MAASTRICHT UNIVERSITY

SCHOOL OF BUSINESS AND ECONOMICS

Bayesian Clustering and Variance-Covariance Matrix Modelling using the Dirichlet Process Prior

Author
Tom KENNES

Supervisor
Dr. Nalan BASTÜRK

June 28, 2018

Contents

List of Figures

List of Algorithms

1

Introduction

Intrigued by the usefulness, flexibility and seemingly natural logic of the Bayesian paradigm and the Dirichlet process (see [Teh et al., 2005] for a full account of the Dirichlet process) as well as interested in explaining the correlated behaviour of stock data, I put the Dirichlet process to the test in this thesis. Examples of implementations of the Dirichlet process include [Blei and Lafferty, 2006], [Xing and Girolami, 2007], [Lienou et al., 2010] and [Ramanathan and Wechsler, 2013] but for the purpose of this study Two new models with strong connections with many other models in the literature have been formulated. The first of the two defines a clustering algorithm on the Kalman-filtered time cross sections of stock data. That is, it first applies a Kalman filter to obtain the smoother states or the local level, and consequently it applies a clustering algorithm on the individual snapshots in time similar to [Escobar and West, 1995], using a tendency to cluster that has some similarity with t-distributed stochastic neighbour embedding (t-SNE), see [Maaten and Hinton, 2008], but rather Bayesian. Whereas traditional machine learning clustering algorithms tend to rely on various limiting assumptions, for example k-means on the number of clusters, affords the Dirichlet process greater flexibility, as shown by the examples of implementations. As such, the clustering algorithm can be described as Bayesian. However, in this case, with great flexibility and complexity come great requirements in computational power as well as great visualization issues. It appears to be difficult to obtain useful visuals of a high-dimensional, partially discrete, partially continuous posterior distributions. In response, I chose to implement a greedy maximum a priori (MAP) methodology to convert distributions to probabilities, which appears not to have been proposed before. The problem of intractable MAP has been discussed before, and solutions that focus on adjusting the Gibbs sampling procedure have been introduced in [Raykov et al., 2014] and [Bródka et al., 2012]. However, the solution I propose leaves the Gibbs sampling procedure fully intact, thereby avoiding additional complexity. However, this might be interesting further research. In the greedy MAP, each iteration within the Gibbs sampling procedure forms an allocation. The set of those allocations can then be used to infer a distribution for a certain observation to be clustered with other observations. The greedy MAP is an effort that obtains the MAP of this distribution in a greedy fashion. It clusters observations that have the highest probabilities to be clustered, while adhering to some sort of transitivity. E.g. the fact that observations can only be clustered once and the observations they are clustered with all have to belong to that certain cluster as well. As such obtaining decent visualization is one of the directions where

future improvements could be made. Besides, although the performance of the clustering model can be decent under the right hyperparametrization, connecting those time-snapshots appears to be an open unsolved puzzle. Several methods that do exist are described in this thesis as well as a short discussion of whether performing clustering on subsequences of time series and later connecting these is actually statistically valid. The well-known account of [Keogh and Lin, 2005] actually quite strongly attempts to articulate the opposite.

Whereas the first model deals with the financial data directly, the second model focuses on the variance-covariance matrices. The literature surrounding this topic is vast, consisting of the prominent GARCH approaches, see [Bollerslev, 1986], as well as the stochastic volatilty framework, see [Asai et al., 2006]t, as well as a multitude of more recent directions including, for example, [Noureldin et al., 2012], [Philipov and Glickman, 2006], [Golosnoy et al., 2012], [Chiriac and Voev, 2011], [Bauer and Vorkink, 2006] and [Jin and Maheu, 2009]. It takes monthly variance-covariance matrices as input, and decomposes it into variances and correlations. It then essentially applies the first model to those correlations and models the variance according to an inverse gamma distribution with a commonly-used parametrization. Furthermore, the parameter that influences the number of clusters set up by the Dirichlet process, α, is estimated using a state space model in order to conveniently connect the model with the time dimension. The literature on modelling of variance-covariance models is already vast, but the foundations I build upon are rarely combined as in this thesis. Nevertheless, the model was inspired by the seminal work of [Uhlig, 1994], and foremost by the work of [Jin and Maheu, 2009] and [Windle et al., 2014], which provides a convenient framework. The top layer of the model, termed the Clustering Correlations model, resembles the standard formulation for the modelling of variance-covariance matrices as proposed by the authors of [Windle et al., 2014]. The Bayesian aspect, and the integration of the Dirichlet Process as a prior for the distributions of correlations have, however, barely any resonance with the literature yet. As such, the inner workings of the model are the contribution of this thesis to science. This thesis forms an exploration of future possibilities of those type of models as it attempts to directly take advantage of the underlying tendency of stocks to move together stocks correlations. This makes sense as global financial variables tend to affect financial markets as a whole, as illustrated by 9.1.

The clustering model appears to be quite effective, notwithstanding that the right tuning of the underlying parameters and hyperparameters remains vital. It is difficult to find a rationale for certain values, but the difference in terms of clustering performance might be significant. The performance of the Clustered Correlations algorithm appears difficult to assess. First of all, in order to visualize the underlying distribution, it is necessary to use creative techniques to summarize simulations as well as aggregation. I present means that are aggregated over several variables within the model. It appears that the resulting variance-covariance matrices underestimate the observations. Next to that, the correlation matrices tend to be quite volatile. However, a rule of thumb that can be concluded from the results would be that correlations among portfolios go hand in hand with financial instability but these results also show that this process happens instantaneously rather than gradually.

This thesis is set up in the following way. Chapter 2 involves a short introduction of the Bayesian framework and way of thinking, whereas Chapter 3 then introduces the Dirichlet Processes by means of the four well-known interpretation. Then an account of the Kalman filter is described in chapter 5. The methods introduced in chapters 4 and 5 are then used to construct the 2-step procedure in chapter 6, which implements the clustering model on Kalman-filtered stock data. Chapter 7 then discusses those results, whereas chapter 8 introduces the concept of evolutionary clustering through direct methods as well as indirect methods that focus on clustering correlation and variance covariance matrices. As such, chapter 8 forms the bridge between the two different models and shows the practical inconvenience and unnecessary complexity inherent in direct evolutionary clustering methods. That does not imply that the Clustered Correlations model, as introduced in Chapter 9, is not complex. It is, however, true that the Clustered Correlations model forms an attempt to fabricate a model using the Dirichlet process and the Kalman filter. The results of the Clustered Correlations Model are discussed in Chapter 10 as well as economic and financial interpretations. Finally, Chapter 11 forms a short conclusion on the work that has been done and reflects on possible further scientific endeavours.

2

The Bayesian Framework

2.1 Frequentist vs. Bayesian

put some research in this The 20th century of statistical inference is often said to be dominated by the frequentist reasoning. A somewhat conflicting view, namely Bayesian reasoning, has steadily been gaining ground as well. Philosophically, the difference between the two paradigms is subtle. The mathematical rhetoric underlying the paradigms more illustrates the difference, nevertheless. Due to the historic dominance of the frequentist framework, it is often common for statistics students to be more comfortable with the frequentist paradigm. They are taught to think about a probability as being defined by the frequency of that event based on previous observations, using asymptotics to establish proofs of the paradigm. Inference then can be done based on the idea that if we repeat an experiment a large number of times, each time obtaining some observation for some statistic of interest, we can construct a probability density function and set up a hypothesis test.

Whereas the frequentist approach argues using asymptotics and fixed models and parameters, the Bayesian paradigm takes a different approach. The Bayesian statisticians set up a prior belief of some process and use the data to fine-tune the parameters of this process. In a sense, by stating a prior belief, they inject a bit of subjectivity in the model, although uninformative prior beliefs are quite common as well. As such, probability is interpreted in the Bayesian framework as a reasonable expectation representing a state of knowledge or as a quantification of personal belief. The Bayesian framework therefore commonly allows for a full account of uncertainty. As such, the Bayesian framework models uncertainty implicitly.

Although the new york times [1] might attempt to invigorate the debate over which paradigm is superior once every while, [Kass, 2011] calls out that a healthy degree of pragmatism nowadays is the dominant statistical philosophy. There are clear problems that are suitable for the frequentist approach and similarly for the Bayesian approach. It is thus not important to choose a side or identify oneself as either frequentist or Bayesian. I position himself on a more moderate ground within the frequentist-Bayesian spectrum as well. Of greater importance is the necessity to applicate either one of the two paradigms in the right way and account for uncertainty and

[1] https://www.nytimes.com/2014/09/30/science/the-odds-continually-updated.html?$_r = 1$

assumptions along the way. As such, the frequentist and Bayesian can find common ground in their focus on defining assumptions, the model and its validity rather than the results.

2.2 The Bayesian Paradigm

A typical Bayesian model relies on the input of two probability distributions. First of all, it requires a specification of the distribution of the data given the parameters ($P[y|\theta]$), e.g. the likelihood. Secondly, a prior on those parameters need to be established ($P[\theta]$). Then, using bayes law we can combine the prior and the likelihood as:

$$P[\theta|y] = \frac{P[y|\theta]P[\theta]}{P[y]} \tag{2.1}$$

Whereas the probability of the data, often termed the marginal likelihood or model evidence:

$$P[y] = \int_{\theta} P[y|\theta]P[\theta]d\theta \tag{2.2}$$

This integral can be hard to evaluate, but as it is a normalizing constant, we can often neglect it in practice and establish:

$$P[\theta|y] \propto P[y|\theta]P[\theta] \tag{2.3}$$

Often we require the parameters of the model to follow some distribution described by some hyperparameters. The model can then be easily extended to account for this hierarchy by conditioning and decomposing:

$$P[\theta] = P[\theta|\alpha]P[\alpha] \tag{2.4}$$

Where α denotes the hyperparameter.

Evidently, the choice of prior is of major importance. In practice, it is often more important to fully discuss or establish the applicability of a certain prior than the results. In order to make use of the advantages of the Bayesian framework without injecting quasi-subjective information, often practioners opt for uninformative priors such as a uniform one or Jeffrey's prior, [Jeffreys, 1946].

If initial information is more properly defined and readily available, it is more useful to allow the model to take this into account. Traditionally, in such a setting it was convenient to rely on conjugate priors. The usefulness of a conjugate prior is that the posterior distribution will be in the same distribution family. Then calculation may be expressed in a closed form. In such a setting we are not required to simulate the posterior. In the early days of Bayesian statistics, this specifically was of great relevance. For example, if the data is distributed according to a binomial distribution given parameters N and θ, where N is known and θ is the parameter of interest, a conjugate prior would be beta(α, β). This gives the posterior beta($\alpha + x$,$b + N - x$). Where x depicts the data:

$$\begin{aligned} \text{Likelihood:} \quad & x|\theta \sim bin(N, \theta) \\ \text{Prior:} \quad & \theta \sim beta(\alpha, \beta) \\ \text{Posterior:} \quad & \theta|x \sim beta(\alpha + x, b + N - x) \end{aligned} \tag{2.5}$$

We can model probabilities according to the example above. Then θ is restricted to the interval $[0, 1]$. Picking $\alpha = 2$, $\beta = 3$ and given some dataset x, the resulting beta distributions will be of the form depicted in figure 2.1.

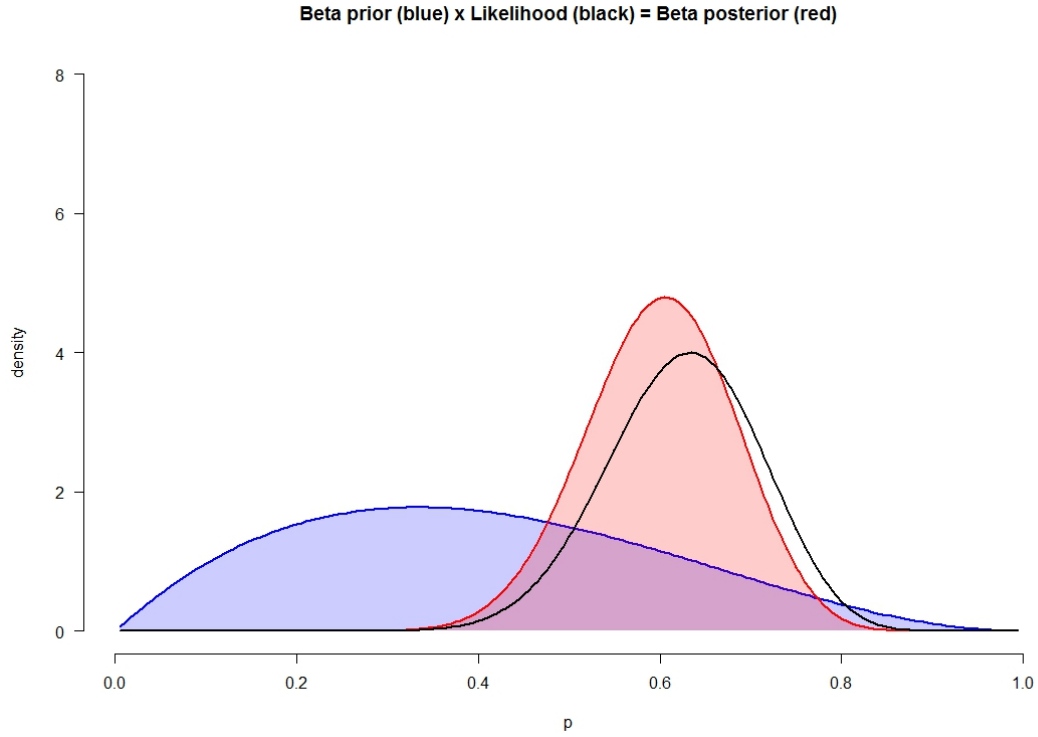

Fig. 2.1: Bayesian Model with Conjugate Beta-prior

Figure 2.1 illustrates the convenience of both Bayesian reasoning and applicability of its methods. Using initial knowledge and stored data, depending on the structure of the model and the hyperparameters, we arrive somewhere in between the our previous knowledge and the observed data. Furthermore, the outcome is a complete characterization of the parameters of interest. Instead of testing for a certain hypothesis and derive a binary answer we obtain probabilities for all posterior values of the parameter. In figure 2.1 a probability was modelled under a prior assigning most probability to the lower values, and combined with the data to obtain a certain degree of proof that the value of the parameter of interest lies between 0.4 and 0.8 with nearly full probability. However, the grand caveat of Bayesian statistics is showcased as well. That is, the choice of prior might heavily influence the posterior values, but in turn requires a thorough account of the choice of the specific prior distribution.

