_t))^2\}. \tag{7.8}$$

Then, we differentiate this expression with respect to κ and set this derivative to zero, in order to find the value of κ that minimizes the a posteriori covariance.

$$0 = \frac{\partial p_t}{\partial k} = \frac{\partial E\{(x_t - \hat{\tilde{x}}_t + \kappa(y_t - c\hat{\tilde{x}}_t))^2\}}{\partial k}$$

$$= 2E\{(x_t - \hat{\tilde{x}}_t + \kappa(y_t - c\hat{\tilde{x}}_t))(y_t - c\hat{\tilde{x}}_t)\}$$

$$= 2E\{x_t y_t - \hat{\tilde{x}}_t y_t - \kappa y_t^2 + \kappa c\hat{\tilde{x}}_t y_t - c x_t \hat{\tilde{x}}_t + (\hat{\tilde{x}}_t)^2 + \kappa c y_t \hat{\tilde{x}}_t - \kappa c^2 (\hat{\tilde{x}}_t)^2\}$$

$$= 2E\{x_t y_t - \hat{\tilde{x}}_t y_t - c x_t \hat{\tilde{x}}_t + (\hat{\tilde{x}}_t)^2\} - 2\kappa E\{y_t^2 - 2c\hat{\tilde{x}}_t y_t + c^2(\hat{\tilde{x}}_t)^2\}.$$

Therefore, we have:

$$\kappa = \frac{E\{x_t y_t - \hat{\tilde{x}}_t y_t - c x_t \hat{\tilde{x}}_t + c(\hat{\tilde{x}}_t)^2\}}{E\{y_t^2 - 2c\hat{\tilde{x}}_t y_t + c^2(\hat{\tilde{x}}_t)^2\}}. \tag{7.9}$$

Let us now consider the numerator and the denominator of the previous quantity separately, in order to simplify it.

$$\text{Numerator} = E\{x_t y_t - \hat{\tilde{x}}_t y_t - c x_t \hat{\tilde{x}}_t + c(\hat{\tilde{x}}_t)^2\}$$

$$= E\{x_t(c x_t + v_t) - \hat{\tilde{x}}_t(c x_t + v_t) - c x_t \hat{\tilde{x}}_t + c(\hat{\tilde{x}}_t)^2\}$$

$$= E\{c x_t^2 - x_t v_t - c\hat{\tilde{x}}_t x_t - \hat{\tilde{x}}_t v_t - c x_t \hat{\tilde{x}}_t + c(\hat{\tilde{x}}_t)^2\}$$

$$= E\{c x_t^2 - 2c\hat{\tilde{x}}_t x_t + c(\hat{\tilde{x}}_t)^2 + (x_t - \hat{\tilde{x}}_t) v_t\}.$$

One of our model's assumptions was that the measurement noise v_t is uncorrelated to the hidden state x_t and as a consequence uncorrelated to the a priori estimate of x_t. Therefore we have:

$$E\{(x_t - \hat{\tilde{x}}_t) v_t\} = E(x_t v_t) - E(\hat{\tilde{x}}_t v_t) = 0 \tag{7.10}$$

Using this the numerator becomes:

$$\text{Numerator} = E\{cx_t^2 - 2c\hat{\tilde{x}}_t x_t + c(\hat{\tilde{x}}_t)^2\}$$

$$= cE\{(x_t - \hat{\tilde{x}}_t)^2\} = cE\{(\tilde{e}_t)^2\} = c\tilde{p}_t. \tag{7.11}$$

Now, let us consider the denominator:

$$\text{Denominator} = E\{y_t^2 - 2c\hat{\tilde{x}}_t y_t + c^2(\hat{\tilde{x}}_t)^2\}$$

$$= E\{(cx_t + v_t)^2 - 2c\hat{\tilde{x}}_t(cx_t + v_t) + c^2(\hat{\tilde{x}}_t)^2\}$$

$$= E\{c^2x_t^2 + 2cx_tv_t + v_t^2 - 2c^2\hat{\tilde{x}}_t x_t - 2c\hat{\tilde{x}}_t v_t + c^2(\hat{\tilde{x}}_t)^2\}$$

$$= E\{(cx_t)^2 - 2c^2\hat{\tilde{x}}_t x_t + c^2(\hat{\tilde{x}}_t)^2 + v_t^2 + 2c(x_t - \hat{\tilde{x}}_t)v_t\}.$$

Again by using equation 7.10 the last term is set to zero and the denominator becomes simpler:

$$\text{Denominator} = E\{(cx_t)^2 - 2c^2\hat{\tilde{x}}_t x_t + c^2(\hat{\tilde{x}}_t)^2 + v_t^2\}$$

$$= c^2E\{x_t^2 - 2\hat{\tilde{x}}_t x_t + (\hat{\tilde{x}}_t)^2\} + E\{v_t^2\}$$

$$= c^2E\{(x_t - \hat{\tilde{x}}_t)^2\} + R$$

$$= c^2E\{(\tilde{e}_t)^2\} + R = c^2\tilde{p}_t + R. \tag{7.12}$$

Therefore, by using the equations 7.11 and 7.12 for the numerator and denominator we have the following simple expression of κ:

$$\kappa = \frac{c\tilde{p}_t}{c^2\tilde{p}_t + R}. \tag{7.13}$$

This expression needs a value for the a priori covariance $\tilde{p}_t$, which we will try to find by using its definition given in the equation 7.6.

$$\tilde{p}_t = E\{(\tilde{e}_t)^2\} = E\{(x_t - \hat{\tilde{x}}_t)^2\}$$

$$= E\{(ax_{t-1} + w_t - a\hat{x}_{t-1})^2\}$$

$$= E\{a^2(x_t - \hat{x}_{t-1})^2 + 2aw_t(x_t - \hat{x}_{t-1}) + w_t^2\}.$$

The process noise w_t is uncorrelated to the hidden state x_t and as a consequence uncorrelated to the a priori estimate of x_t. Therefore, we have:

$$E\{w_t(x_t - \hat{x}_{t-1})\} = E(w_t x_{t-1}) - E(w_t\hat{x}_{t-1}) = 0. \tag{7.14}$$

So, the a priori covariance becomes:

$$\tilde{p}_t = E\{a^2(x_t - \hat{x}_{t-1})^2 + w_t^2\} = a^2 E\{(x_t - \hat{x}_{t-1})^2\} + E\{w_t^2\}$$

$$= a^2 E\{(e_{t-1})^2\} + Q = a^2 p_{t-1} + Q. \tag{7.15}$$

Now, this expression needs a value for the a posteriori covariance p_{t-1}. Again, we start from its definition and we have:

$$p_t = E\{(e_t)^2\} = E\{(x_t - \hat{x}_t)^2\} =$$

$$= E\{[x_t - (\hat{\tilde{x}}_t + \kappa(y_t - c\hat{\tilde{x}}_t))]^2\}$$

$$= E\{[x_t - (\hat{\tilde{x}}_t - \kappa c\hat{\tilde{x}}_t + \kappa(cx_t + v_t))]^2\}$$

$$= E\{(x_t - \hat{\tilde{x}}_t + \kappa c\hat{\tilde{x}}_t - \kappa c x_t - \kappa v_t)^2\}$$

$$= E\{[(1 - \kappa c)(x_t - \hat{\tilde{x}}_t) - \kappa v_t]^2\}$$

$$= E\{(1 - \kappa c)^2(x_t - \hat{\tilde{x}}_t)^2 - 2\kappa v_t(1 - \kappa c)(x_t - \hat{\tilde{x}}_t) + (\kappa v_t)^2\}.$$

Using equation 7.10 the a posteriori covariance becomes:

$$p_t = E\{(1 - \kappa c)^2(x_t - \hat{\tilde{x}}_t)^2 + (\kappa v_t)^2\}$$

$$= (1 - \kappa c)^2 E\{(x_t - \hat{\tilde{x}}_t)^2\} + \kappa^2 E\{(v_t)^2\}$$

$$= (1 - \kappa c)^2 E\{(\tilde{e}_t)^2\} + \kappa^2 R$$

$$= (1 - \kappa c)^2 \tilde{p}_t + \kappa^2 R \tag{7.16}$$

Using equations 7.13 and 7.16 we have:

$$\kappa = \frac{c\tilde{p}_t}{c^2\tilde{p}_t + R}$$

$$\Rightarrow \kappa(c^2\tilde{p}_t + R) = c\tilde{p}_t$$

$$\Rightarrow R = \frac{c\tilde{p}_t}{\kappa} - c^2\tilde{p}_t = \frac{c\tilde{p}_t(1 - c\kappa)}{\kappa}. \tag{7.17}$$