2.3 Obtaining the Posterior

The development of computational power has enabled the use of approximate algorithms to simulate distributions. One simulation class depends on trimming down a certain benchmark that is close to the posterior distribution, as in acceptance/rejectance and importance sampling. In acceptance/rejectance sampling, draws are generated by the benchmark and accepted depending on the ratio between the probability assigned to the sample by the benchmark and the

probability obtained from the posterior pdf. Importance sampling in contrast retains all the draws, but approximately weights the draws. More sophisticated algorithms use some form of Markov chain Monte Carlo (MCMC) methodology where the elements of the markov chain are distributions and monte carlo the procedure that defines the steps within that Markov chain such that the MCMC as a whole converges to the posterior distribution of interest. Two widely algorithms are the Gibbs Sampler and the Metropolis Hastings. The Gibbs sampler iteratively assigns new values to parameters of interest conditioning on the other parameters being fixed. As such, it relies on marginal distributions and these being obtainable from the posterior distribution. Metropolis Hastings in contrast uses a candidate density to describe jumps between Markov states and accepts those jumps based on a predefined condition. See for example [Geweke, 2005], [Hartig et al., 2011] and figure 2.2 below, retrieved from [Hartig et al., 2011], as well.

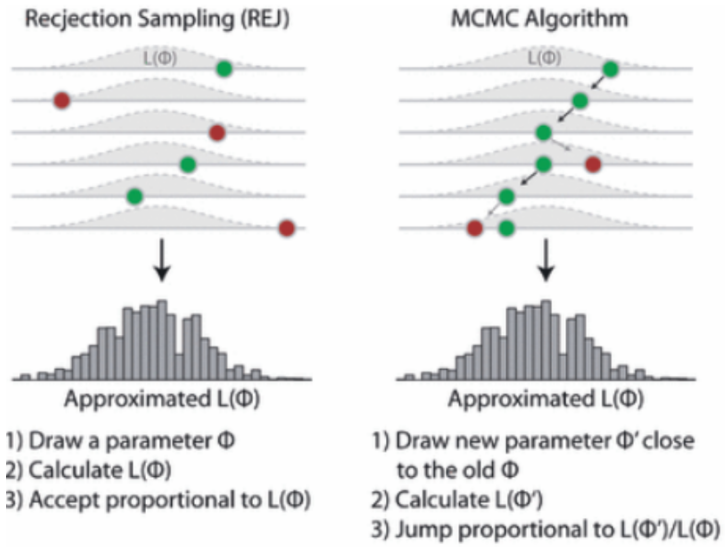

Fig. 2.2: Posterior Simulation Methodologies

2.4 Gibbs Sampler

Chapter 4 introduces the Dirichlet process, which is one of the most important building blocks of this thesis as well. As the estimation of the Dirichlet process makes use of a Gibbs sampler, this section provides a more thorough explanation of that sampler.

Denote the vector $\boldsymbol{\theta} = (\theta_1, ..., \theta_n)$ as a vector of parameters of the model, a state variable of the Markov chain. Within the Gibbs algorithm, an elementary update then stochastically assigns a new value on one of the x_i conditioned on the others. These one-dimensional conditional distributions are called full conditionals and are derived from the posterior distribution. The Gibbs update is then characterized by:
The Gibbs sampler is not unique in Bayesian Statistics. It was coined earlier in statistical mechanics where it was called the "heat bath" algorithm. As noted by [Van Laarhoven and Aarts, 1987] "In condensed matter physics, annealing denotes a process in which a solid in a heat bath is

Algorithm 1 Gibbs Sampler

Require: θ_0
1: **for** $\forall t \in [1, T]$ **do**
2: Update $\theta_{1,t}$ according to $P[\theta_{1,t}|\theta_{2,t-1}, ..., \theta_{n,t-1}]$
3: Update $\theta_{2,t}$ according to $P[\theta_{2,t}|\theta_{1,t}, \theta_{3,t-1}..., \theta_{n,t-1}]$
4: ...
5: Update $\theta_{n,t}$ according to $P[\theta_{n,t}|\theta_{1,t}, \theta_{2,t}, ..., \theta_{n-1,t}]$

heated up by increasing the temperature of the heat bath to the maximum value at which all particles of the solid randomly arrange themselves in the liquid phase, following by cooling through slowly lowering the temperature of the heat bath". In the cooling process the bath and the solid iteratively reach new thermal equilibria. The probability of being in a state characterized by a certain energy given a certain temperature value T is characterized by a Boltzman Distribution. The path of thermal equilibria comes down to solving a large stochastic combinatorial optimization problem due to the system of particles and states in the liquid and the solid. It turns out that this can be accurately simulated using the Gibbs sampling methodology. This analogy is also were the term simulated annealing comes from and thus explains the relationship between Gibbs sampling and simulated annealing.

In order to show the Gibbs sampler in practice, follow the example of [Gelfand and Smith, 1990] of a simple variance component model:
Let the observed data $y_{i,j}$ be distributed with $i = 1, ..., K$ and $j = 1, ..., K$ as:

$$\begin{aligned} y_{ij} &\sim Normal\big(\theta_i, \frac{1}{\lambda_\epsilon}\big) \\ \theta_i &\sim Normal\big(\mu, \frac{1}{\lambda_\theta}\big) \end{aligned} \tag{2.6}$$

Note that the parameter μ, λ_θ and λ_ϵ are treated as quantities with prior distributed instead as unknown constants in frequentist paradigm. Then, defining conjugate priors with known hyperparameters:

$$\begin{aligned} \mu &\sim Normal\big(\mu_0, \frac{1}{\lambda_0}\big) \\ \lambda_\theta &\sim Gamma(a_1, b_1) \\ \lambda_\epsilon &\sim Gamma(a_2, b_2) \end{aligned} \tag{2.7}$$

Then in order to able to apply the Gibbs sampler it is necessary to set up the joint distribution and derive the full conditional probability distributions, if it is possible to do so. The joint is:

$$h(\theta_1, ..., \theta_k, \mu, \lambda_\theta, \lambda_\epsilon) = \lambda_\epsilon^{-\frac{\lambda_\epsilon}{2}\sum_{ij}(y_{ij}-\theta_i)^2}\lambda_\theta^{K/2} e^{-\frac{\lambda_\theta}{2}\sum_i(\theta_i-\mu)^2} e^{-\frac{\lambda_\epsilon}{2}\sum_{ij}(y_{ij}-\theta_i)^2}\lambda_\theta^{a_1-1}e^{-b_1\lambda_\theta}\lambda_\epsilon^{a_2-1}e^{-b_2\lambda_\epsilon} \tag{2.8}$$

From where it is possible to derive the conditional distributions as:

$$\begin{aligned}
\lambda_\theta|\boldsymbol{\theta} &\sim Gamma(a_1 + K/2, b_1 + \frac{1}{2}\sum_i(\theta_i - \mu)^2) \\
\lambda_\epsilon|\boldsymbol{\theta} &\sim Gamma(a_2 + JK/2, b_2 + \frac{1}{2}\sum_{ij}(y_{ij} - \theta_i)^2) \\
\mu|\boldsymbol{\theta} &\sim Normal(\frac{\lambda_0\mu_0 + K\lambda_\theta\bar{\theta}}{\lambda_0 + K\lambda_\theta}, \frac{1}{\lambda_0 + K\lambda_\theta}) \\
\theta_i|\boldsymbol{\theta} &\sim Normal(\frac{\lambda_\theta\mu + J\lambda_\epsilon\bar{y}_i}{\lambda_\theta + J\lambda_\epsilon}, \frac{1}{\lambda_\theta + J\lambda_\epsilon})
\end{aligned} \tag{2.9}$$

The rationale can easily be extended to multivariate cases where variables are update in groups. This gives rise to the block Gibbs sampler, because one updates a block from their joint conditional distribution given the other variables.

The beauty of Gibbs sampling is that one can sample the joint distribution knowing only full conditionals. But as mentioned before, if these are not readily available, then it is greatest weakness as well. In such cases, Gibbs should be completely disregarded as a method to simulate the posterior. It is interesting to note that following the publication of [Gelfand and Smith, 1990] the Gibbs sampling procedure was met with excitement by the Bayesian community. But only a few years later it was recognized that the Gibbs sampler gained prominence by accident. It appeared that the Gibbs sampler was overhyped. Relying solely on the Gibbs sampler and thereby excluding other MCMC methods is nowadays considered bad practice and ignorance. Where it not only because Gibbs sampling is in fact a special case of the Metropolis Hastings algorithm where the transition operators are chosen such that the acceptance probability is 1.

3

The Dirichlet Process

Latent Dirichlet Allocation (LDA) algorithms are commonly used to create an unobserved identity for certain items of interest using merely several useful variables. In the neural-nets nomenclature and literature these variables are often referred to as features. LDA's are for example used to build a identity of series of documents in dynamic topic modelling as in [Blei and Lafferty, 2006], to build the identity of users to detect fraudulent behaviour in telecommunications ([Xing and Girolami, 2007]), semantically annotate spatial data such as satellite images ([Lienou et al., 2010]) and more recently even phishing ([Ramanathan and Wechsler, 2013]) Although computer science and machine computing sciences have been quick to catch up, Bayesian econometrics and statistics been developing its properties as a prior for mixture modelling and methods of computation well ahead. Among others, see for example [Antoniak, 1974].

A common problem of traditional clustering methods are the fact that they often assume some fixed number as the number of clusters to be obtained. For example, k-means algorithms generally rely on a predetermined number of clusters, k. Values of k are then chosen based on information criteria like in [Hansen and Yu, 2001] or the commonly used Bayesian and Akaike information criteria. Another approach, following [Tibshirani et al., 2001], is relying in gap statistics. Here I however approach the problem from a Bayesian perspective and more explicitly let the data decide how many clusters there exist. As such, I follow the idea introduced by [Ferguson, 1973] and [Rasmussen, 2000] in using a non-parametric Dirichlet process to account implicitly for the uncertainty surrounding the number of clusters. Chapter 6 shows an implementation of these ideas relying on the implementation described in [Escobar and West, 1995].

In order to avoid going to much in detail from the start, this chapter introduces the concept of the Dirichlet process. Following [Teh et al., 2005], it presents 3 different perspectives on the Dirichlet process, that provide a deeper understanding about the ideas and the distributions for the reader.

3.1 Stick Breaking Construction

It was established by [Ferguson, 1973], and made explicit by [Sethuraman, 1994], that measures drawn from a Dirichlet process are discrete with probability one. We can define a stick-breaking

process as independent sequences of random variables $(\pi'_k)_{k=1}^{\infty}$ and $(\theta_k)_{k=1}^{\infty}$:

$$\pi'_k|\alpha_0, G_0 \sim Beta(1, \alpha_0) \qquad \theta|\alpha_0, G_0 \sim G_0 \tag{3.1}$$

Defining the random measure G as:

$$\pi_k = \pi'_k \prod_{l=1}^{k-1}(1 - \pi'_l) \qquad G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k} \tag{3.2}$$

Where δ_{θ_k} is a probability measure concentrated at θ. According to [Sethuraman, 1994] G then is a random probability measure distributed according to a Dirichlet Process: DP(α_0, G_0).

From the stick-breaking analog we can deduce that $\sum_{k=1}^{\infty} \pi_k = 1$ with probability one, and therefore π is a random measure on the positive integers. Furthermore, note that G, a random measure, is again distributed above according to some random measure random probability measure $\mathrm{DP}(\alpha_0, G_0)$, implying that the Dirichlet process is a distribution over distributions. This is a convenient characteristic with respect to clustering, because the distributions of cluster-locations and observation-allocations highly depend on the number of clusters. It requires thus a distribution over the number of clusters and hence a distribution over distributions.

3.2 The Chinese Restaurant Process

The polya urn scheme introduced by [Blackwell and MacQueen, 1973] shows an intriguing interpretation of the Dirichlet process as well. Let A denote the initial number of urns and denote $\alpha_i \ \forall i \in [0, k]$ the number of balls in each urn. The polya urn scheme then assigns incoming balls to one of the existing urns with a probability measure depending on the number of balls in each urn, or to a new urn with some positive probability.

A different interpretation of the Polya Urn scheme is given by [Aldous, 1985] and known as the chinese restaurant process. Let θ_i be infinitely large round tables in a infinitely large Chinese restaurant. Customers ϕ_i arrive at the restaurant and take seats according to the current seating of $\phi_1, ..., \phi_{i-1}$. We can then express the distribution as:

$$\phi_i|\phi_1, ..., \phi_{i-1}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{\eta_k}{i - 1 + \alpha_0} \delta_{\theta_k} + \frac{\alpha_0}{i - 1 + \alpha_0} G_0 \tag{3.3}$$

Where the second term denotes the probability of customer ϕ_i taking a seat on a new table and η_k the number of customers on table θ_k.

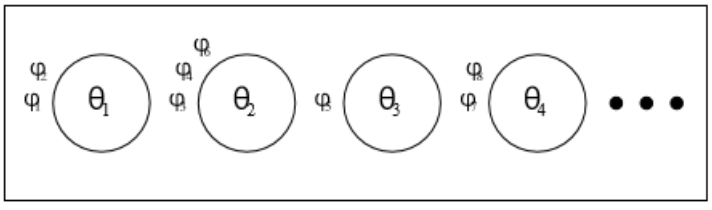

Fig. 3.1: Depiction of the Chinese Restaurant Process from [Teh et al., 2005]

Figure 3.1 illustrates both the applicability to clustering of the Dirichlet process as well as a possible problem; generally the polya urn/chinese restaurant schemes imply a certain attractability of larger collections (urns or tables). The richer get rich property. Depending on G_0 and α_0, it is a common problem that larger tables tend to get larger faster than smaller tables. As such, there is a tendency of customers to sit on the larger tables. Similarly for clustering, larger clusters tend to have a higher attractibility the smaller clusters and thus make the formation of clustering whole datasets into 1 cluster more likely. Trivially, both the result of 1 cluster and the number of clusters equal to the number of observations are not particularly useful.

3.3 Mixture Models

A common application of the Dirichlet process is a nonparametric prior distribution on the components of a mixture model as in [Antoniak, 1974]. Let L denote the number of mixture components and $\boldsymbol{\pi} = (\pi_1, ..., \pi_L)$ the mixing proportions. We place a Dirichlet prior on $\boldsymbol{\pi}$ with symmetric parameters $(\alpha_0/L, ..., \alpha_0/L)$. Let θ_k be the parameter vector associated with mixture component k and let θ_k have prior distribution G_0. Drawing an observation x_i from the mixture model involves picking a specific mixture component with probability given by the mixing proportions. Let z_i denote that component. Following [Teh et al., 2005] we thus have the model:

$$\begin{aligned} \boldsymbol{\pi}|\alpha_0 \sim Dir(\alpha_0/L, ..., \alpha_0/L) \qquad z_i|\boldsymbol{\pi} \sim \boldsymbol{\pi} \\ \theta_k|G_0 \sim G_0 \qquad x_i|z_i, (\theta_k)_{k=1}^L \sim F(\theta_{z_i}) \end{aligned} \tag{3.4}$$

For clustering, the mixture model is applicable as well. Denote $\pi_i \in [0, 1]$ constrained to $\pi_i \in \mathbb{Z}$ and $\sum_{i=1}^L \pi_1 = 1$. Then automatically the mixture clusters the observations. Analogously, we can relax the integer assumption and continue assigning observations to clusters based on probabilities or fuzzy logic clusters. See for example [Yang, 1993] for a survey on fuzzy clustering. In our case, the fuzzy clustering would vastly increase the flexibility and reduce the interpretability. I would also shift the focus to the number of clusters instead of the clustering itself. Furthermore, fuzzy clustering relies on slightly different probability interpretations which are not directly compatible with the Dirichlet process nor Bayesian methodology of this thesis. All in all, fuzzy clustering could be interesting, but would complicate the model and underlying mathematics too much.

3.4 The Dirichlet Process

Suppose an observation y_i follows a given distribution F with parameters ϕ_i. Then a Dirichlet process is defined by the following model:

$$\begin{aligned} y_i|\phi_i &\sim F(\phi_i) \\ \phi_i|G &\sim G \\ G &\sim DP(\alpha, G_0) \end{aligned} \tag{3.5}$$

To obtain a representation of the prior distribution of the ϕ_i in terms of successive conditional distributions, integrate over G:

$$\phi_i|\phi_1,...,\phi_{i-1} \sim \frac{1}{i-1+\alpha}\sum_{j=1}^{i-1}\delta(\phi_j) + \frac{\alpha}{i-1+\alpha}G_0 \tag{3.6}$$

Where $\delta(\phi)$ is the distribution concentrated at the single point ϕ. G_0 is a base distribution with the same support as the random probability measure G, whereas α can be interpreted as a positive parameter indicating the precision of the base distribution.

Using the stick breaking process we can generate samples from the Dirichlet process easily. Iteratively we break a part β_i of the remaining the stick of length w_{i-1}. Then:

$$w_i = \beta_i \prod_{j=1}^{i-1}(1-\beta_j) \tag{3.7}$$

Where $\beta_1 \sim Beta(1,\alpha)$. If we continue iteratively breaking the stick in this manner and if $w_i \sim G_0$ then:

$$P = \sum_{i=1}^{\infty} w_i \delta_{w_i} \sim DP(\alpha, G_0) \tag{3.8}$$

For a simulation, take G_0 standard normal and for $\alpha = (10, 50)$. Then notice that the Dirichlet Process converges to the base distribution G_0 as $\alpha \to \infty$.

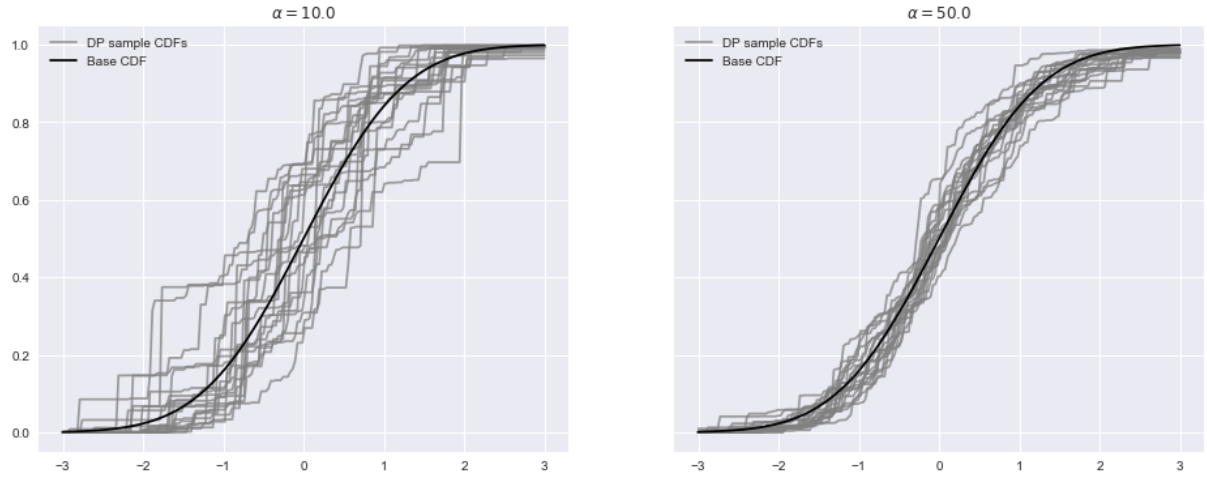

Fig. 3.2: Dirichlet Process CDF in comparison to the Standard Normal CDF

With respect to clustering, a larger α indicates that the model will tend towards more clusters. In terms of the chinese restaurant process, it will be more likely that a newly arrived customer will take a seat at a empty table.

As we are interested in clustering, our goal is to find values for ϕ_i such that a clustering is established. However, we do not particularly know what form a cluster should have nor whether the data is a priori separable. These issues are adressed in next the chapters.

4

Local Level Model and Kalman Filter

The Kalman filter is an estimation technique that relies on an assumption of unavailability of the true value and attemps to estimate the true value by a proxy that we can observe. As such, it uses a measurement to gain insight about the unobservable state, given a certain model. Depending on the choice of the parameters, it filters out a part of the volatility of the measurement and obtains a more smoothed curve, as it decomposes the data in a local level plus additional noise. The unobserved states can be seen as a stochastic trend underlying the behaviour that is witnessed. As I make use of the Fama and French industry portfolios, I am directly using the fluctations of large group of stocks over time to gain an insight in the clustering over time. Generally, it is likely that stock data follow a volatile path characterized by volatile periods. In order to stabilize that aspect of the data, a Kalman filter is a good option as it trims down extreme stochastic behaviour.

The reason to believe that there exists a certain degree of noise in the data is multitude. The most convincing argument is the fact that it is impossible to quantify the changes in stock value as calculated by the investor in terms of actual intrinsic company value. But that makes sense as well. Stocks have more usage for investors than simply as investment vehicles. Among others, there is simple speculation, risk hedging, portfolio construction and rippling effects within the market only exacerbate the effect. Therefore, in order to avoid clustering stochastic behaviour instead of underlying dynamics as described by the local levels, I apply the Kalman filter.

4.1 Kalman Filter

Denote y_t as the observed variable at time t and a_t as the state variable at time t. Then the local level model is specified as:

$$\begin{aligned} y_t &= Z_t a_t + e_t \\ a_{t+1} &= T_t a_t + R_t w_t \end{aligned} \tag{4.1}$$

With $e_t \sim N(0, H_t)$ and $w_t \sim N(0, V_t)$. It is necessary to specify the parameters Z_t, T_t, R_t, H_t, V_t beforehand.

Using these parameters we can then iteratively estimate values one step ahead as more information enters the system. Let the observations up to time t be known and let the states be

constructed up to time t using those observations as well. E.g. two series, $(y_i)_{i=1}^t$ and $(a_i)_{i=1}^t$ are known. Then the Kalman filter can be applied in a two-step procedure. The first step consists of prediction (a_{t+1}, v_{t+1}), where $v_{t+1} = y_{t+1} - Za_{t+1}$, e.g. the realized error. The second step then consists of an updating step. Denote $\hat{a}_t$ as the mean of the estimated state and $\hat{P}_t$ as the estimated variance, P_t, at time t. The algorithm then can be summarized as follows:

Algorithm 2 Kalman Filter Update Step

Require: $(y_i)_{i=1}^t$ and $(a_i)_{i=1}^t$

Prediction:

$\hat{a}_{t+1} = Ta_t$

$\hat{P}_{t+1} = T'P_tT + R'VR$

$v_{t+1} = y_{t+1} - Z\hat{a}_{t+1}$

$F_{t+1} = Z'\hat{P}_{t+1}Z + H$

Updating:

$K_{t+1} = \hat{P}_{t+1}ZF_{t+1}^{-1}$

$a_{t+1} = \hat{a}_t + K_{t+1}v_{t+1}$

$P_{t+1} = \hat{P}_{t+1} - K_{t+1}Z\hat{P}_{t+1}$

Where F_t denotes the estimated variance of observation error at time t and K_t the Kalman gain. The Kalman gain is the relative weight given to the measurements and current state estimate. With a high gain, the filter places more weight on the most recent measurements and thus follows them more closely whereas a lower gain implies that the filter follows the predictions more closely. This implies that the lower the Kalman gain, the less responsive the state will be to noise in the observations. Thus, depending on how accurate the observations are, the parameters should be adjusted accordingly.

Note that the Kalman filter requires initial values for a and P. The literature on this is broad, but if we fix the parameters over time, as I already did in algorithm 2, P converges to a stable value. See also figure 4.2. Furthermore, after a burn-in phase, a will be "stochastically stable" as well.

In the end, the Kalman filter thus results in a serie of smoothed states. For example, taking the observation of the Fama and French portfolio of the agriculture industry and setting the variables such that $Z = T = R = V = P_0 = 1$, $a_0 = 0$ and $H = 10$, meaning that the observation is much noisier than the state, the result of applying the Kalman filter is shown in figures 4.1 and 4.2. Vica versa, if the variance of the state, V, is much higher than the variable of observation, H, and keeping all the parameters fixed again, the state is much noisier than the observation. Meaning that we can actually do best by taking the observation as the state. With $V = 10$, $H = 1$, ceteris paribus, one obtains the results as depicted in figures 4.3 and 4.4.

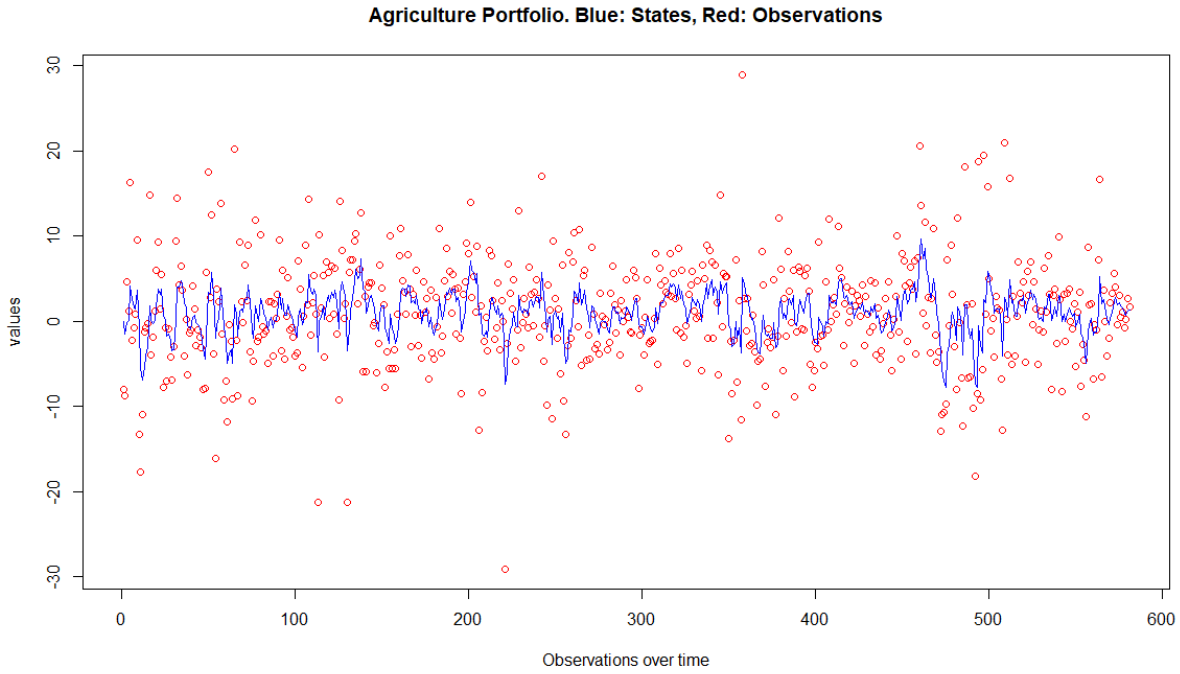

Fig. 4.1: Example of Kalman Filter on the Agriculture Portfolio of Fama and French