Using equations 7.16 and 7.17 we have:

$$p_t = (1 - \kappa c)^2\tilde{p}_t + \kappa^2\frac{c\tilde{p}_t(1 - c\kappa)}{\kappa}$$

$$= \tilde{p}_t(1 - 2\kappa c + \kappa^2 c^2 \kappa c - \kappa^2 c^2)$$

$$= \tilde{p}_t(1 - \kappa c). \tag{7.18}$$

Summary

To sum up, we start with a system described as

$$x_t = ax_{t-1} + w_{t-1}$$

$$y_t = cx_t + v_t,$$

where $w_t \sim N(0, R)$ and $v_t \sim N(0, Q)$.
We calculate the a priori state estimate based on the previous state estimate:

$$\hat{\tilde{x}}_t = a\hat{x}_{t-1}.$$

We calculate the a priori covariance:

$$\tilde{p}_t = a^2 p_{t-1} + Q.$$

Then, we calculate the Kalman gain κ using:

$$\kappa = \frac{c\tilde{p}_t}{c^2\tilde{p}_t + R}.$$

Then, we correct the a priori estimates (state and covariance) to obtain the a posteriori state estimate and the a posteriori covariance as follows:

$$\hat{x}_t = \hat{\tilde{x}}_t + \kappa(y_t - c\hat{\tilde{x}}_t)$$

$$p_t = \tilde{p}_t(1 - c\kappa).$$

Kalman Filter algorithm

1. Choose initial values $\hat{x}_0$ and P_0 (state and covariance estimates).

2. Given that the chain is at step t, our parameters are $\hat{x}_t$ and P_t. We compute $\hat{\tilde{x}}_{t+1}$ and $\hat{P}_{t+1}$ from:

$$\hat{\tilde{x}}_{t+1} = A\hat{x}_t$$

$$\hat{P}_{t+1} = AP_tA^T + Q$$

3. We compute the Kalman gain K_{t+1} from:

$$K_{t+1} = \tilde{P}_{t+1}C^T(C\tilde{P}_{t+1}C^T + R)^{-1}$$

   We obtain new measurement (observation) y_{t+1} from:

$$y_{t+1} = C\hat{\tilde{x}}_{t+1}$$

4. We correct our estimates:

$$\hat{x}_{t+1} = \hat{\tilde{x}}_{t+1} + K_{t+1}(y_{t+1} - C\hat{\tilde{x}}_{t+1})$$

$$P_{t+1} = (I - K_{t+1}C)\tilde{P}_{t+1}$$

5. We iterate this procedure.

Appendix E

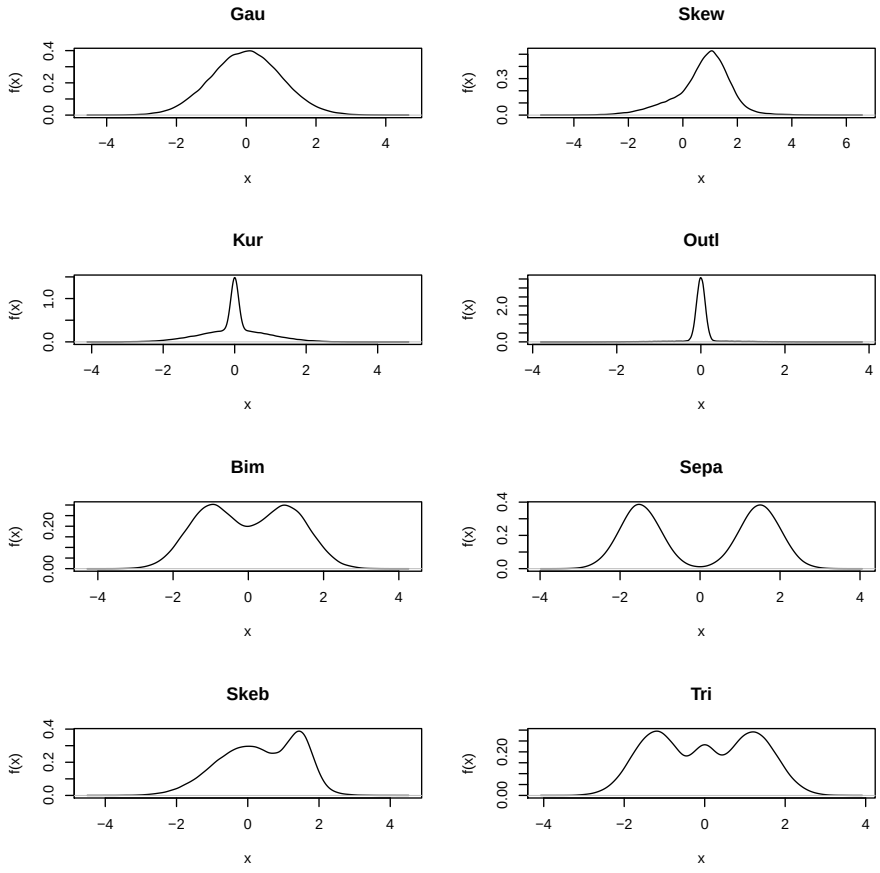

Figure 7.1: Mixtures of Normal distributions.

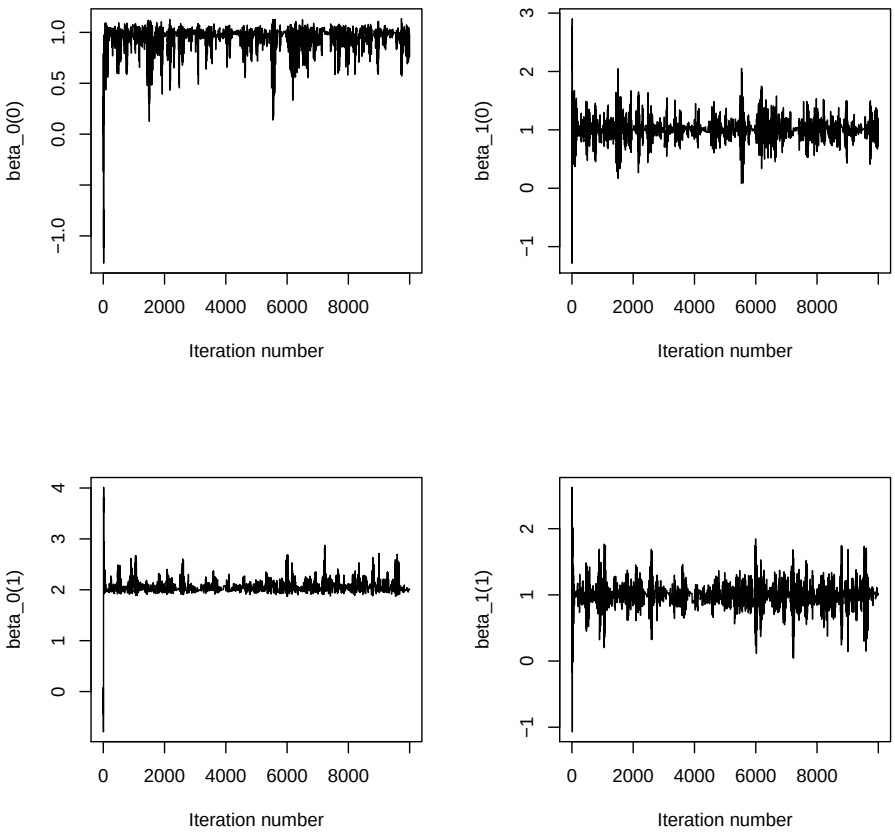

Figure 7.2: Convergence of model 1 parameters.

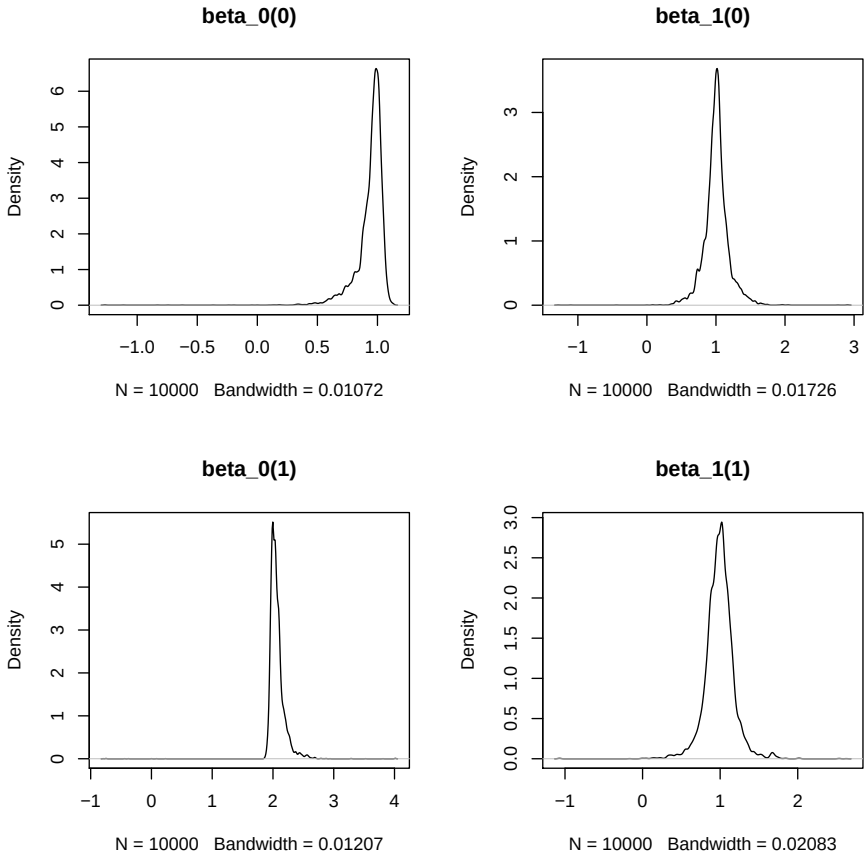

Figure 7.3: Density plot of model 1 parameters.