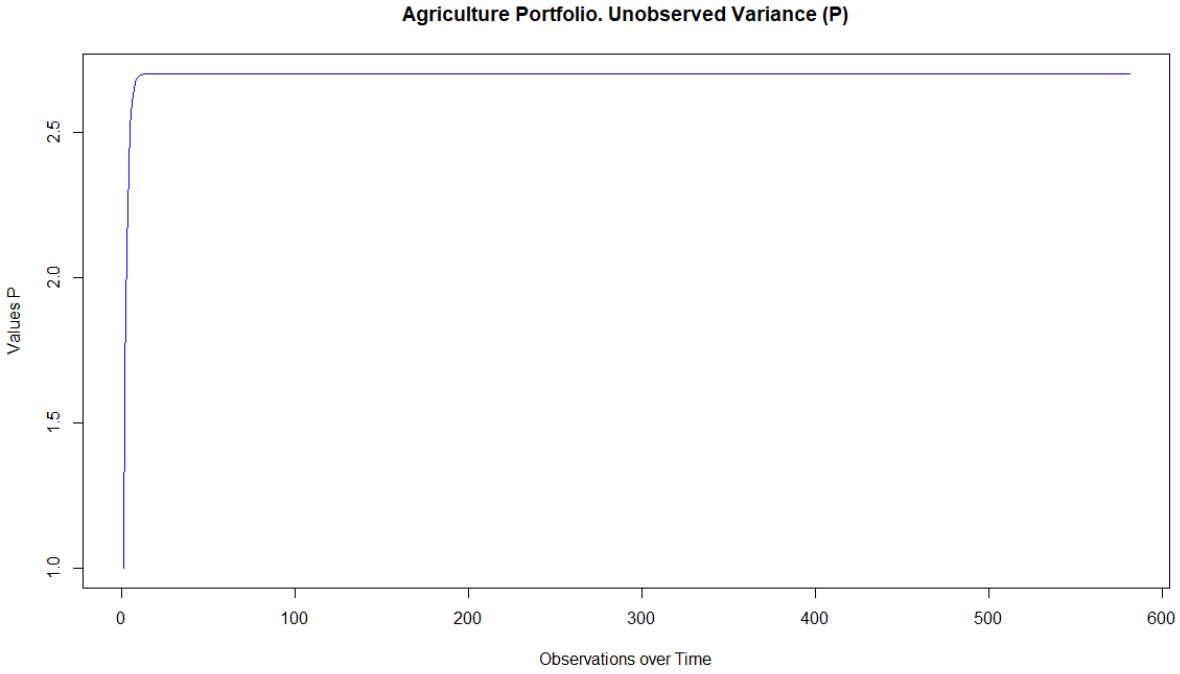

Fig. 4.2: Converging Unobserved Variance of the State

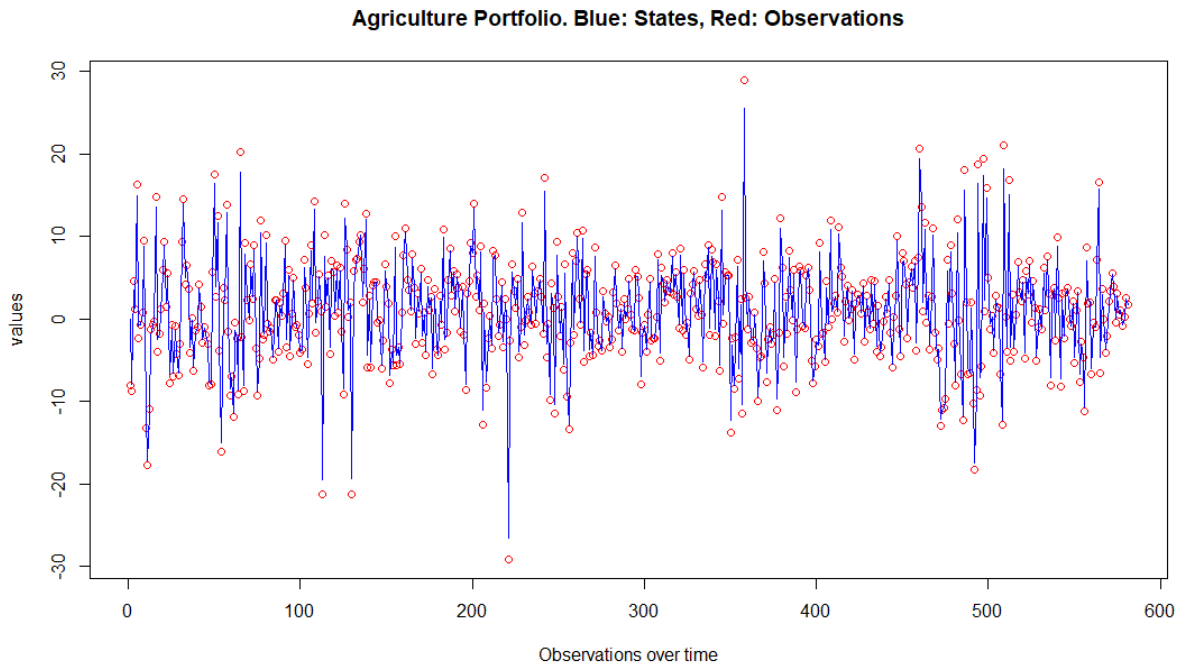

Fig. 4.3: Example of Kalman Filter on the Agriculture Portfolio with Volatile States

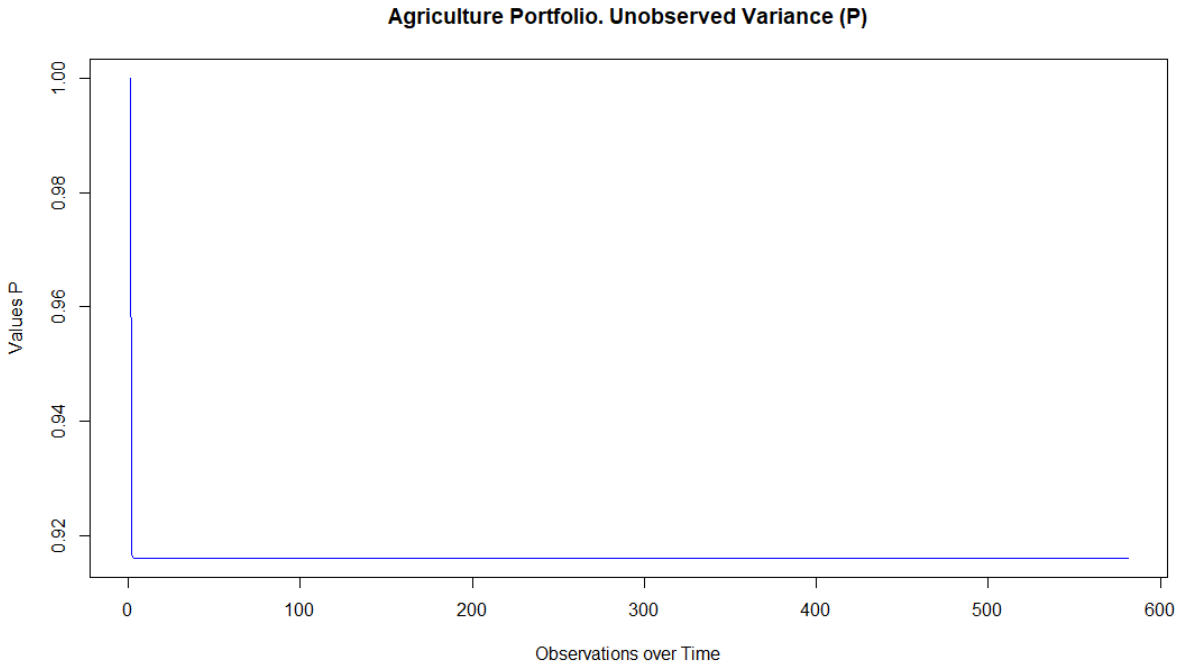

Fig. 4.4: Converging Unobserved Variance of the State, Volatile States

5

The 2-step Procedure

5.1 Escobar and West, 1995

In order to formulate a Bayesian model for clustering, it is necessary to overcome the following two issues:

1. Where are my clusters located and how do they look like?
2. How are the observations allocated over the clusters?

In order for the first one to fit within the Bayesian framework as introduced in Chapter 1, it is convenient to approach it from a probabilistic point of view. That is, we can view the clustering as a mixture model of normal distributions, where each individual normal distribution of the mixture denotes a cluster. Then, the mean corresponds to the center of the cluster and the variance to the size of the cluster. In higher dimensions, the covariances then correspond to the symmetry of the boundary of the cluster. I write boundary, but actually there is no physical boundary as with ordinary clustering. The further away from the center of a cluster, the likelihood of belonging to that normal, that cluster, reduces. This is not so much of a drawback and it can be related to the second part fairly straightforward. For the second part, I make use of the Dirichlet Process as introduced in Chapter 3. In the case where a certain observation is significantly far from a center of a certain cluster, the Dirichlet Process is able to put a low probability of belonging to that cluster and a higher probability of belonging to a closer cluster or perhaps even starting a new cluster with a center nearby that specific observation.

I use the normal-inverse-gamma prior to describe clusters. If the center of that cluster lies further away from the observations of the clusters, it becomes more likely that observations tend to break of from the cluster and join other nearby clusters. As such, those clusters that remain have achieved prominence within the set of observations and likely have their center close to several observations. The variance I either picked by hand or allowed to be such that the variance describes a bound of the cluster. Depending on the distribution of the observations, it might be more likely to actually fix the variance. If the clusters are nicely separable, letting the observations decide the circumference of the clusters makes sense. However, if they appear to be hardly separable it makes more sense to force the cluster size to be fixed in order to prevent the clusters from growing significantly large and thereby encompassing a larger than reasonable

part of the observations. See also the results in Chapter 6.

Forcing the variance, or the probabilistic size, of the cluster to be fixed is in essence not an objective thing to do. However, it can be that certain outliers obscure the difference between other observations. If there is one observation that is significantly far away from the others, it will appear as if the others are fairly close. But if the outlier is deleted from the dataset, the relative difference between the observations becomes less apparent and clustering might be more succesful. In the other case, tweaking of the underlying parameters, such as the parameters that define a cluster, might be a solution. Both the presence of outliers and hardly separable data might have profound effects on the effectivity of clustering the data and in turn may lead to speculation about the robustness of many clustering algorithms.

Firstly this section introduces the methodology as explained in [Escobar and West, 1995], sticking to the naming of the parameters as well. I follow it closely in terms of implementation, but deviate slightly here and there, which this section explains later.

Given an assignment $\pi_{-i'}$ of observations $-i$ to clusters, the assignment of observation i, denoted as π_i follows a Dirichlet Process prior as introduced in previous chapters. Which yields the conditional posterior:

$$\pi_i|\pi_{i-1} \sim \alpha q_0 G_i(\pi_i) + \sum_{j=1, j\neq i}^{n} q_j \delta_{\pi_j}(\pi_i) \tag{5.1}$$

This can be denoted as DP($\alpha, q_0 G_0$). $G_i(\pi_i)$ denotes the distribution of a certain cluster. E.g. it is the prior distribution that represents the believe we initially have about the structure of a cluster in terms of location and size. As such, given a set of observation that are clustered together, e.g. forming a cluster named i, let the prior of the cluster be defined as a bivariate normal-inverse gamma distribution:

$$\begin{aligned} V_i^{-1} &\sim gamma\Big(\frac{1+s}{2}, S + \frac{(y_i - m)^2}{1+\tau}\Big) \\ \mu_i|V_i &\sim N\Big(\frac{m+\tau y_i}{1+\tau}, \frac{\tau}{1+\tau} V_i\Big) \end{aligned} \tag{5.2}$$

The parameters in this prior have to be specified beforehand. τ denotes a measure of believe we have in the accuracy of our prior. For $\tau \to 0$:

$$\begin{aligned} lim_{\tau\to 0} V_i^{-1} &\sim gamma\Big(\frac{1+s}{2}, S + (y_i - m)^2\Big) \\ lim_{\tau\to 0}\ \mu_i|V_i &\sim N(m, 0) \end{aligned} \tag{5.3}$$

And for $\tau \to \infty$:

$$\begin{aligned} lim_{\tau\to\infty}\ V_i^{-1} &\sim gamma\Big(\frac{1+s}{2}, S\Big) \\ lim_{\tau\to\infty}\ \mu_i|V_i &\sim N(y_i, V_i) \end{aligned} \tag{5.4}$$

Thus, a lower τ states that our prior is very trustworthy, whereas a higher τ denotes that we have more confidence in the data. As such the relative effects of prior and data can be mediated.

Next to that the other parameters represent the usual input for the NIG prior. That is with s and S as location and scale parameter, respectively, of the inverse gamma distribution characterizing the size of the cluster. m is a prior value describing the location of the cluster. Due the symmetry of the normal distribution m represents also a believe on the value of the center of the cluster.

The weights (q_0, q_j) are defined as:

$$\begin{aligned} q_0(s, S, m, y_i, \tau) &\propto \boldsymbol{c}(s)\Big[\frac{1+(y_i-m)^2}{sM}\Big]^{-(1+s)/2} M^{-1/2} \\ q_j(y_i, \mu_j, V_j) &\propto exp\Big(-(y_i-\mu_j)^2/2V_j\Big)\Big(2V_j\Big)^{-1/2} \end{aligned} \tag{5.5}$$

Where:

$$\begin{aligned} \boldsymbol{c}(s) &= \frac{\Gamma\Big(\frac{1+s}{2}\Big)}{\Gamma(s/2)} s^{-1/2} \\ M &= (1+\tau)S/s \end{aligned} \tag{5.6}$$

The question still remains how to set up $\delta_{\pi_j}(\pi_i)$. In [Escobar and West, 1995] it is stated that it denotes a point mass at $\pi_i = \pi_j$. Although not concrete, from the Dirichlet distribution it becomes apparent that it should be interpreted as a probability distribution over the other clusters. This is most congruent with the first part of the equation as the first part denotes the probability that π_j will start the existence of a new cluster. Actually, another representation in [Escobar and West, 1995] provides more clarity:

$$\pi_{n+1}|\boldsymbol{\pi} \sim \frac{\alpha}{\alpha+n} T_s(m, M) + \frac{1}{\alpha+n}\sum_{j=1}^{k} n_j N(\mu_j^*, V_j^*) \tag{5.7}$$

Where T_s is the student T distribution and the second part of the equation denotes the mixture of existing clusters (μ_j^*, V_j^*). Note how n_j explicitly brings the "rich-get-richer"-characteristic of Dirichlet Processes into the model.

Using the model given by [Escobar and West, 1995] as a guiding example, I can now properly introduce the implementation of the model. Because the goal eventually is to use these models to analyse the clustering behaviour of stock data, as argued before it seems intuitive and logical to focus on the local levels. In order to do so, there are essentially 2 ways we could go. The first consists of a 2-step procedure, which is also the one this section turns to.