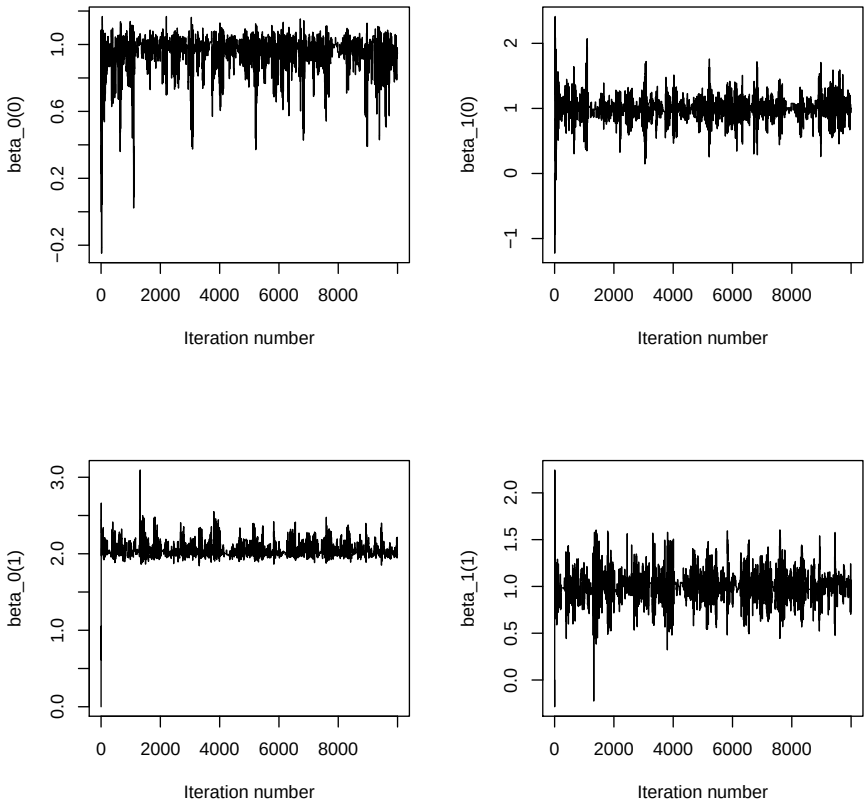

Figure 7.4: Convergence of model 2 parameters.

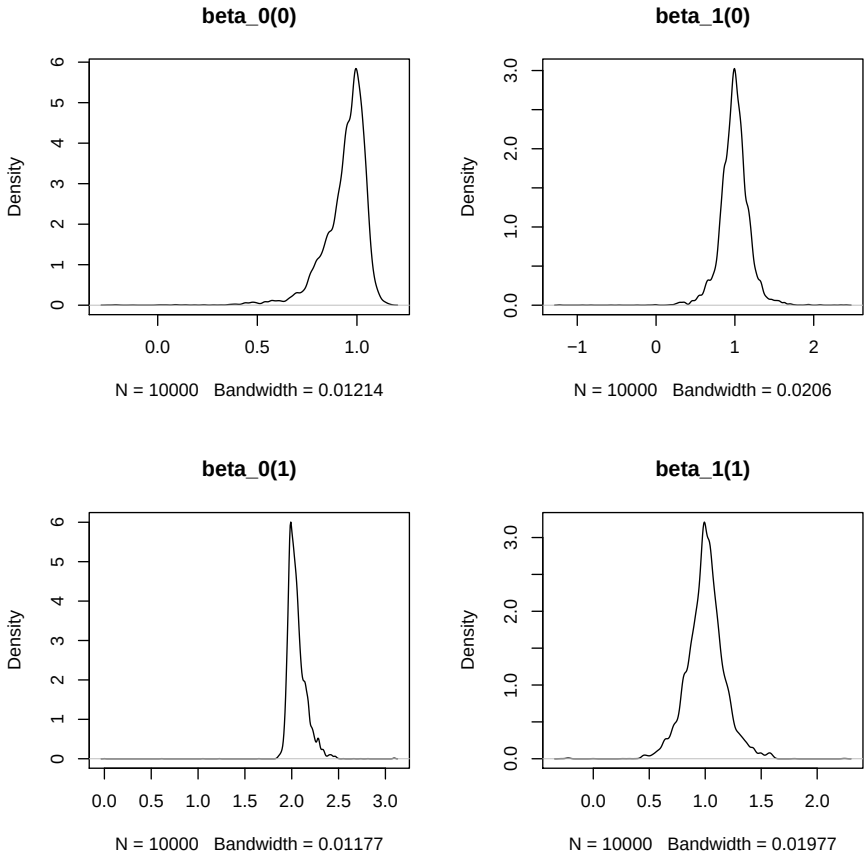

Figure 7.5: Density plot of model 2 parameters.

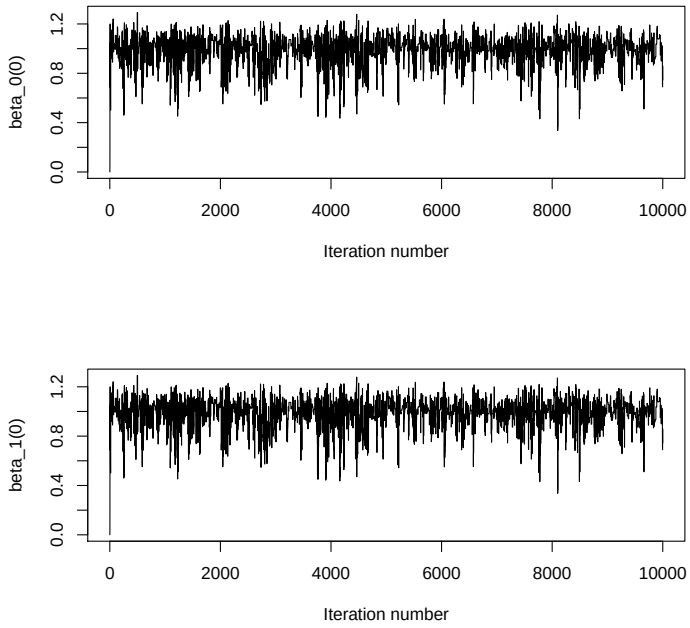

Figure 7.6: Convergence of model 3 parameters.

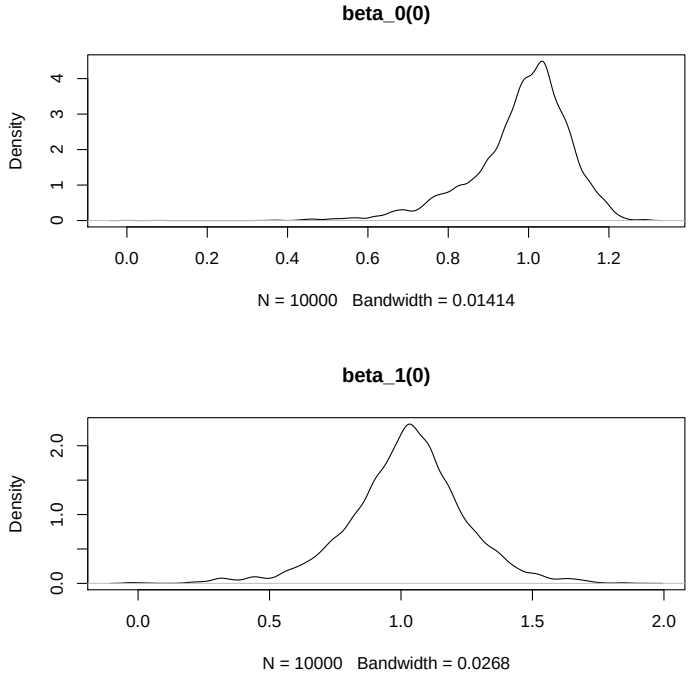

Figure 7.7: Density plot of model 3 parameters.

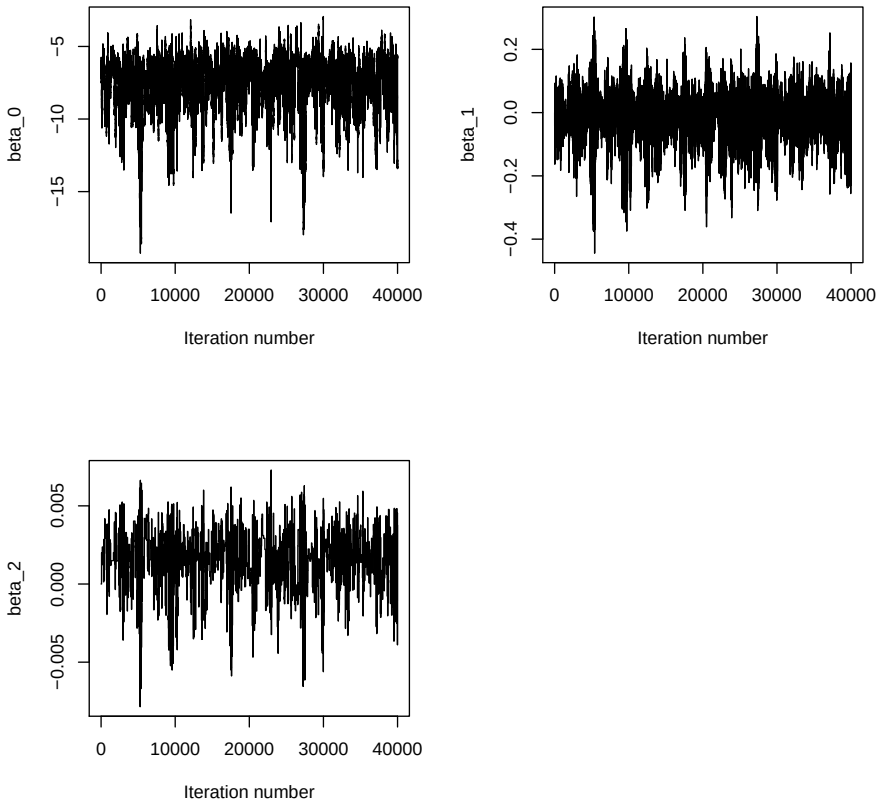

Figure 7.8: Convergence of the quadratic model parameters, for $p = 0.001$.