5.2 Multivariate Extension

A common attribute of stock data is the strong autocorrelation. As I aim to cluster the data per time-window in order to gain insight in the driving dynamics, it is necessary to open up the possibility to use a multivariate extension of the model of [Escobar and West, 1995]. A side of theoretical issues, in practice it appears often hard to cluster in one dimension. It is for example common in support vector machines to make use of kernels that decompose the data into many dimensions. As such, the data gains separability and clustering algorithms perform better. In a

similar fashion one can add dimensions to the data to allow for that separability. If the data is characterized by a certain inseparability, clusters, especially if we allow the data to decide how many clusters there are, tend to merge into one or more giant clusters. The same observation was made by [Escobar, 1994], which coined "clumpiness" to describe separability and sparsity as its antagonist. The problem can be illustrated by the following example:

$$\begin{aligned} x_i &\sim \begin{cases} U[0.2, 0.3] \text{ if i} \leq 100 \\ U[0, 1] \text{ if } 100 < i < 600 \\ U[0.7, 0.8] \text{ if i} \geq 600 \end{cases} \\ y_i &\sim N(x_i, 0.25) \end{aligned} \tag{5.8}$$

Figure 5.1 shows the results. If the data is characterized by a uniform density over domain, it is hard to depict different clusters by hand. In this case the data should be clustered together evidently, but a practitioner seeking to cluster a dataset similar as in figure 5.1 will generally not have the DGP readily available however, and might thus attempt to define multiple clusters in the figure. In the simulation example, we know beforehand that the data is simulated from the

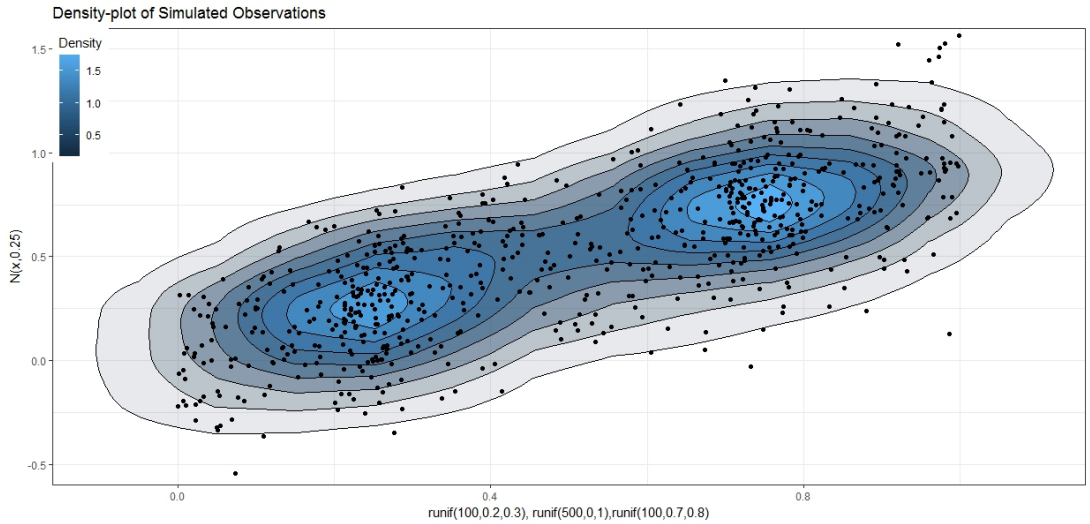

Fig. 5.1: Example of Hardly Separable Data

same source, however, if we map the observations to a density estimation, it becomes apparent that there are two equally high peaks. But there hardly is a valley between those peaks, e.g. the density is still quite high between them. In this case it would still be likely that whole set would be fitted to one cluster, but if the transition between the peaks would more of a valley, the question would become harder as it is exactly at those density-contours that the problem of separability becomes an issue. In essence, the question thus comes down to; how to qualify the interpeak observations.

Mathematically, the multivariate extension is readily available. However, extending the NIG generally implies that V_i obtains a diagonal structure. This in turn implies that the shape of the cluster will not tilt in terms of the axis of the individual dimensions, as existence of correlation between them would allow a tilt. However, the t-distribution as in [Escobar and West, 1995] has

to be altered slightly. [Kotz and Nadarajah, 2004] denote the pdf of a p-variate t distribution with degrees of freedom v, mean vector $\boldsymbol{\mu}$ and correlation matrix $\boldsymbol{\Sigma}$:

$$f(\boldsymbol{x}) = \frac{\Gamma(\frac{v+p}{2})}{(\pi v)^p \Gamma(v/2)|\boldsymbol{\Sigma}|^{1/2}} \Big[1 + \frac{1}{v}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\Big]^{-\frac{v+p}{2}} \tag{5.9}$$

Where $p = 1$ leads back to the univariate case as in [Escobar and West, 1995].

5.3 2-step Model

In the first step of the model, the local level staes are obtained using a Kalman filter. Then the states are used as input for the Dirichlet process with the NIG prior.

5.3.1 Kalman Filter Specification

Following equation 4.1

5.3.2 Clustering Implementation

Denote $z_{i'}$ as the cluster to which observation i' is assigned, e.g. a label (Similar to π in the model of [Escobar and West, 1995]). Denote the total number of clusters as K. Let $\boldsymbol{\theta}$ be the parameters necessary for founding new clusters according to Dirichlet-Process-base-function G_0. Denote $\boldsymbol{y}_k$ as a vector of observations that are in a certain cluster with label k and $(\boldsymbol{\mu}_k, \boldsymbol{V}_k)$ as the information those observation bring with regard to the cluster. In appendix A2 a derivation of the Dirichlet process is presented. The model I use is a modification of that. The Dirichlet process as in appendix A2 is given as:

$$\begin{aligned} P[\boldsymbol{\theta}|\boldsymbol{\alpha}] &\sim Dirichlet(\alpha) \\ P[z_{i'}|\boldsymbol{\theta}] &\sim categorical(\boldsymbol{\theta}) \end{aligned} \tag{5.10}$$

And more specifically as equation A.28:

$$\begin{aligned} P[y_N|z_N]P[z_N|\boldsymbol{z}_{-N},\boldsymbol{\alpha}] &= P[y_N|z_N = k : n_k = 0]P[z_N = k : n_k = 0|\boldsymbol{z}_{-N},\boldsymbol{\alpha}] \\ &+ P[y_N|z_N = k : n_K \geq 1]P[z_N = k : n_K \geq 1|\boldsymbol{z}_{-N},\boldsymbol{\alpha}] \\ &= \frac{\tilde{\alpha}}{N + \sum_{k=1}^K \alpha_k - 1} G_0 + \sum_{k=1}^K \frac{n_k}{N + \sum_{k=1}^K \alpha_k - 1} G_k \end{aligned} \tag{5.11}$$

I set up the following model:

$$\boldsymbol{y}_{i'} = \boldsymbol{\mu}_{i'} + \boldsymbol{\epsilon}_{i'} \qquad \text{with} \qquad \boldsymbol{\epsilon}_{i'} \sim IID(0, \boldsymbol{V}_k) \tag{5.12}$$

And further develop it by the following hierarchical structure:

$$
\begin{aligned}
P[\boldsymbol{y}_{i'}|\boldsymbol{\mu}_{i'}, \boldsymbol{V}_{i'}] &= N(\boldsymbol{\mu}_{i'}, \boldsymbol{V}_{i'}) \\
P[\boldsymbol{\mu}_{i'}|z_{i'}, \boldsymbol{V}_{i'}, \tau] &= \begin{cases} N(\boldsymbol{y}_{i'}, \frac{\tau}{1+\tau}\boldsymbol{V}_{i'}) & \text{if } z_{i'} = k : n_k = 0 \\ N(\frac{\boldsymbol{m}_k + \tau \boldsymbol{y}_{i'}}{1+\tau}, \frac{\tau}{1+\tau}\boldsymbol{V}_{i'}) & \text{if } z_{i'} = k : n_k \geq 1 \end{cases} \\
P[\boldsymbol{V}_{i'}|z_{i'}, s, \boldsymbol{S}, \tau] &\sim \begin{cases} IG(\frac{1+s}{2}, \boldsymbol{S}) & \text{if } z_{i'} = k : n_k = 0 \\ IG(\frac{1+s}{2}, (\boldsymbol{S} + \frac{(\boldsymbol{y}_{i'} - \boldsymbol{m}_k)'(\boldsymbol{y}_{i'} - \boldsymbol{m}_k)}{1+\tau})/2) & \text{if } z_{i'} = k : n_k \geq 1 \end{cases} \\
P[z_{i'}|\boldsymbol{p}] &\sim categorical(\boldsymbol{p}) = Multinomial(1, \boldsymbol{p}) \\
P[\boldsymbol{p}|y_{-i'}, \boldsymbol{z}_{-i'}, \boldsymbol{\mu}_{-i'}, \boldsymbol{V}_{-i'}] &\sim DP(\bar{\alpha}, G_0)
\end{aligned}
\tag{5.13}
$$

Where:

$$
\boldsymbol{m}_k = \frac{1}{n_k} \sum_{j:z_j=k} \boldsymbol{y}_j \tag{5.14}
$$

Note that $\boldsymbol{y}_{i'}$ is also included, as it is already allocated to cluster k at this stage. Therefore, if $n_k = 0$, τ is not present in the term that defines the mean distribution. It is not hard to see why:

$$
\frac{\boldsymbol{m}_k + \tau \boldsymbol{y}_{i'}}{1+\tau} = \frac{\boldsymbol{y}_{i'} + \tau \boldsymbol{y}_{i'}}{1+\tau} = \boldsymbol{y}_{i'} \tag{5.15}
$$

5.13 and, notably, A.28/5.11 imply:

$$
p_k = E[\mathbb{1}(z_{i'} = k)] \propto \sum_{j:z_j=k} q_j(y_{i'}, \mu_j, V_j) \qquad \forall k \in [1, K] \tag{5.16}
$$

$$
p_0 = E[\mathbb{1}(z_{i'} > K)] \propto \alpha q_0(s, S, y_{i'}, \hat{m}, \tau) \tag{5.17}
$$

Note that due to the way how q_j and q_0 were defined in equation 5.5, where p_0 denotes the probability of starting a new cluster, it is necessary to constrain the sum of these probabilities as equal to 1.

$$
p_0 + \sum_{k=1}^{K} p_k = 1 \tag{5.18}
$$

As in [Escobar and West, 1995], G_0 is incorporated in q_0 as in equation 5.5.

In the trivial case, V_k is a diagonal matrix en each element is modelled by a univariate inverse-gamma. The multivariate extension of the inverse gamma is an interesting topic, but the literature generally extends the inverse-gamma into an inverse-Wishart in order to model the multivariate case.

Then for the next hierarchical layer find that:

$$
P[z_{i'}|\boldsymbol{p}] = \prod_{i=0}^{K} p_k^{\mathbb{1}[z_{i'}=k]} \tag{5.19}
$$

Obtaining:

$$
\begin{aligned}
P[\mu_k, \sigma_k|z_{i'}]P[z_{i'}|\boldsymbol{p}] \sim\ & \mathbb{1}(z_{i'} = k : n_k > 0) P[\mu_k|z_{i'} = k : n_k = 0, \sigma_k, \tau] P[\sigma_k|z_k = k : n_k = 0, s, S, \tau] + \\
& \sum_{k=1}^{K} \mathbb{1}(z_{i'} = k : n_k \geq 1) P[\mu_k|z_{i'} = k : n_k \geq 1, \sigma_k, \tau] P[\sigma_k|z_k = k : n_k \geq 1, s, S, \tau]
\end{aligned}
\tag{5.20}
$$

With probabilities (5.16 and 5.17) as modelled by:

$$P[\boldsymbol{p}|\boldsymbol{y}_{-i'}, \boldsymbol{z}_{-i'}, \boldsymbol{\mu}_{-i'}, \boldsymbol{V}_{-i'}, s, S, m, \tau] = \alpha q_0(s, S, y_{i'}, \hat{m}, \tau) + \sum_{j:z_j=k} q_j(y_{i'}, \mu_j, V_j) \tag{5.21}$$

Note how m and $y_{i'}$ are swapped in equations 5.21, 5.17 compared to 5.5. Whereas m denotes a prior belief of the center of a cluster for a specific observation in 5.5, in the model and paradigm I am setting up, it is more logical to let an observation belief that the location of a new cluster, if it appears that it does not fit in with any of the other clusters, is somewhere close to itself. Therefore the prior on the location of a new cluster should be close to the observation itself. Contrastingly, the current location of the cluster that he is assigned to, in the Gibbs estimation procedure, is located by $\hat{m}$. Logically, this should be reflected in the probability of starting its own cluster.

In order to avoid unnecssary complication, I take α as constant in this setting. This is however quite common in the literature. α is taken as fixed, or following a gamma distribution with fixed parameters, see [Escobar and West, 1995], in general. In the clustered correlations model α is described by a prior distribution that changes over time and depends on previous distributions and contemporary realizations.

In equation 5.16 the 'rich get richer'-property becomes apparent. As a cluster contains more observations, it will be more likely that $p_k > p_{k'}$. For some cluster $k' \neq k$. Nevertheless, these becomes most apparent in local settings. If a large cluster will be further away from an observation compared to a nearby smaller cluster, the effect will be less pronounced.

Note the similarity of 5.17 with the visualization algorithm t-distributed stochastic neighbour embedding (t-SNE) as described by [Maaten and Hinton, 2008], as well as the similarity of 5.16 with SNE as described before by [Hinton and Roweis, 2003]. The difference between the two is that SNE assumes a normal- where t-SNE assumes a t-distribution. These algorithms decompose high-dimensional data into lower dimensional clustering based using a t- or a normal-distribution, respectively, as a similarity measure. As these algorithms already know the clustering in advance, the objective and complexity of the task is completely different, but the inner workings are similar nonetheless. Basically, they describe for each observation its surrounding cluster and add observations to that cluster if its p-value corresponding to a t- or normal-distribution, respectively, is sufficiently high. By letting the variance parameter free, they succeed in allowing the cluster to be of data-dependent size. Most notably t-SNE has been met with much appraisal, including a comfortable place within contemporary data mining and machine learning courses as well as the Merck Viz price 2012 [1], awarded by Kaggle (an enormous online datascience community recently bought by Google, including members of IBM's Watson and Google's Deepmind). Only at the end of writing I was notified by this mild similarity. The difference should be clear too. In this setting, I make use of a t- and normal-distributions to bayesionally quantify the tendency of observations to cluster together,

[1] http://blog.kaggle.com/2012/11/02/t-distributed-stochastic-neighbor-embedding-wins-merck-viz-challenge/

while modelling the cluster locations and sizes as stochastic as well. Furthermore, the objective is very different and the giants upon whom I stand are of a more (Bayesian) statistical nature.

It is important to point out as well that the clusters are not modelled directly. Although the vector $\boldsymbol{z}$ accounts for partitioning the observations in clusters, the size and location of the clusters are not modelled or used to in the partitioning. Instead, each observations formulates a belief about the location of the cluster it is assigned to given the other observations in that cluster. From the sum of those beliefs we can then infer the location of the cluster simply by summing over the individual normal-inverse-gamma probability density functions. As such, the certainty surrounding the location of a cluster is larger once more observations join it.

The model is estimated by algorithm 3. As mentioned before, allowing too much freedom for the data to decide how large a cluster should be, might cause unwanted results. That is, the data being clustered into one or several gigantic clusters. Algorithm 4 implements a fixed variance V, by minimally altering the algorithm, that might be used to overcome this problem.

Algorithm 3 Gibbs Sampling Estimation Implementation, Cluster Variance Random

Let $\boldsymbol{z}_0$ denote an initial partition of observations into clusters
Let $\boldsymbol{\mu}_0$ denote a vector of initial cluster means for each observation
Let $\boldsymbol{V}_0$ be a matrix of initial cluster variances and covariances for each observation
Let N denote the total number of iterations in the Gibbs Sampling procedure
for $\forall i \leq N$ **do**
    $\boldsymbol{\mu}_{i,\cdot} = \boldsymbol{\mu}_{i-1,\cdot}$; $\boldsymbol{V}_{i,\cdot} = \boldsymbol{V}_{i-1,\cdot}$; $\boldsymbol{z}_{i,\cdot} = \boldsymbol{z}_{i-1,\cdot}$
    for $\forall i' \leq$nr of variables **do**
        Set Up Probabilities
            $p_k = \sum_{j':z_{j'}=k} q_{j'}(\boldsymbol{y}_{i'}; \boldsymbol{\mu}_{i,j'}; \boldsymbol{V}_{i,j'})$ (Probability of assignment to cluster k)
            $p_0 = \alpha q_0(s, \boldsymbol{S}; \boldsymbol{y}_{i'}; \boldsymbol{\mu}_{i-1,i'}; \tau)$ (Probability of assignment to new cluster)
            $[p_0; p_1; ..., p_K] = \frac{1}{\sum_k p_k + p_0}[p_0; p_1; ...; p_K]$
        Determine Transition
            $z_{i,i'} \sim$ Categorical$(p = [p_0; p_1; ..., p_K])$
        Update parameters $\mu_{i,i'}$, $V_{i,i'}$
            if $z_{i,i'} == 1$ **then** (Assign it to a new cluster)
                $V_{i,i'} \sim IG(\frac{1+s}{2}, \boldsymbol{S})$
                $\mu_{i,i'}|\boldsymbol{V}_{i,i'} \sim N(\boldsymbol{y}_{i'}, \frac{\tau}{1+\tau}\boldsymbol{V}_{i,i'})$
            else (Assign it to an existing cluster k)
                $m_k = \frac{1}{n_k}\sum_{n:z_n=k} y_n$
                $V_{i,i'} \sim IG\left(\frac{1+s}{2}, \boldsymbol{S} + \frac{(\boldsymbol{y}_{i'} - \boldsymbol{m}_k)'(\boldsymbol{y}_{i'} - \boldsymbol{m}_k)}{1+\tau}\right)$
                $\mu_{i,i'}|\boldsymbol{V}_{i,i'} \sim N(\boldsymbol{m}_k, \frac{1}{1+\tau}\boldsymbol{V}_{i,i'})$
Output: $\forall i$): $\boldsymbol{\mu}_{i,\cdot}$; $\boldsymbol{V}_{i,\cdot}$; $z_{i,\cdot}$

As the algorithm above is extended in further chapters, it is of vital importance to understand the algorithm, and specifically the labels, above. As such, note that i denote the iterator that loops over the number of Gibbs sampling steps that are ought to be performed. Within each iteration of this loop, we update each of the variables given the other variables. In order thus to loop over the variables of interest, I manufacture the indicator i' to keep track of this. As such, $z_{i,i'}$ specifies the cluster observation i' belongs to in iteration i. In line 9, I implicitly loop over the clusters that already exist using indicator j' and labelling of clusters using indicator k. As such $\sum_{j':z_{j'}=k} q_j()$ denotes the sum of individual probabilities of assigning observation i' to the same cluster k as observation j' belongs to. The reason for the sub-optimal nomenclature for the indicators, iterators and labels is because it allows me to preserve a logical connection between this algorithm and later algorithms that extend it.

As $\boldsymbol{y}$ denote the observations, these do not require a decomposition as $\boldsymbol{\mu}$, $\boldsymbol{V}$ and $\boldsymbol{z}$ require.

Algorithm 4 Gibbs Sampling Estimation Implementation, Cluster Variance Random

1: Let $\boldsymbol{z}_0$ denote an initial partition of observations into clusters
2: Let $\boldsymbol{\mu}_{.0}$ denote a matrix of initial cluster means
3: Let $\boldsymbol{V}_0$ be a matrix of initial cluster variances
4: Let N denote the total number of iterations
5: **for** $\forall i \leq N$ **do**
6: $\quad \boldsymbol{\mu}_{i,.} = \boldsymbol{\mu}_{i-1,.}, \boldsymbol{V}_{i,.} = \boldsymbol{V}_{i-1,.}, \boldsymbol{z}_{i,.} = \boldsymbol{z}_{i-1,.}$
7: $\quad$ **for** $\forall i' \leq$nr of variables **do**
8: $\quad\quad$ Set Up Probabilities
9: $\quad\quad\quad p_k = \sum_{j':z_{j'}=k} q_{j'}(\boldsymbol{y}_{i'}; \boldsymbol{\mu}_{i,j'} l \boldsymbol{V}_{i,j'})$ (Probability of assignment to cluster k)
10: $\quad\quad\quad p_0 = \alpha q_0(s, \boldsymbol{S}, \boldsymbol{y}_{i'}, \boldsymbol{\mu}_{i-1,i'}, \tau)$ (Probability of assignment to new cluster)
11: $\quad\quad\quad [p_0; p_1; ...; p_K] = \frac{1}{\sum_k p_k + p_0}[p_0; p_1; ...; p_K]$
12: $\quad\quad$ Determine Transition
13: $\quad\quad\quad z_{i,i'} \sim$ Categorical$(p = [p_0; p_1; ...; p_K])$
14: $\quad\quad$ Update parameters $\mu_{i,i'}$, $V_{i,i'}$
15: $\quad\quad\quad$ **if** $z_{i,i'} == 1$ **then** (Assign it to a new cluster)
16: $\quad\quad\quad\quad V_{i,i'} = V$
17: $\quad\quad\quad\quad \mu_{i,i'} | \boldsymbol{V}_{i,i'} \sim N(\boldsymbol{y}_{i'}, \frac{\tau}{1+\tau}\boldsymbol{V}_{i,i'})$
18: $\quad\quad\quad$ **else** (Assign it to an existing cluster k)
19: $\quad\quad\quad\quad m_k = \frac{1}{n_k}\sum_{n:z_n=k} y_n$
20: $\quad\quad\quad\quad V_{i,i'} = V$
21: $\quad\quad\quad\quad \mu_{i,i'} | \boldsymbol{V}_{i,i'} \sim N(\boldsymbol{m}_k, \frac{1}{1+\tau}\boldsymbol{V}_{i,i'})$

Output: $\forall i$): $\boldsymbol{\mu}_{i,.}; \boldsymbol{V}_{i,.}; z_{i,.}$

5.3.3 Greedy MAP as solution to the MAP Problem

When attempting to make sense of the obtained posterior distribution using Gibbs sampling by defining the modes, it is apparent that it is not possible to make use of the cluster allocation directly. First of all, because of some form of the label switching problem. There is no guarantee, within the algorithm, that clusters that pop up during the sampling procedure are that do not have different do not actually represent the same cluster. This merely comes from the fact that clusters are stochastically pop up and are deleted. This renders the use of the labels z, not particularly useful, after the simulation. However, it is more useful to derive a Maximum A Priori clustering by using the frequencies with which 2 observations are clustered together throughout the sampling procedure. Nevertheless, it is known that complex posteriors tend to yield the MAP problem, or, more bluntly, suffer from intractable MAPs. This problem has been accounted in the literature, see [Raykov et al., 2014] and [Broderick et al., 2013], however solutions focus on adjusting the Gibbs sampling procedure. This gives rise to quite a complex high-dimensional probability distribution of the system. The problem that hitherto arises is how to quantify those results, the probabilities, into actual clusters. For example, consider a fictional result as in table 5.1: Observation 1 has a high probability to cluster with 2,3 and 4. However,

Table 5.1: Gibbs Probabilities of Clustering together observations

Observation	1	2	3	4
1	1	0.9	0.9	0.7
2	0.9	1	0.7	0.2
3	0.9	0.7	1	0.2
4	0.7	0.2	0.2	1

both 2 and 3 seem to have a low probability to be clustered with 4. The example exagerates the problem, but it should be clear that from these probabilities that one should be careful with regard to summarizing the posterior in a few statistics. Nevertheless, popular are both the Maximum A Priori (hereinafter MAP), which is simple and popular, and the last partition revisited, which is surprisingly often used in LDA's. However, in this case it is not possible to sort the probabilities and accordingly define clusters. As with the example, it is questionable whether there exist two clusters, namely (1,2,3) and (4), or one larger cluster (1,2,3,4). In order to make sense of those distribution, one would like to obtain the MAP in this scenario. But that is however a bit more complex as such a MAP only makes sense if the it does not violate some form of transitivity. That is, if 1 and 2 are clustered together, as well as 2 and 3, than it is necessary that 1 and 3 are clustered together as well. In order to adhere to these principles and obtain an image of the posterior, I propose to make use of a greedy MAP procedure setting up a probability matrix that describes $\forall i, j$ whether observations y_i and y_j are in the same cluster. As far as I know, there does not exist an account within the literature currently of this procedure. Using a lower bound in terms of probability needed to define a cluster, we can then set up those clusters. Then, by lowering the bound, iteratively we find new strong connections between the observations and either add them to existing clusters or put them together in a new cluster.

Once clusters grow bigger in this procedure, it becomes harder to join it as for a observation to join a cluster it needs to have connection stronger than the current lower bound with each of the current cluster occupants. Schematically, the greedy MAP procedure can be summarized as:

Algorithm 5 Greedy MAP to convert posteriors to clusters

Input: $P_{lowerbound}$, P_{min}, P_{step}

while not all y_i assigned to clusters And $P_{lowerbound} > P_{min}$ **do**
  for all i, j **do**
    if $P_{i,j} > P_{lowerbound}$ **then**
      if $C_{z_i} = 0$ and $C_{z_j} = 0$ **then** y_i and y_j in new cluster: $z_i = z_j = \max C + 1$
      if $C_{z_i} = 0$ and $C_{z_j} > 0$ **then**
        if $\forall\ y_k \in C_{z_j}$: $P_{k,j} > P_{min}$ **then** Assign y_i to cluster: C_{z_j}: $z_i = z_j$
      if $C_{z_i} > 0$ and $C_{z_j} = 0$ **then**
        if $\forall\ y_k \in C_{z_i}$: $P_{i,k} > P_{min}$ **then** Assign y_j to cluster: C_{z_i}: $z_j = z_i$
  $P_{min} = P_{min} - P_{step}$

Where z_i denotes the cluster label of observation y_i and C the collections of clusters.

It is important to note that the greedy MAP algorithm does not change the underlying distribution, but is merely a tool to obtain an actual clustering that can be used to construct an image of the distribution and to further assess the performance of the model with this data for several different values for the underlying (hyper)parameters, which are discussed in the next chapter.

6

2 Step Procedure Results

In order to showcase the results that can be obtained from the methodology as described in the Chapter 5, this chapter first introduces the Fama and French data set that is used. For these observations of stock portfolios, the model is tested for several hyperparameter-settings. The values of the hyperparameters in the Kalman filter remain unaltered as introduced in Chapter 4, whereas the values of the parameters in the secondary Bayesian clustering step are one-by-one adjusted to analyse the sensitivity of the model. The results with several values for the hyperparameters can be found in the appendix C. Both the greedy MAP procedure, as introduced in section 5.3.3, as well as a last partition revisited are employed to obtain visuals. Although the underpinning of the model is intuitive, it appeared that optimal tunage of the hyperparameters is vital. For this particular dataset, robustness of the model should be largely rejected.

6.1 Data 2-Step Model

The data consists of 49 different industry groups and is part of the Fama and French dataset, available at their website: `http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html`. These industry groups are observed monthly starting from July 1926. Because some industries came into existence after the starting date, it is convenient to drop the observation before. That appears to be July 1969. At the moment of obtaining the data, the end date was October 2017. There are thus 580 monthly observations of the 49 portfolios without a single gap in the data. As for the time-window used, although there are many options, I have decided to take a time-window of length 6 starting at the 26th observation, which happens to be the timewindow: september 1971 - Januari 1972. This timewindow is convenient because it shows a certain degree of separability as well as series that tend to have similar dynamics.

The starting dates per industry as well as the industries themselves can be found in appendix B.

Note that in all of these plots colors correspond to clusters. As such, when series have identical colors, they are in the same cluster. Because the colors are chosen as random and the colors might be overlapping or hard to depict with the naked eye, the number of clusters is also depicted in the appendix B.