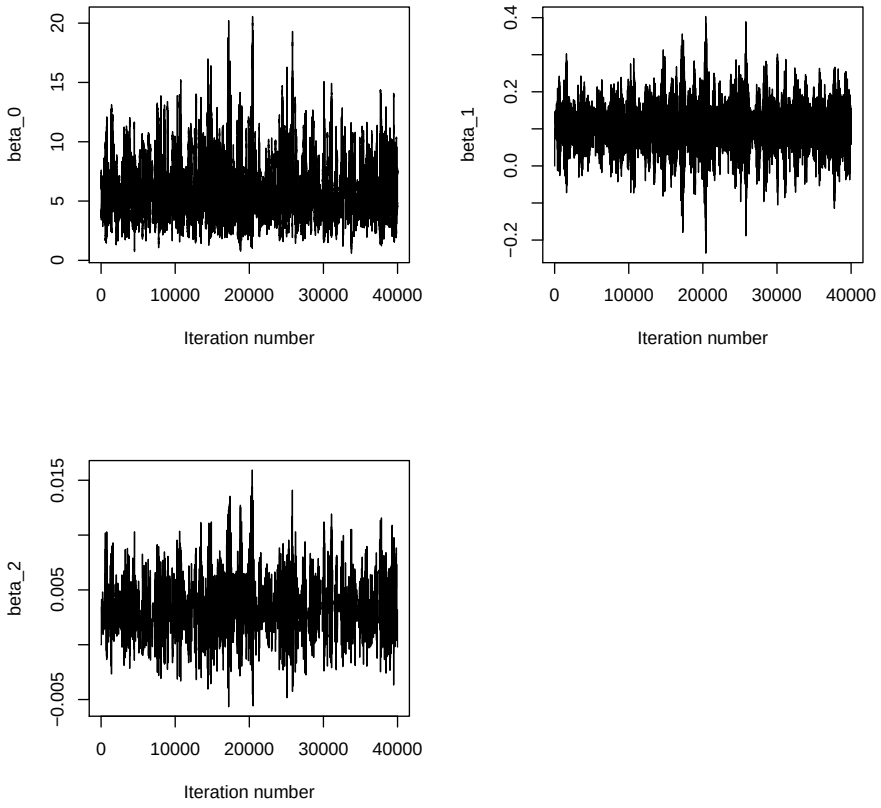

Figure 7.9: Convergence of the quadratic model parameters, for $p = 0.999$.

Appendix F

Traceplots of Model Parameters

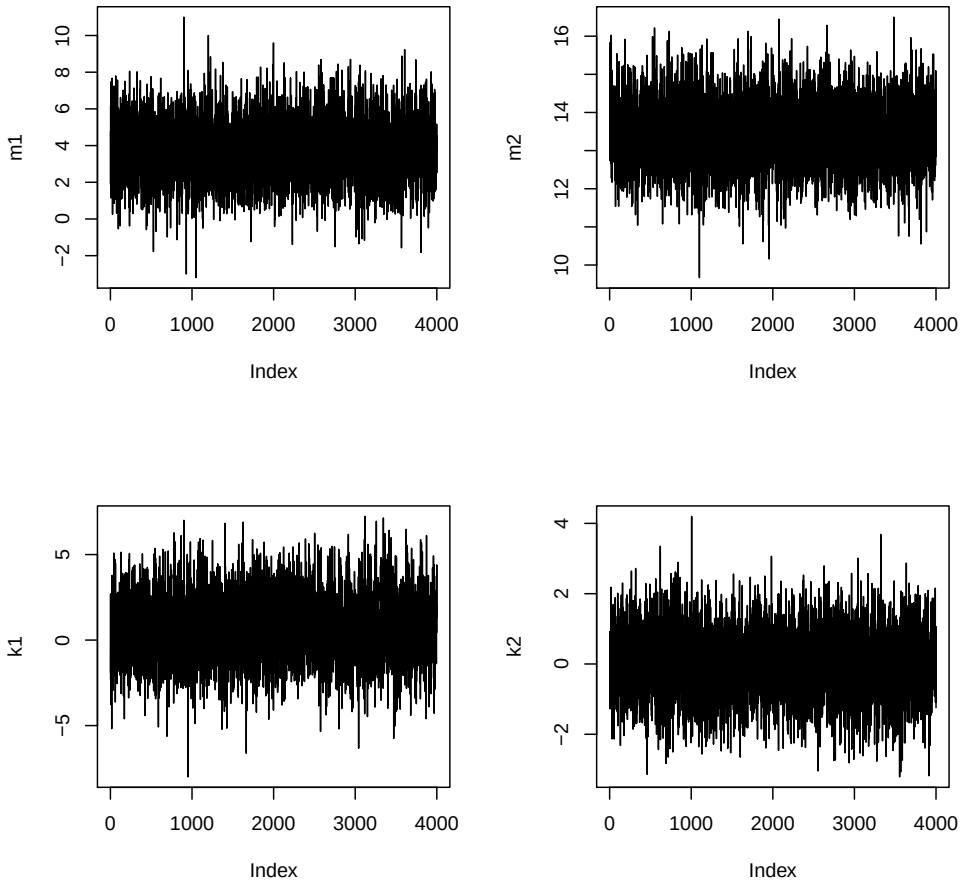

Figure 7.10: Traceplots of the estimates of the Normal parameters for the US ex-post real interest rates, for the 2-state Normal HMM, for the highest extreme quantile. The first two correspond to the means μ_i and the other two correspond to the precisions κ_i, $i = 1, 2$.

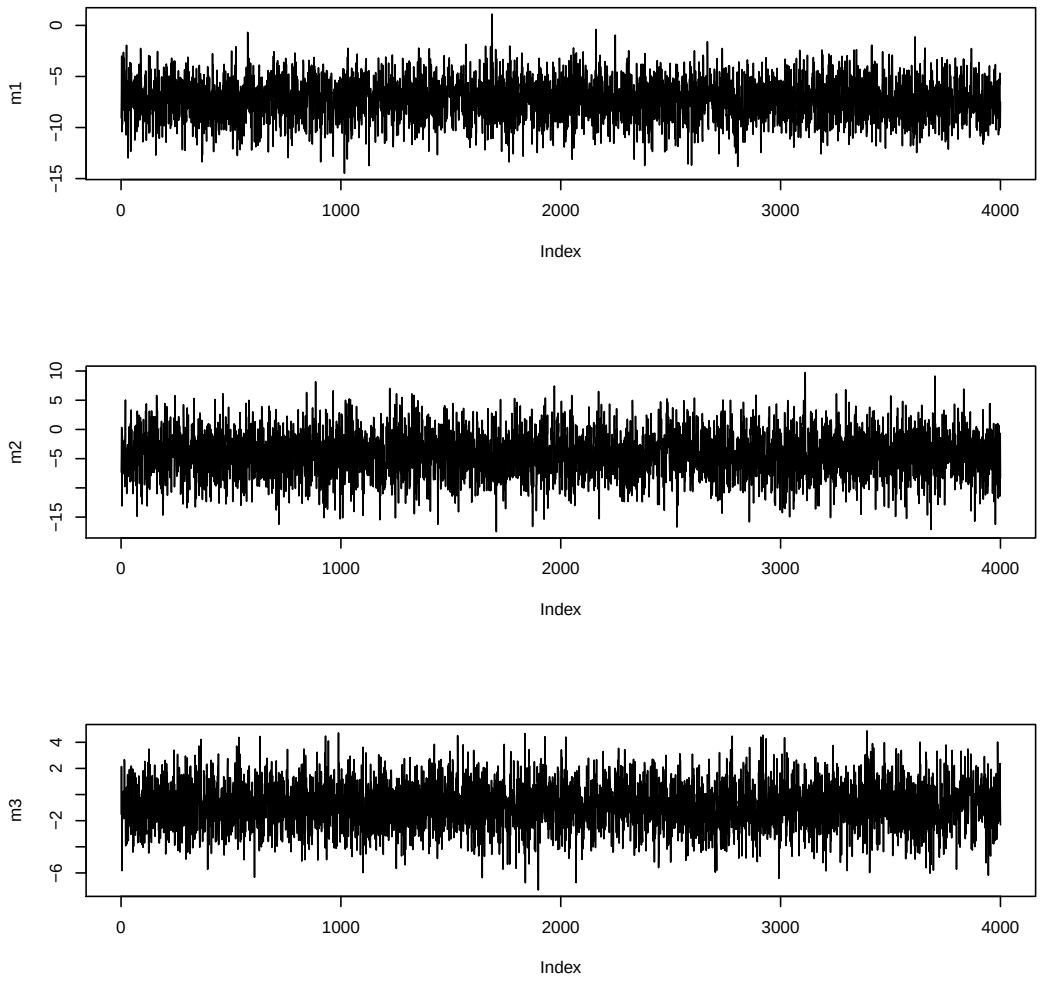

Figure 7.11: Traceplots of the estimates of the ALD parameters for the US ex-post real interest rates, for the 3-state ALD HMM, for the lowest extreme quantile. They represent the location parameters $\mu_i, i = 1, 2, 3$.

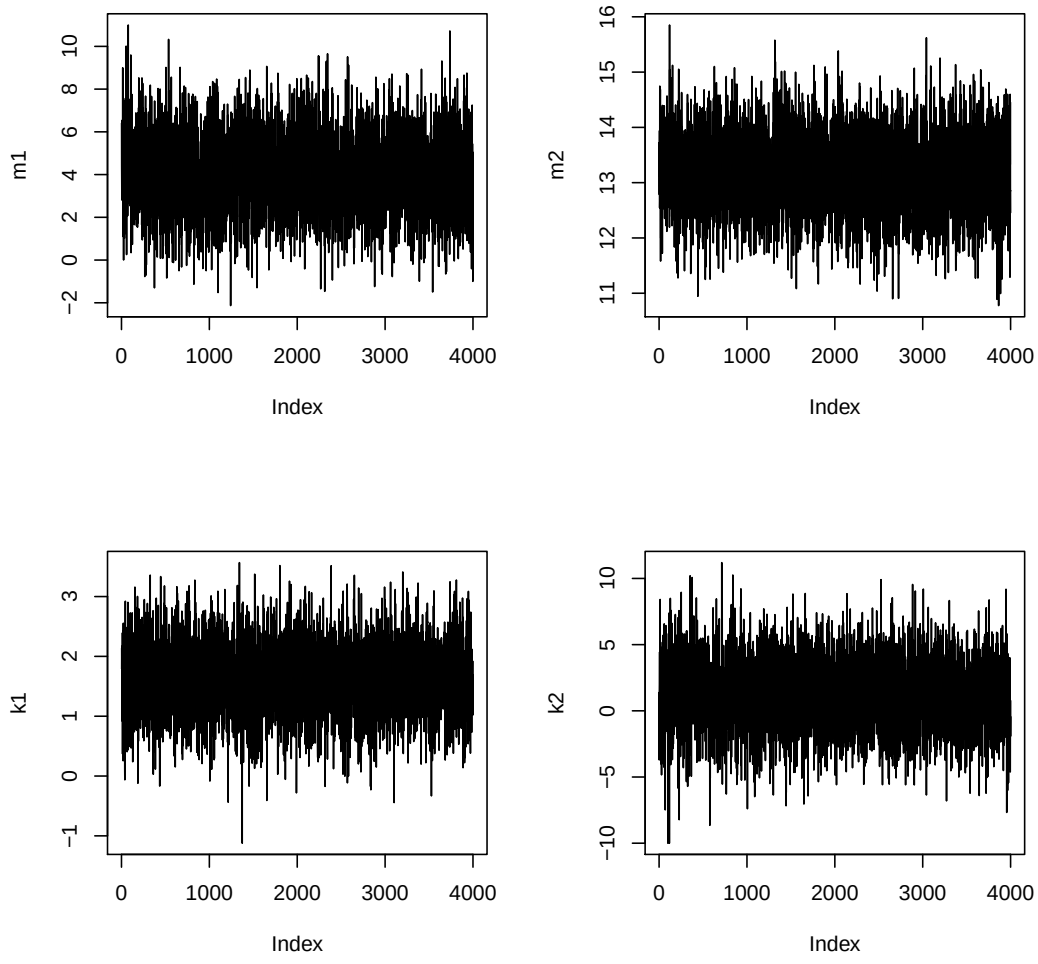

Figure 7.12: Traceplots of the estimates of the Normal parameters for the US ex-post real interest rates, for the single break-point Normal HMM, for the highest extreme quantile. They represent the means μ_i, and the precisions κ_i, $i = 1, 2$.

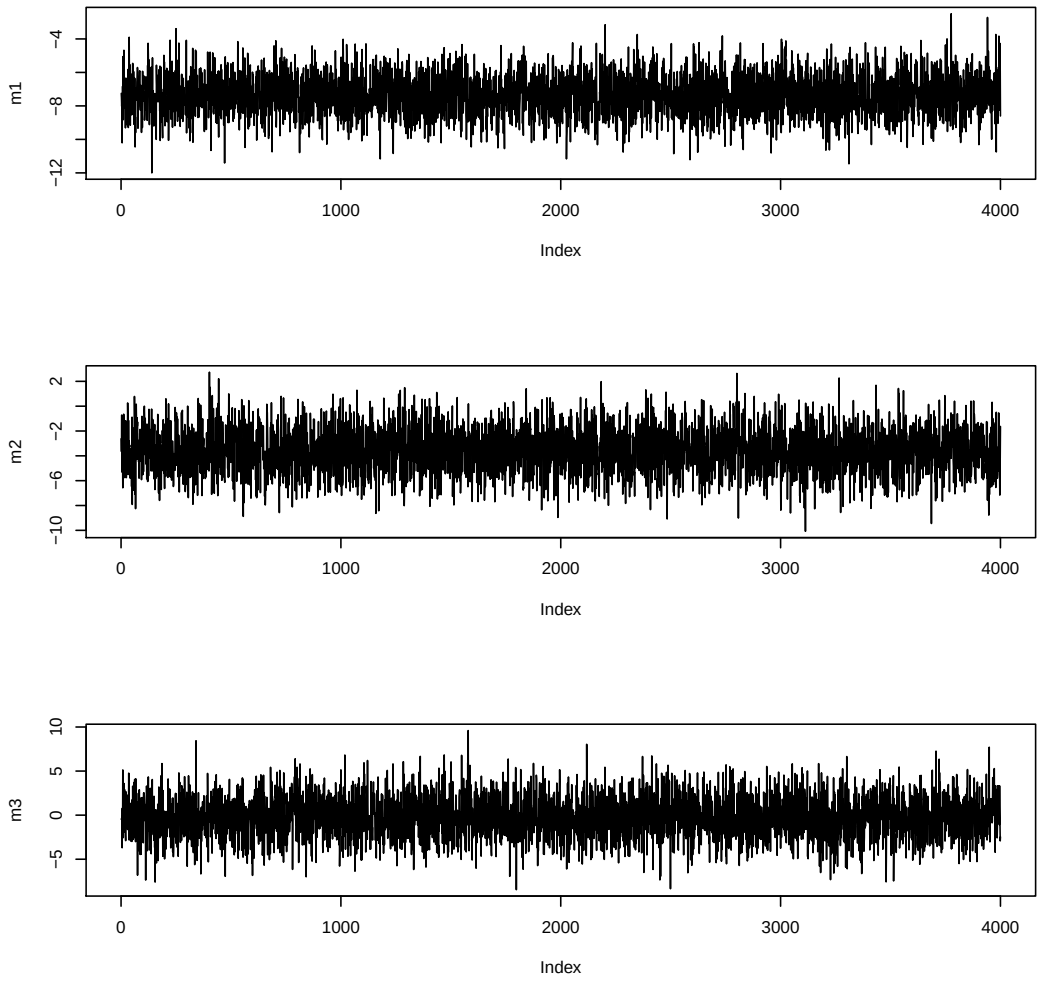

Figure 7.13: Traceplots of the estimates of the ALD parameters for the US ex-post real interest rates, for the 2 break-point ALD HMM, for the lowest extreme quantile. They represent the location parameters μ_i, $i = 1, 2, 3$.