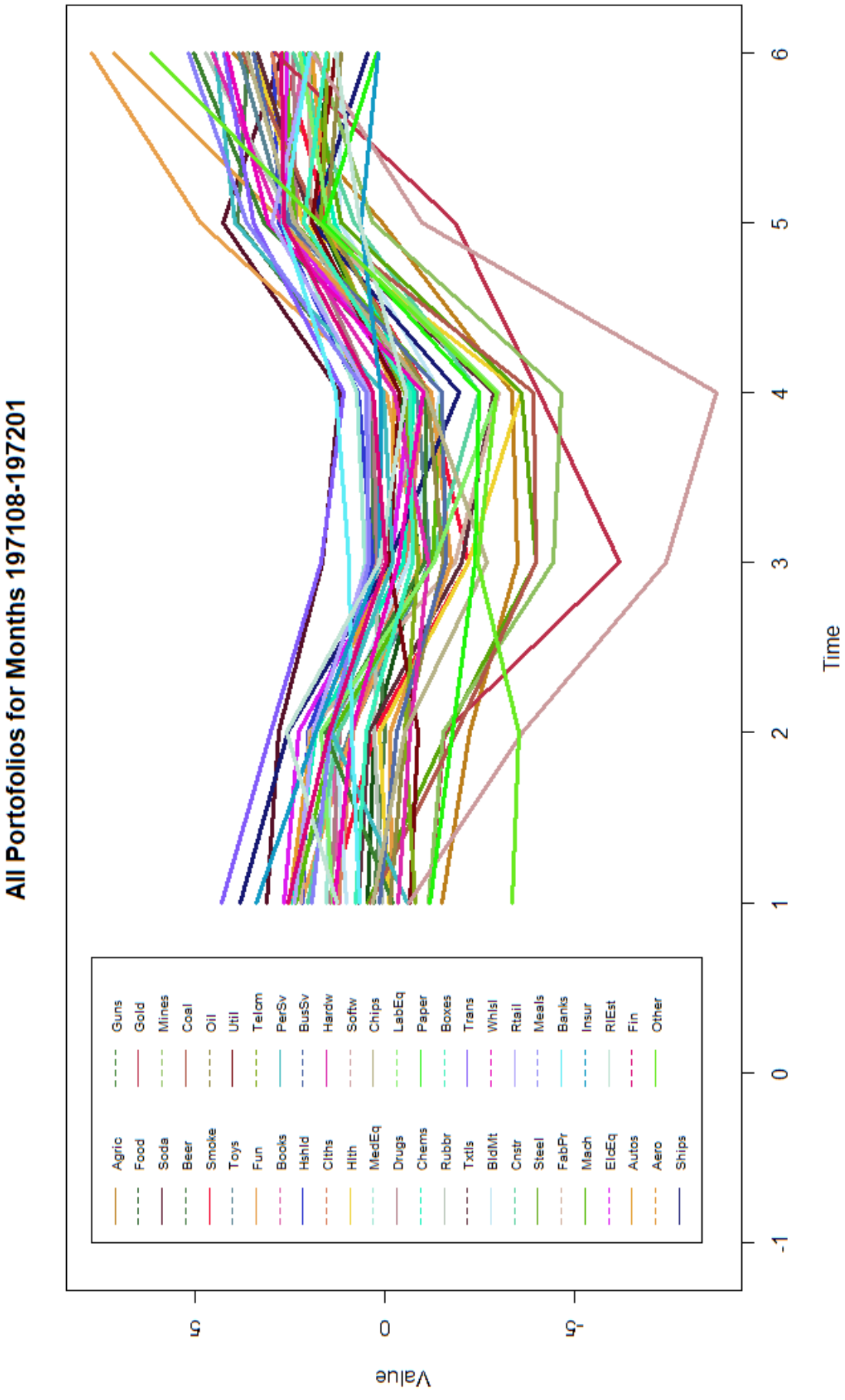
All Portofolios for Months 197108-197201
Value
5
0
-5
Time
-1
0
1
2
3
4
5
6
Agric
Food
Soda
Beer
Smoke
Toys
Fun
Books
Hshld
Clths
Hlth
MedEq
Drugs
Chems
Rubbr
Txtls
BldMt
Cnstr
Steel
FabPr
Mach
ElcEq
Autos
Aero
Ships
Guns
Gold
Mines
Coal
Oil
Util
Telcm
PerSv
BusSv
Hardw
Softw
Chips
LabEq
Paper
Boxes
Trans
Whlsl
Rtail
Meals
Banks
Insur
RlEst
Fin
Other

Fig. 6.1

6.2 Results for Different Hyperparameter Values

6.2.1 Default Parameter Values

The Kalman filter (Chapter 4) was applied with parametrization:

$$\begin{aligned} Z = T = R = H = V = 1 \\ H = 10 \\ a_0 = 0 \\ P_0 = 1 \end{aligned} \tag{6.1}$$

For the model:

$$\begin{aligned} y_t = Za_t + \epsilon_t, \qquad & \epsilon_t \sim N(0, H) \\ a_{t+1]} = Ta_t + Rw_t, \qquad & w[t] \sim N(0, V) \end{aligned} \tag{6.2}$$

Then the hyperparameters of the second step are set as:

$$\begin{aligned} s &= 0.05 \\ S &= \frac{diag(colStdevs(y))}{10} \\ \tau &= 2 \\ m &= colMeans(y) \\ N_{iter} &= 1000 \\ \alpha &= 0.001 \\ \bar{V} &= mean(diag(S)) \end{aligned} \tag{6.3}$$

Whereas N_{iter} denotes the number of iterations and $\bar{V}$ denotes a fixed estimation of the parameter V_i in model 5.14. This might be necessary in order to constrain the size of the clusters. Otherwise the nature of the observations might force the number of clusters to converge to 1. A less meaningful result. Evidently, this is tested for as well.

To understand the effect of the parameters and obtain a rough analysis of the performance, the parameters (α, s, S and τ) are slightly altered. Next to that the model as stated in the previous chapter without fixing the variance V is estimated as well. The greedy MAP (section 5.3.3) and the last partition revisited results are displayed for all these submodels.

6.2.2 Varying α, figures C.3 - C.6

The results in section C.2 indicate that the effect of α is quite strong. Multiplying and dividing the default value by 10 increases the number of clusters significantly whereas downsizing has the opposite effect. Higher alphas increase the likeliness of the foundation of new clusters. Therefore these results should not be too surprising. It is however interesting that we require the value for α to be so low, as this implies that the prior has a small effect on the predictive distribution. In other words, the prior is not very informative in general, but large deviations might have strong effects on the clustering results.

6.2.3 Varying s, figures C.7 - C.10

If V was not fixed to a certain value, s would have had a more profound effect. In this case, however, s only is used in modelling the probability for a certain observation to set up its own cluster (5.17). In fact, an increase in s, increases this probability as well. This is due to the fact that this measure assesses the fit of a certain observation and the cluster it currently is allocated. Larger values for s tend to stretch out t-distribution, as it parametrizes the degrees of freedom. Whether this is wanted or not depends in turn on the data and the tuning of the other parameters.

This is apparent from C.3 as well. Images C.7 and C.8 show the increased difficulty of the simulation process to allocate with certainty, whereas C.9 and C.10, with s equal to $5e-04$ instead of 5, observations that lie close together tend to find each other within clusters with greater ease.

6.2.4 Varying S, figures C.11 - C.14

From the results in appendix C.4 it becomes apparent that S has a profoundly different effect, as compared to s. Equation 5.5 shows this as well. That is, inflating S by a factor 100 forces the clusters to cascade into 1 large cluster. Vica versa, decreasing S by a factor 100 breaks the clustering the other way around as it clusters 49 portfolios into 31 and 33 clusters, for greedy MAP and last partition revisited respectively. As such the robustness hinges most strongly on a proper tuning of S, in this default setting. A larger S indicates that the algorithm favors rather large clusters, an observation denoted by [Escobar and West, 1995] as well.

6.2.5 Varying τ, figures C.15 - C.18

τ indicates the degree of confidence one has in the accuracy of the prior. From appendix C.5 it becomes apparent that a greater confidence in the prior in turn leads to a lower number or clusters. In essence, from judging the plots alone the performance of the algorithm in terms of discerning clusters seems to be most promising for figures C.15 and C.16, with $\tau = 200$. Next to that, note that the effect of blowing up τ has less of a pronounced effect as well.

6.2.6 Conclusion

Although the model has decent intuition complemented by the possibility to cluster observations in a fully Bayesian fashion, the performance seems to be questionable for this dataset. An appropriate degree of robustness seems hard to be attained, which is problematic if it is not straightforward to set the values of the hyperparameters. Next to that, solely with these results it is difficult to pinpoint the reason the simulation procedure has obtained a certain clustering, nor whether it obtained the complete posterior density. For example, the Gibbs sampling procedure could encounter strong competition among clusters and get stuck at a sub-optimal optimum where a multitude of clusters are competing for survival. However, instead of competing for survival, it could similarly be the case that the clusters are actually stable but tiny. Solely from obtaining the allocations, this remains hard to tell. Whereas [Escobar and West, 1995] notes that

nonparametric empirical Bayes have limited performance, I do not find enough evidence to refute that statement from these results.

Finally, one of the strongest part of this algorithm is that it allows to model the whole distribution instead of a single or several points. It is therefore inconvenient that the resulting posterior attains such a complexity that requires one to focus solely on several noteworthy aspects of that solution. As such, this model might inhibit an intriguingly better performance than the particular sample of results that I could obtain in appendix C. It might be the case that the proposed Greedy MAP procedure (section 5.3.3) as well as last partition revisited appear to be sufficient to display the information withheld from the posterior, but that is something that remains to be seen. Being able to set up a distribution that describes all the interactions and dynamics is great, but being able to effectively evaluate seems not to be straightforward.

7

Evolutionary Clustering and Shift to Variance-Covariance Matrix Modelling

Whereas the previous chapter dealt with picking the clusters on individual time-windows, it seems valuable to extend the model to incorporate relations spanning the time dimension as well. Although the Bayesian clustering model is intuitive and performs well, this cannot be said about the methods that are available to further translate this into actual clustering in the time domain. The hard part of translating the distributions obtained from the Gibbs sampling procedure into actual clusters over time is that first of all these distributions have to be summarized into a certain clustering in order to visualize this later. An even harder problem is making sense of the individual time-snap-shots. How to connect these distributions in a meaningful way across the time-domain? Is it even possible to derive a meaningful relation from this construction? At this point of my thesis, I started to wonder whether usage of clusters makes sense and how to proceed to scale the previous model. At such a point, one should ask him- or herself whether there even is a reason to stay on the path one was on. Furthermore it is questionable whether clustering subsequences of time series have any added value in analytics. An interesting article and fun read, [Keogh and Lin, 2005], strongly doubts its performance and attemps to prove its claim: "Clustering of time-series subsequences is meaningless". They state that cluster extracted by any clustering algorithm are essentially random due to being forced to obey certain defining pathological constraints. As the paper continues its argument, it shows that the obtained results from any clustering algorithm on subsequence time series are not the consequence of properties of stock and market data, but more likely due properties of the sliding window feature extraction. They find that clusters found by cluster algorithms are not significant more similar to each other than they would have been to a clusters centers derived from a random walk. As such they denote the clustering of time-series meaningless because the output of said algorithms is independent of the input. The consequence is that one cannot bluntly apply clustering techniques without taking into account the inherent properties of time series. Therefore, any clustering algorithm should not heavily depend on the clustering of subsequences of time series.

In conclusion, in order to analyze the dynamics of a chaotic system such as the economy, it is necessary to deviate from the path I headed on in Chapters 5 and 6. It is necessary to come up with a more time-integrated version and model time-dynamics directly. In the literature

there are various models offered, but a holy grail is yet to be found. This chapter concludes the literature by firstly introducing several ideas that come from the clustering and data mining fields, while explaining their shortcomings with regard to the task at hand. Then it introduces more statistically grounded methodologies. Finally, it discusses how to proceed. In Chapter 8 several of the ideas are used either directly or as inspiration to create a model that does suffice the criteria that are implicitly laid out in my considerations. That is, a model that efficiently incorporates the clustering model of Chapter 5 and extends it into a time-dimensions while adhering to Bayesian and statistical validity.

7.1 Direct Evolutionary Clustering

7.1.1 Minimizing loss functions

7.1.1.1 Agglomerative Hierarchical Clustering

The agglomerative hierarchical clustering starts with a cluster for each observation and a similarity matrix between them. Then it selects those objects that have the highest similarity, merges them into one cluster and recalcultates the similarity matrix. This is repeated until a certain threshold is reached. By merging the object, the cluster is redefined as the average of the objects. This is often visualized by a bottom-up binary tree whose leaves are the initial observations. These leaves come together in branches, and these branches in larger branches, that form in this analog the clusters. The final tree T_t is then clustering at time t. The goal is then to find a clustering such that the historical cost is minimized (hc) and the accuracy of the cluster sequence (sq) is optimized. Denoting a particular cluster at time t as T_t and a similarity matrix as M_t, this then becomes for a total number of timewindows N:

$$\max_{C} \sum_{t=1}^{N} sq(T_t, M_t) - \lambda \sum_{t=2}^{N} hc(T_{t-1}, T_t) \tag{7.1}$$

For some nonnegative penalty λ.

7.1.1.2 K-Means

Another commonly used method in non-evolutionary clustering is the K-means method, which relies on minimizing the within-cluster sum of squares:

$$\arg\min_{\mu} \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2 \tag{7.2}$$

An extension of the methodology in the time-domain necessitates reformulating the objective function by incorporating again a historic cost function in terms of the means of the clusters:

$$\arg\min_{\mu} \sum_{t=2}^{T} \sum_{i=1}^{K} ||\mu_{i,t} - \mu_{i,t-1}|| + \sum_{t=1}^{T} \sum_{i=1}^{k} \sum_{x_t \in S_{i,t}} ||x_t - \mu_{i,t}||^2 \tag{7.3}$$

Where the clusters are denoted by $\boldsymbol{S} = (S_1, ..., S_K)$

7.1.1.3 Organizing Maps

Finally, a well studied but less known methodology stems from the "Self-Organizing Map" ([Kohonen, 1998], SOM). The basic idea is to map data patterns onto a n-dimensional grid of neurons or units. The grid that is formed during the process is known as the output space. If we assume that each unit is a cluster center, and there are k units in total, then the same task as k-means is performed by the algorithm. The algorithm is described by: During the training,

Algorithm 6 Self-Organizing Map

Let $\boldsymbol{X}$ be the set of observations: $x_1, ..., x_n$
Let W be a grid of units w_{ij} where i and j are their coordinates on that grid
Let α the learning rate, taking values in $[0, 1]$
let r be the radius of the neighborhoud function $h(\boldsymbol{w}_{i,j}, \boldsymbol{w}_{i',j'}, r)$

while $\alpha > 0$ **do**
    for $k = 1$ to n **do**
        for all do $\boldsymbol{w}_{i,j}$ calculate $d_{ij} = ||\boldsymbol{x}_k - \boldsymbol{w}_{i,j}||$
            Select w_{ij} that minimizes d_{ij}, denote it as $w_{i',j'}$
            update: $w_{i,j} = w_{i,j} + \alpha h(w_{i',j'}, w_{i,j}, r)||x_k - w_{ij}||$
    Decrease α and r

the SOM forms an elastic net that folds onto the input data. As such, it can be preserved as a mapping from input space to a lower-dimensional grid of map units that preserves the topology. Methods derived from SOM perform reasonably good in the cross-section domain and applications to evolutionary clustering are known, but have not been subjected to proper peer-review. For an application to evolutionary clustering, see for example [Ramanathan and Guan, 2006].