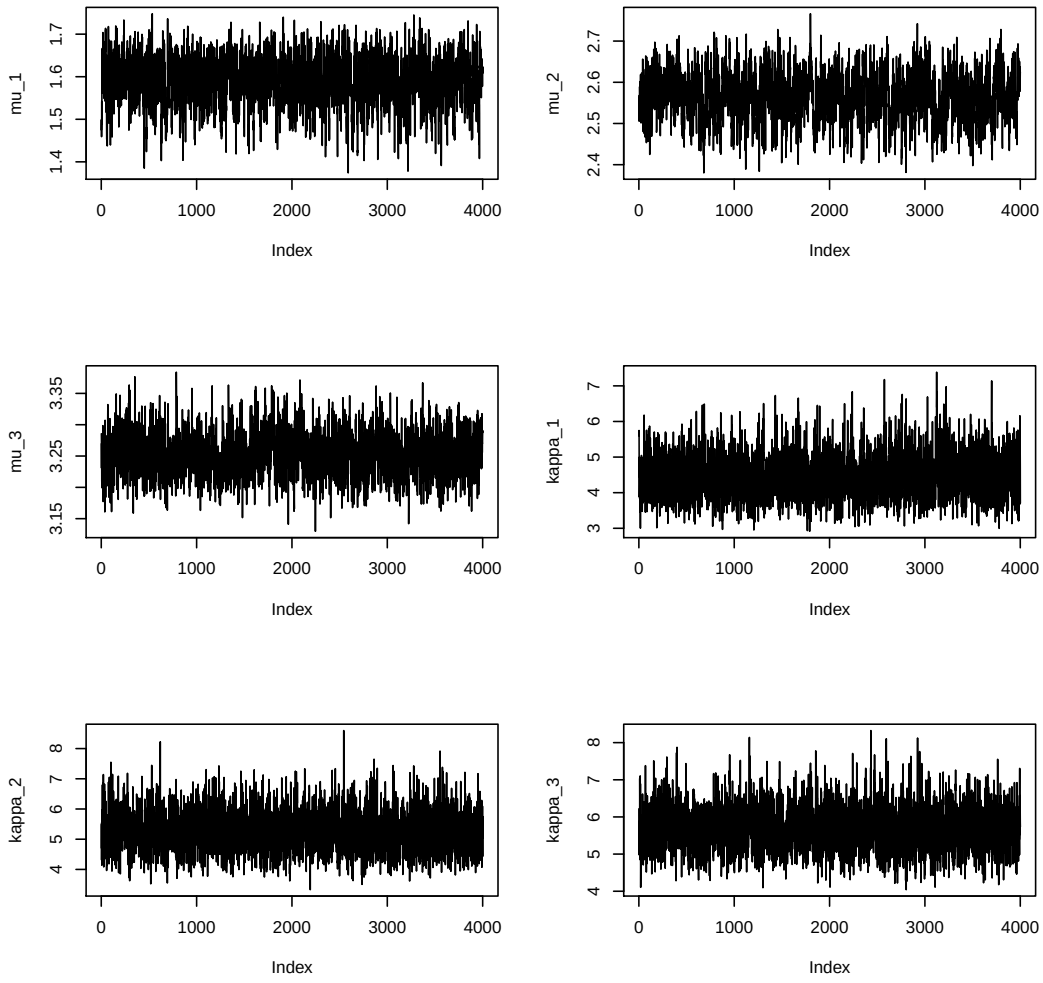

Figure 7.14: Traceplots of the estimates of the Normal parameters for the US real interest rates, for the 3-state Normal HMM, for the lowest extreme quantile. The first three correspond to the means μ_i and the other three correspond to the precisions κ_i, $i = 1, 2, 3$.

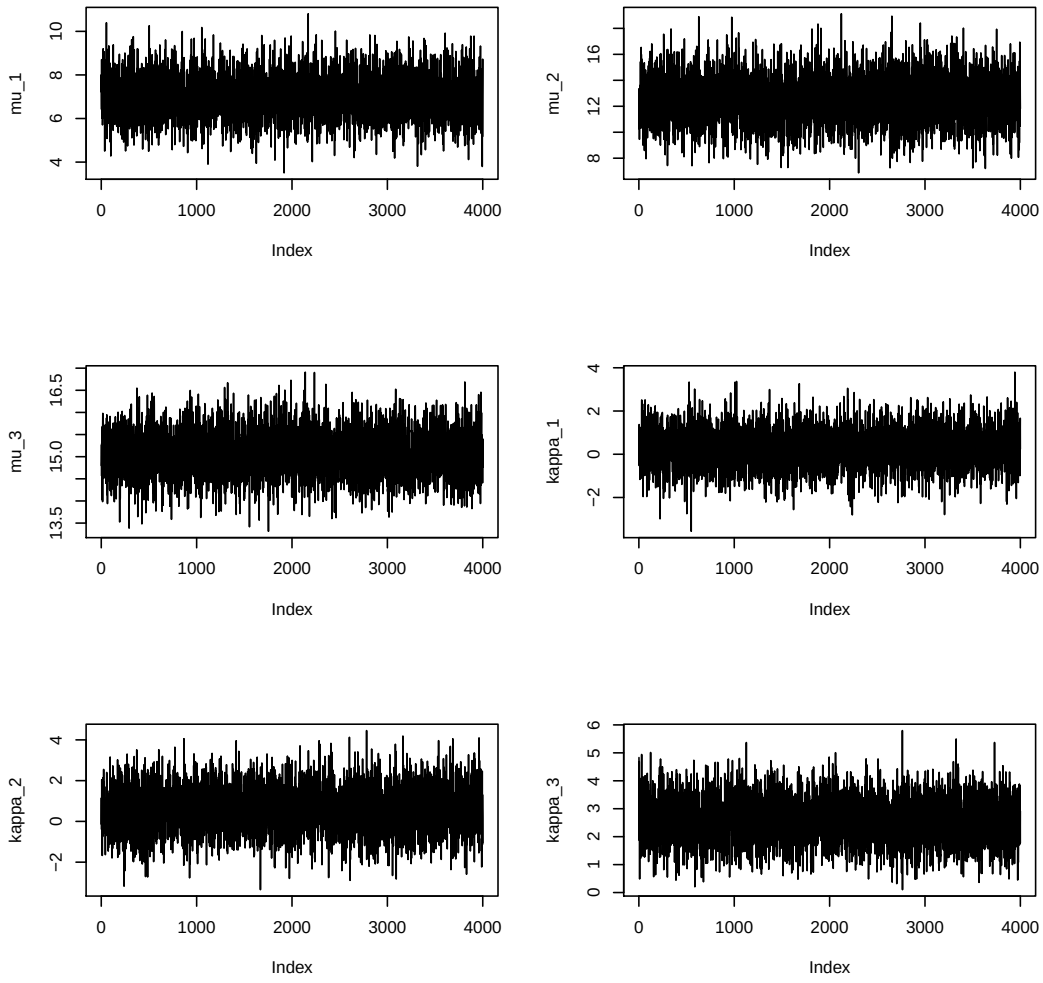

Figure 7.15: Traceplots of the estimates of the Normal parameters for the US real interest rates, for the 3-state Normal HMM, for the highest extreme quantile. The first three correspond to the means μ_i and the other three correspond to the precisions κ_i, $i = 1, 2, 3$.

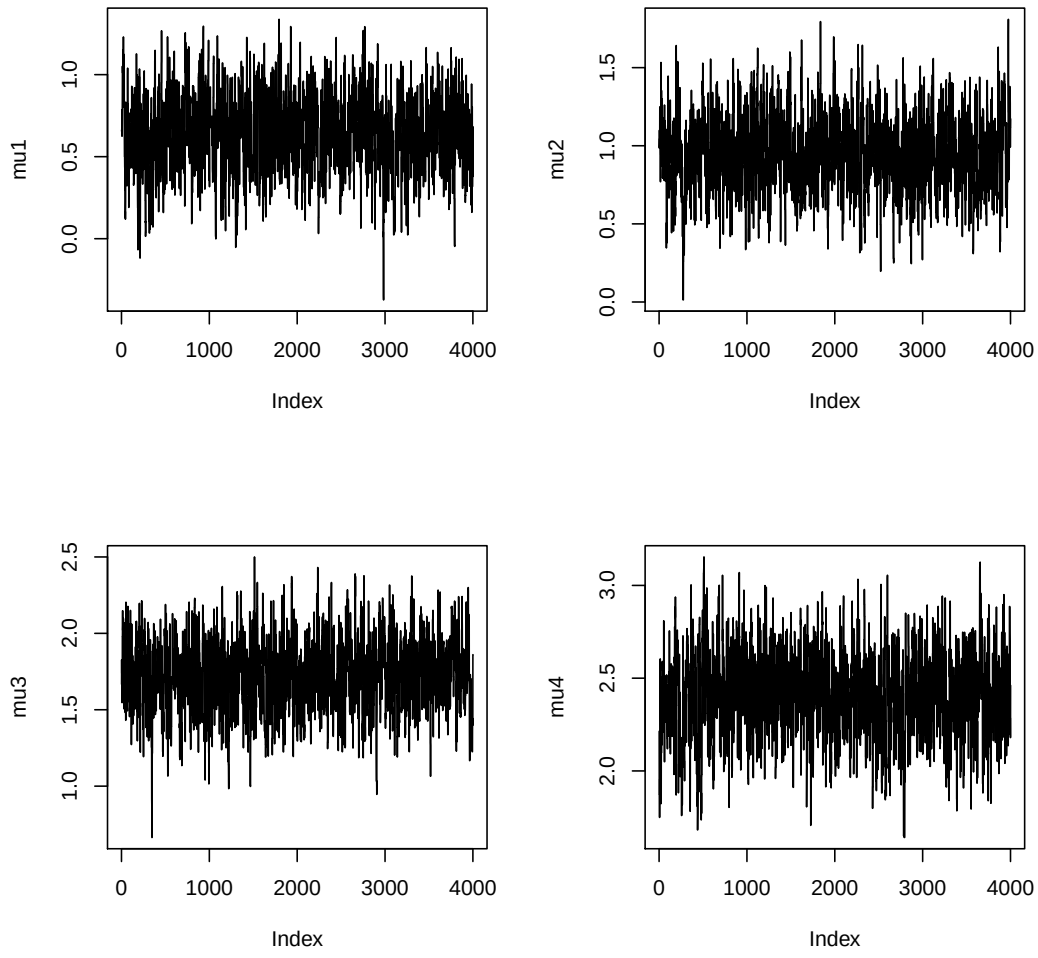

Figure 7.16: Traceplots of the estimates of the ALD parameters for the US real interest rates, for the 4-state ALD HMM, for the lowest extreme quantile. They represent the location parameters μ_i, $i = 1, 2, 3, 4$.

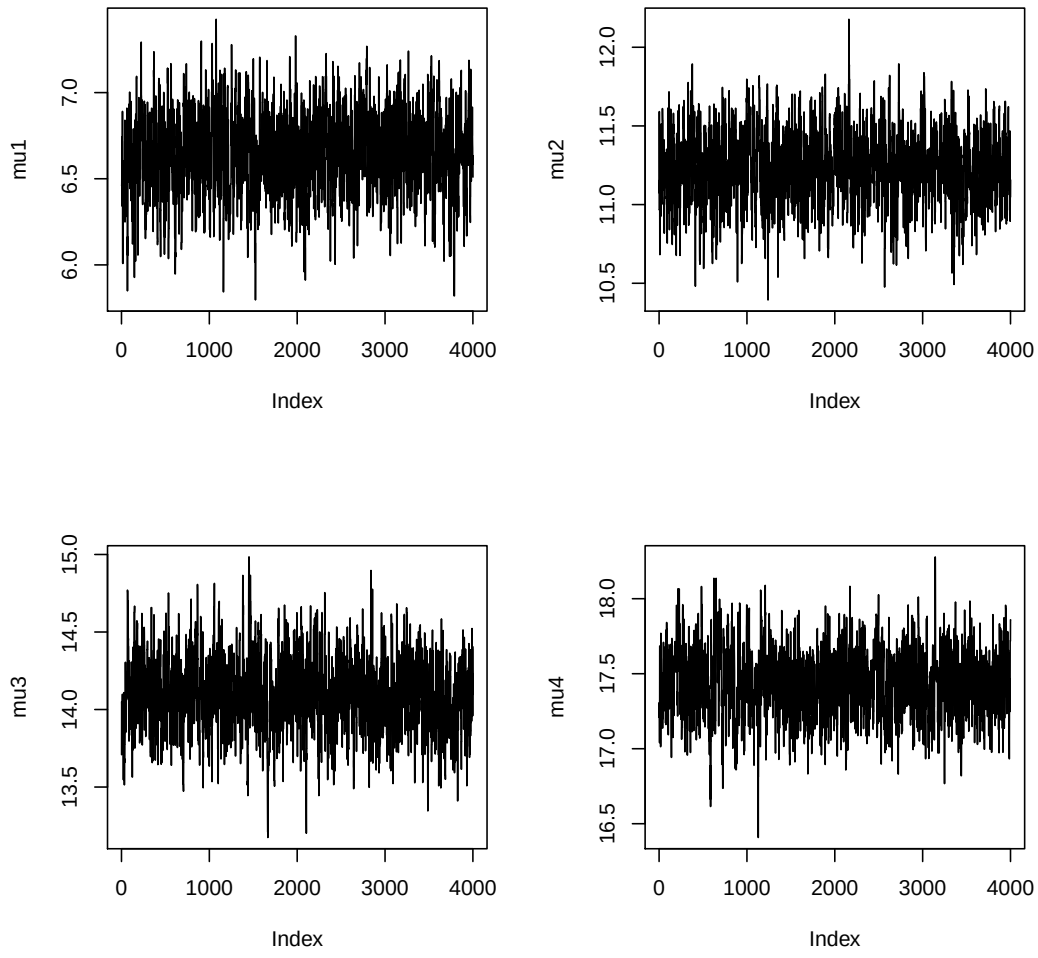

Figure 7.17: Traceplots of the estimates of the ALD parameters for the US real interest rates, for the 4-state ALD HMM, for the highest extreme quantile. They represent the location parameters μ_i, $i = 1, 2, 3, 4$.

Bibliography

1. Abel, A. B. and Eberly, J. C. (1994). A unified model of investment under uncertainty. *American Economic Review*. **84** (5), 13691384.

2. Abreveya, J. (2001). The Effects of Demographics and Maternal Behavior on the Distribution of Birth Outcomes. *Journal of Economics*. **26**, 247-257.

3. Albert, J. H. and Chib, S. (1993). Bayes inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts. *Journal of Buisiness and Economic Statistics*. **11**, 1-15.

4. Andrews, D. W. K.,Lee, I., Ploberger, W. (1996). Optimal changepoint tests for normal linear regression. *Journal of Econometrics*. **70**, 9-38.

5. Andrieu, C. and Douchet, A. (1999). Joint Bayesian model selection and estimation of noisy sinusoids via reversible jump MCMC. *IEEE Trans. Signal Process*. **47**, 2667-2676.

6. Bai, J. and Perron, P. (2003). Computation and analysis of multiple structural change models. *Journal of Econometrics*. John Wiley and Sons, Ltd. **18**, 1-22.

7. Bakis, R. (1976). Continuous speech recognition via centisecond acoustic states. Presented at the 91st Meeting of the Acoustical Society of America.

8. Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities*. **3**, 1-8.

9. Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970). A maximisation technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*. **41**, 164-171.

10. Beck, N., and Katz, J. (2007). Random Coefficient Models for Time-Series-Cross-Section Data: Monte Carlo Experiments, *Political Analysis*. **15**, 182195.

11. Bedi, A. S., Edwards, J. H. Y. (2002). The impact of School Quality on Earnings and Educational Returns: Evidence from a Low-Income Country. *Journal of Development Economics*. **68**(1), 157-185.

12. Bishop, C. M. (1995). Neural Networks for Pattern Recognition. Oxford University Press.

13. Brown, R. G. and Hwang, P. Y. C. (1992). Introduction to Random Signals and Applied Kalman Filtering, Second Edition, John Wiley & Sons Inc.

14. Buchinsky, M. (1994). Wage Structure 1963-1987: Application of Quantile Regression. *Econometrica*. **62**, 405-458.

15. Budd, J. W., McCall, B. (2001). The Grocery Stores Wage Distribution: A Semi-parametric Analysis of the Role of Retailing and Labor Market Institutions. *Industrial and Labor Relations Review*. **54**(2), 483-501.

16. Cappé, O., Moulines, E. and Rydén, T. (2005). Inference in Hidden Markov Models. New York: Springer.

17. Castellano, R. and Scaccia, L. (2007). Bayesian inference for Hidden Markov Models. Working Papers 43-2007, Macerata University, Department of Finance and Economic Sciences, revised Oct 2008.

18. Cerra, V. and Saxena, S. C. (2005). Did Output Recover from the Asian Crisis?. *IMF Staff Papers*. **52**, 1-23.

19. Chamberlain, G. (1994). Quantile regression, censoring, and the structure of wages. *Advances in Econometrics*, Sixth World Congress, Vol. 1, 171-209.

20. Chay, K. Y., Honore, B. E. (1998). Estimation of semiparametric censored regression models: an application to changes in black-white earnings inequality during the 1960s. *Journal of Human Resources*. **33**, 4-38.

21. Chernozhukov, V. (2005). Extremal Quantile Regression. *The Annals of Statistics*. **33**, No. 2, 806-839.

22. Chib, S. (1998). Estimation and comparison of multiple change-point models. *Journal of Econometrics*. **86**, 221-241.

23. Churchill, G. A. (1989). Stochastic models for heterogeneous DNA sequences. *Bulletin of Mathematical Biology*. **51**, 79-94.

24. Cole, T. J., Green, P. J. (1992). Smoothing Reference Centile Curves: The LMS Method and Penalized Likelihood. *Statistics in Medicine*. **11**, 1305-1319.

25. Davig, T. (2004). Regime-Switching Debt and Taxation. *Journal of Monetary Economics*. **51**, 837-859.

26. Delorio , M. and Robert, C. P. (2002). Discussion on the paper by Spiegelhalter, Best, Carlin and van der Linde (2002). *Journal Royal Statistics, Series B*. **64**, 629-630.

27. De Rossi, G. and Harvey, A. C. (2006). Time-Varying Quantiles. CWPE 0649, University of Cambridge.

28. De Rossi, G., Harvey, A. (2009). Quantiles, expectiles and splines. *Journal of Econometrics*. doi:10.1016/j.jeconom.2009.01.001

29. Eide, E., Showalter, M. (1999). Factors Affecting the Transmission of Earnings Across Generations: A Quantile Regression Approach. *Journal of Human Resources*. **34**(2), 253-267.

30. Eide, E., Showalter, M., Sims, D. P. (2002). The Effects of Secondary School Quality on the Distribution of Earnings. *Contemporary Economic Policy*. **20**, 160-170.

31. Engel, C. and Hamilton, J. D. (1990). Long swings in the dollar: are they in the data and do markets know it?.*American Economic Review*. **80**, 689-713.

32. Farcomeni, A. (2010). Quantile regression for longitudinal data based on latent Markov subject-specific parameters. *Statistics and Computing*. 1-12. doi:10.1007/s11222-010-9213-0.

33. Fortin, N. M., Lemieux, T. (1998). Rank Regressions, Wage Distributions and the Gender Gap. *Journal of Human Resources*. **33**(3), 610-643.

34. Fox, E., Sudderth, E., Jordan, M. and Willsky, A. (2009). The sticky HDP-HMM: Bayesian nonparametric hidden Markov models with persistent states.

35. Fredkin , D. R. and Rice, J. A. (1992a). Bayesian restoration of single-channel patch clamp recordings. *Biometrics*. **48**, 427-448.

36. Fredkin , D. R. and Rice, J. A. (1992b). Maximum likelihood estimation and identification directly from single-channel recordings. *Proceedings of the Royal Society of London B.* **249**, 125-132.

37. Garcia, R. and Perron, P. (1996). An Analysis of the Real Interest Rate Under Regime Shifts, *The Review of Economics and Statistics*. **78**, 111-125.

38. Gelb, A. (1974). Applied Optimal Estimation, MIT Press, Cambridge, MA.

39. Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **6**(6), 721741.

40. Geraci, M. and Bottai, M. (2007). Quantile regression for longitudinal data using the asymmetric Laplace distribution. *Biostatistics*. **8**, 140154.

41. Gerlach, R. H., Chen, C. W. S., Chan, N. Y. C. (2011). Bayesian Time-Varying Quantile Forecasting for Value-at-Risk in Financial Markets. *Journal of Business and Economic Statistics*. **29** (4), 481-492.

42. Gopich, I. V. and Szabo, A. (2009). Decoding the pattern of photon colors in single-molecule FRET. *J. Phys. Chem.* B. **113**, 10965-10973.

43. Grewal, M. S. and Andrews, A. P. (1993). Kalman Filtering Theory and Practice. Upper Saddle River, NJ USA, Prentice Hall.

44. Guttorp, P. (1995). Stochastic Modeling of Scientific Data. Chapman and Hall. London.

45. Hamilton, J. D. (1988). Rational-Expectations Econometric Analysis of Changes in Regime: An Investigation of the Term Structure of Interest Rates. *Journal of Economic Dynamics and Control*. **12**, 385-423.

46. Hamilton, J. D. (1989). A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle, *Econometrica*. **57**, 357-384.

47. Hamilton, J. D. (1990). Analysis of time series subject to changes in regime. *Journal of Econometrics*. **45**, 39-70.

48. Hamilton, J. D. (2005). Whats Real About the Business Cycle?. Federal Reserve Bank of St. Louis Review, forthcoming.

49. Hayashi, F. (1982). Tobins marginal q and average q: A neoclassical interpretation. *Econometrica*. **50** (1), 213224.

50. Horenko, I. and Schütte, C. (2008). Likelihood-based estimation of multidimensional Langevin models and its application to biomolecular dynamics. *Multiscale Modlng Simuln*. **7**, 731-773.

51. Hughes, J. P., Guttorp, P., Charles, S. P. (1999). A non-homogeneous hidden Markov model for precipitation occurence. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*. **48** (1), 15-30.

52. Jacobs, O. L. R. (1993). Introduction to Control Theory, 2nd Edition, Oxford University Press.

53. Jarrett, R. G. (1979). A note on the intervals between coal-mining disasters. *Biometrika*. **66**, 191-193.

54. Jeanne, O. and Masson, P. (2000). Currency Crises, Sunspots, and Markov- Switching Regimes. *Journal of International Economics*. **50**, 327-350.

55. Juang, B. H. and Rabiner, L. R. (1991). Hidden Markov models for speech recognition. *Technometrics*. **33**, 251-272.

56. Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Transaction of the ASME. *Journal of Basic Engineering*. **82**, 35-45.

57. Karmarkar, N. (1984). A New Polynomial-Time Algorithm for Linear Programming. *Combinatorica*. **4**, 373-395.

58. Khachian, L. G. (1979). A Polynomial Algorithm in Linear Programming. Dokl. Akad. Nank. SSSR, **244**, 1093-1096. English translation in Soviet Math. Dokl. **20**, 191-194.

59. Koenker, R., Basset, G. Jr. (1978). Regression quantiles. *Econometrica*. **46**, 33-50.

60. Koenker, R., Hallock, K. F. (2001). Quantile Regression. *Journal of Economic Perspectives*. Vol. 15, Number 4, 143-156.

61. Koenker, R., Geling, O. (2001). Reappraising Medfly Longevity: A quantile regression survival analysis. *Journal of the American Statistical Association*. **96**, 459-468.

62. Kim, C., Nelson, C. R. and Startz, C. R. (1998). Testing for mean reversion for heteroscedastic data based on Gibbs-sampling-augmented randomization. *Journal of Empirical Finance*. **5**, 131-154.

63. Koschinski, M., Winkler H. J. and Lang, M. (1995). Segmentation and recognition of symbols within handwritten mathematical expressions. *Acoustics, Speech and Signal Processing*. ICASSP-95. **4**.

64. Leroux, B. and Putterman, M. (1992). Maximum-penalised-likelihood for independent and Markov-dependent mixture models. *Biometrics*. **48**, 545-558.

65. Lewis, R. (1986). Optimal Estimation with an Introduction to Stochastic Control Theory, John Wiley & Sons Inc.

66. Levinson, S. E. (1986). Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition. *Computer, Speech and Language*. **1**, 29-45.

67. Liu, J., Wu, S., Zidek, J. V. (1997). On segmented multivariate regressions. *Statistica Sinica*. **7**, 497-525.

68. Liu, Y. and Bottai, M. (2009). Mixed-effects models for conditional quantiles with longitudinal data. *The International Journal of Biostatistics*. 5.

69. Lumsdaine, R. L., Papell, D. H. (1997). Multiple trend breaks and the unit root hypothesis. *Review of Economics and Statistics*. **79**, 212-218.

70. Machado, J. A. F., Mata, J. (2005). Counterfactual decomposition of changes in wage distributions using quantile regression. *Journal of Applied Econometrics*. **20**(4), 445-465.

71. Manning, C. D. and Schuetze, H. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge: MIT Press.

72. Maybeck, P. S. (1979). Stochastic Models, Estimation and Control, Volume 1, Academic Press Inc.

73. McKinney S. A., Joo, C. and Ha, T. (2006). Analysis of single-molecule FRET trajectories using hidden Markov modeling. *Biophys. J.*. **91**, 1941-1951.

74. Melly, B. (2005). Public-private sector wage differentials in Germany: Evidence from quantile regression. *Empirical Economics*. **30**(2), 505-520.

75. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). Equations of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*. **21** (6), 10871092.

76. Morimune, K., Nakagawa, M. (1997). Unit root tests which allow for multiple breaks. Discussion Paper No. 457, Kyoto Institute of Economic Research, Kyoto University.

77. Nuamah, N. (1986). Pooling Cross Section and Time Series Data, *The Statistician*. **35**, 345351.

78. Pesaran, M. H. and Timmermann, A. G. (2000). A Recursive Modelling Approach to Predicting UK Stock Returns. *Economic Journal*. Royal Economic Society. **110**, 159-191.

79. Rabiner, L. (1989). A tutorial on HMM and selected applications in speech recognition. *Proc. IEEE*. **77**, 257-286.

80. Reich, B. J., Fuentes, M., Dunson, D. B. (2010). Bayesian Spatial Quantile Regression. *Journal of the American Statistical Association*. **106** (493), 6-20.

81. Robert, C. P., Rydén, T. and Titterington, D. M. (2000). Bayesian inference in Hidden Markov models through the reversible jump Markov chain Monte Carlo method. *Journal of the Royal Statistical Society*, B. **62**, 57-75.

82. Romberg, J. K., Choi, H. and Baraniuk, R. (1999). Bayesian tree structured image modelling using Wavelet-Domain Hidden Markov Model. *Proceeding of SPIE*. Denver, CO. **3816**, 31-44.

83. Rossi, A. and Gallo, G. M. (2006). Volatility estimation via Hidden Markov models. *Journal of Empirical Finance*. **13**, 203-230.

84. Royston, P., Altman, D. G. (1994). Regression using fractional polynomials of continuous covariates: parsimonious parametric modelling. *Applied Statistics*. **43**, 429-467.

85. Rydén, T., Teräsvirta, T. and Åsbrink, S. (1998). Stylized facts of daily return series and the Hidden Markov model. *Journal of Applied Econometrics*, **13**, 217-244.

86. Sims, C. and Zha, T. (2004). Were There Switches in U.S.Monetary Policy?. working paper, Princeton University.

87. Smyth, P. (1994a). Detecting novel fault conditions with hidden Markov models and neural networks. *In Pattern Recognition in Practice IV*. Elsevier Science B.V. 525-536.

88. Smyth, P. (1994b). Hidden Markov models for fault detection in dynamic systems. *Pattern Recognition*. **27** 149-164.

89. Scott, S. L. (2002). Bayesian methods for hidden Markov models: Recursive computing in the 21st century. *Journal of the American Statistical Association*. **97**, 337-351.

90. Sorenson, H. W. (1970). Least-Squares Estimation : from Gauss to Kalman, *IEEE Spectrum*. **7**, 63-68.

91. Spiegelhalter, D., Best, N., Carlin, B. and van der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society, Series B*. **64**, 583-639.

92. Swerling, P. (1958). A proposed stagewise differential correction procedure for satellite tracking and prediction, *Tech. Rep.*, p-1292, Rand Corporation.

93. Teh, Y. W., Jordan, M. I., Beal, M. J. and Blei, D. M. (2006). Hierarchical Dirichlet processes. *J. Am. Statist. Ass.*. **101**, 1566-1581.

94. Wood, M. K. and Dantzig, G. B. (1949). Programming of Independent Activities. I. General Discussion. *Econometrica* B. **17**, 193-199.

95. Yau, C., Papaspiliopoulos, O., Roberts, G. O. and Holmes, C. (2011). Bayesian non-parametric hidden Markov models with applications in genomics. *J. R. Statist. Soc.* B. **73**, Part 1, 37-57.

96. Yu, K., Lu, Z., Stander, J. (2003). Quantile regression: applications and current research areas. *The Statistician*. **52**, Part 3, 331-350.

97. Yu, K., Moyeed, R. A. (2001). Bayesian quantile regression. *Statistics and Probability Letters*. **54**, 437-447.

98. Yu, K., Van Kerm, P., Zhang, J. (2005). Bayesian Quantile Regression: An Application to the Wage Distribution in 1990s Britain. *The Indian Journal of Statistics*. **67**, Part 2, 359-377.

99. Zu, H., Li, S. and Yu, K. (2011). Crude oil shocks and stock markets: A panel threshold cointegration approach. *Energy Economics*. **33**, 987- 994.