7.1.2 Extending the Dirichlet Process

Another methodology considers a dependent Dirichlet process. Although there are quite some implementations available, they can broadly be summarized by those that define a relation over time via π_t, as in the stick breaking construction, and those that do so by using G_0. For example, the Dirichlet Process Chain Model as introduced by [Xu et al., 2008a]. It uses a exponential smoother for the transition between time-windows by defiing the cluster mixtuer at time t as:

$$\boldsymbol{\pi}_t = \sum_{\tau=1}^{t} exp\big(-\eta(t-\tau)\big)\boldsymbol{\pi}_\tau \tag{7.4}$$

Which is then further illustrated and developed by defining a smooth prior weight for cluster k at the beginning of time t as:

$$w_{t,k} = \sum_{\tau=1}^{t-1} exp\big(-\eta(t-\tau)\big)n_{\tau,k} \tag{7.5}$$

With $n_{\tau,k}$ as the number of observations belonging to cluster k. Similar to the Chinese Restaurant process, one obtains then:

$$P[z_{t,i} = k|\boldsymbol{z}] \propto \begin{cases} \frac{w_{t,k}+n_{t,k}^{-i}}{\alpha+\sum_{j=1}^{K} w_{t,j}+n_t-1} \text{ if } k \text{ is an existing cluster} \\ \frac{\alpha}{\alpha+\sum_{j=1}^{K} w_{t,j}+n_t-1} \text{ if } k \text{ is a new cluster} \end{cases} \tag{7.6}$$

On the other hand, [Ahmed and Xing, 2012] model the dimension by parametrizing the base distribution G_0. It introduces the temporal Dirichlet process Mixture model (TDPM) as a framework in which it explicitly moels the parametrization of each component over time in a markovian fashion. With other words, the random measure G is time-varying and modelled as:

$$G_t|\boldsymbol{G}_{t-1}, G_0, \alpha \sim DP\Big(\alpha + \sum_k m_{k,t}, \sum_k \frac{m_{k,t}}{\sum_t m_{l,t}+\alpha} G_{t-1} + \frac{\alpha}{\sum_t m_{l,t}+\alpha} G_0\Big) \tag{7.7}$$

Where $m_{k,t}$ is defined as:

$$m_{k,t} = \sum_{\delta=1}^{\Delta} exp\Big(\frac{-\delta}{\lambda}\Big) n_{k,t-\delta} \tag{7.8}$$

With $n_{k,t}$ denoting the number of observatons in cluster k at time t, Δ denoting the range of relevant historic distributions and λ the decay factor.

7.1.3 Hidden Markov State Transitions

Finally, instead of modelling the clusters on static time-windows and incorporating past information in the estimation procedure, we can avoid a bit of the complexity and model the change between time-windows. As such one has to define states and transition matrices to connect them. Consider for example the infinite hierarchical hidden markov state model (iH^2MS) as defined in [Xu et al., 2008b], which establishes a hierarchical transition matrix using again a Dirichlet process. Evidently, non-hierarchical models exist as well.

7.1.4 Model Selection

First of all, although the methods that minimize certain clever loss functions might be effective and an important part of the machine learning field, they generally do not address uncertainty nor integrated biases. Furthermore, it is not a priori true that the feasible set of the objective function is convex. If it ends up being non-convex, there is also no certainty of the procedure arriving at the optimal. It is not impossible to overcome these problems, they do not take into account statistical validity and particularly questionable with regard to overcoming the issue raised by [Keogh and Lin, 2005]. They do not seem very appropriate to model processes of such stochastic nature. As such, they are not line with the Bayesian nature of this thesis and notably the clustering model of Chapter 5.

Both an extension of the Dirichlet process within the model as introduced in Chapter 5 into the time-domain by augmenting the stick-breaking process or implementing a hidden Markov might pose drawbacks with regard to building and estimating that certain model in an appropriate

fashion. Although the models might have excellent performance for certain datasets[1], the clustering models as introduced in the Chapter 5 and 6 shows that that might not be the case for financial datasets at hand. The questionable robustness (Chapter 6), the fact that the implementation of the clustering algorithm does not allow to be scaled into these models efficiently and the issue surrounding clustering of subsequences of time series (section 7.1.1), cause me to shy away from implementation. Furthermore, the power of the Dirichlet process, as introduced in Chapter 3, seems to lie in its ability to attach a degree of similarity to seemingly distinct observations. In light of that, it does not seems to make sense to parametrize the Dirichlet process as invasive as TDPM (section 7.1.2). The clustering model as proposed in chapter 5 already is quite flexible and modifying the base distribution to accomodate for time dynamics might weaken the structure of colluding dynamics I intend to impose. The same is true for the hidden state markov transition models (section 7.1.3), as they require quite an extensive restructuring as well. On the contrary, simplifying the time dynamics using exponential smoothers seems to restrictive. As such, the evolutionary clustering methodologies do not provide a middle ground. That is; A methodology perfectly suited for the problem at hand that does allow for modelling the time dimension appropriately as well as leaving the Dirichlet process fairly intact. Keeping the Dirichlet process intact would allow for a fairly direct implementation of the previously defined clustering model, and thus for a further exploration of the usability of the Dirichlet process and the proposed estimation methodology in particular. Therefore it seems logical to turn to the modelling the dynamics in terms of variance covariance matrices rather than clustering dynamics directly, and hence stick closer to econometrics than machine learning.

7.2 Indirect Evolutionary Clustering: Correlation and Variance-Covariance Matrices

Consider the hierarchy of the model as given in section 5.3.2:

$$\begin{aligned} y_t|\phi_t &\sim f(\phi_t) \\ \phi &\sim NIG(z_t) \\ z &\sim categorical(p_t) \\ \boldsymbol{p}_t &\sim DP \end{aligned} \tag{7.9}$$

Although most attention, up until now, has been given to the realization of the clusters in terms of its stochastic locations and probabilistic occupants, it actually appears that the most important part of this model actually is at the bottom. Namely the distribution of the mixture-proportions, the underlying probability that assigs observations to clusters and hitherto sets the parameters of the normal-inverse-gamma distribution. However, if we turn our attention to the modelling of the probabilities of belonging to a certain cluster in order to describe their dynamics as reasonably similar, then clustering those probabilities and use those clusters to condition, actually is a weak metric for the correlation matrix. Leading to the question whether it is actually more beneficent to model the correlation matrix directly, but with applying a sparsity

[1] Most notably NLP

prior such that the clusters arise more naturally. As such, the focus of the remainder of the thesis is shifted slightly of clustering directly, but focuses on modelling correlation using existing Bayesian methods. As such, the remainder of this chapter focuses on the state of the art literature with regard to Bayesian semi- and nonparametric variance-covariance matrix modelling. As such, the rest of this chapter recaps the literature on the field of variance-covariance modelling, which appears to be the wild-west of econometric modelling. Many different models have been proposed and therefore this chapter does not claim to survey them all equally in depth. The next chapter develops an applicable model that employs the variance-covariance matrix in order to obtain a more appealing methodology of clustering.

7.2.1 Schools of Thought on Multivariate Volatility Models

It has been known for quite some time that variances and covariances change over time ([Andersen et al., 2007]). Furthermore, recently with both increased computational power and the availability of high-frequency stock trading data the interest for modelling the variance and covariances of these stocks has taken a spike. The idea is quite interesting, but it inevitably involves the inobservability constraint. We simply can only observe realizations of the data generating process and not underlying dynamics that are at play. It is therefore naturally yet stringent to impose structure in order to derive the dynamcis that are at play. The literature of multivariate volatility models can be broadly divided into 3 sections. Namely the multivariate GARCH models, stochastic volatility models and predetermined variance models.

7.2.1.1 Multivariate GARCH

Let y_t be a vector of dimension $N \times 1$, than define the general model:

$$\begin{aligned} y_t &= \mu_t + \epsilon_t \\ \epsilon_t &= H_t^{1/2} z_t \end{aligned} \tag{7.10}$$

Whereas:

$$z_t \sim N(0, I_n) \tag{7.11}$$

The further specification of H_t tells us which GARCH model we are exactly making use of. A side-condition is that H_t is positive definite, as it represents the variance-covariance matrix.

A general formulation, called the VEC(p,q)-model, as proposed by [Bollerslev et al., 1994], models elementwise the H_t matrix as a ARMA(p,q) model. The VEC(1,1) is represented by:

$$h_t = c + A\eta_{t-1} + Gh_{t-1} \tag{7.12}$$

Whereas $h_t = ech(H_t)$ and $\eta_t = vech(\epsilon_t \epsilon_t')$. The vech operator stacks the lower triangular portion of the $N \times N$ matrix as a $N(N+1)/2 \times 1$-vector. A and G are square parameter matrices of order $(N+1)N/2$ and c is a parameter vector with dimension $N(N+1)/2 \times 1$. The number of required parameter parameters that need to be specified is equal to $N(N+1)(N(N+1)+1)/2$. and practically overwhelming. In order to circumvent the problem, [Bollerslev et al., 1994] suggested the diagonal VEC (DVEC) in which A and G are restricted to be diagonal matrices. There

are various other parametrisations available from the literature as well including exponential GARCH (EGARCH), quadratic GARCH (QGARCH) and integrated GARCH (IGARCH). See for example [Palm, 1996].

An interesting model with close ties to the GARCH modelling paradigm is the Adaptive forgetting factor for evolutionary clustering, hereinafter AFFECT. It was introduced by [Xu et al., 2014] and involves shrinkage estimation. The idea treats evolutionary clustering as a problem of tracking followed by static clustering. It involves modelling the observed matrix of proximities between objects at each time step as a linear combintaion of a true proximity matrix and a zero-mean noise matrix. Note the similarity of the procedure with the shrinkage estimation of covariance matrices as in [Ledoit and Wolf, 2003]. Whereas the true proximities can be viewed can be viewed as unobserved states of a dynamic system:

$$W_t = \Psi_t + N_t \tag{7.13}$$

Where Ψ_t is an unknown deterministic matrix of unobserved states, the true proximity matrix. N_t is the mean-zero error matrix. Although a common approach would be to use the Kalman filter, as the size of the matrices (n) grow large, taking the inversion of an $O(n^2)\times O(n^2)$ covariance matrix, might become infeasible. However, we can take a different approach:

$$\hat{\Psi}_t = \alpha_t \hat{\Psi}_{t-1} + (1-\alpha_t) W_t \tag{7.14}$$

As such it obtains a smoothed proximity matrix. Where α is the forgetting factor. Indeed, the model 7.14 can be seen as a matrix variant of IGARCH. The integrated GARCH model, where parameters are restricted to sum to one.

7.2.1.2 Predetermined Variance Models

Although important to note that these type of models exist and provide easily applicable ways for preliminary research with respect to variances and covariances, these models are generally not met with the underlying statistical frameworks. Models in this category include for example the exponentially weighted moving average model:

$$\sigma^2_{EWMA,t} = (1-\lambda)\varepsilon^2_{t-1} + \lambda \sigma^2_{EWMA,t-1} \tag{7.15}$$

7.2.1.3 Multivariate Stochastic Volatility Modelling

Alternatively, it is possible to set up a model containing an unobserved variance component, the logarithm of which directly modelled as a linear stochastic process. These models are known as the multivariate stochastic volatility models (MSV). These discrete time models can be thought of as Euler approximations of underlying continuous diffusion models. Although they are often difficult to estimate, they generalize the multivariate series in a more natural way and offer more flexibility with regard to the data at hand. The first MSV model proposed in the literature

is due to [Harvey et al., 1994] as:

$$\begin{aligned} y_t &= H_t^{1/2}\epsilon_t \\ H_t^{1/2} &= diag\Big(exp(h_{1t}/2, ..., exp(h_{nt}/2)\Big) \\ h_t &= \mu + \phi \circ h_t + \eta_t \\ \begin{pmatrix}\epsilon_t \\ \eta_t\end{pmatrix} &\sim N\left(\begin{bmatrix}0 \\ 0\end{bmatrix}, \begin{bmatrix}P_\epsilon & 0 \\ 0 & \Sigma_n\end{bmatrix}\right) \end{aligned} \tag{7.16}$$

Whereas $\circ$ denotes the hadamard operator, which multiplies matrices elementwise. Although the GARCH literature is reasonably mature, the field of MSV is nowadays more dynamic. As late as 2006, [Asai et al., 2006] noted posed that "the MSV literature is still in its infancy".

7.2.2 Recent Developments

Although the existing literature on GARCH models is abundant, it has not been successfull in eliminating drawbacks. Particularly symmetry and positivity constraints require compliciated restrictions on parameters that are difficult to interpret already. Furthermore, generally the GARCH models are not invariant to a change in portfolio allocation. Although the MSV models offer more flexibility, it requires sophisticated simulation methods. Especially when the model was specified using latent variables.

Recent advancements of the scientific field, and simultaneously the direction this thesis is heading, consists for several cases of making use of a dynamic extension of the Wishart distribution. These were developed around the same time and form attempts to improve upon earlier stochastic covariance models such as [Bollerslev, 1990]. This section presents several interesting directions that exist in the literature.

7.2.2.1 HEAVY Models

High-frEquency-bAsed VolatilitY (HEAVY) models was introduced by
[Shephard and Sheppard, 2010] and further developed by [Noureldin et al., 2012]. Its univariate specification slightly differs from the GARCH model and depends, as the name gives away, on realized volatility. The usage of realized volatility revolves on what frequency the data should be observed, which is a not the main concern of this thesis. It is nonetheless an ongoing discussion as it is unclear where the balance between more data and better data can be find. That is, as the frequency increases, micro-structures effects become more outspoken and tend to dominate the models. It remains questionable if and in what degree this helps modelling variance-covariance matrices. Anyhow, the HEAVY model can be seen as a natural extension of the GARCH model

to a 2-equation system:

$$\begin{aligned} &\text{GARCH:} \\ &E[r_t^2|F_{t-1}^{LF}] := h_t^* = c_g + b_g h_{t-1}^* + a_g r_{t-1}^2 \\ &\text{HEAVY:} \\ &E[r_t^2|F_{t-1}^{HF}] := h_t = c_h + b_h h_{t-1} + a_h \epsilon_{t-1} \\ &E[v_t|F_{t-1}^{HF}] := m_t = c_m + b_m m_{t-1} + a_m v_{t-1} \end{aligned} \tag{7.17}$$

Whereas c_h, c_m, c_g, b_h, b_m, b_g, a_h, a_m, a_g denote constants that have to be estimated, v_t the realized measure at time t and h_t the conditional variance of return. F_{t-1}^{LF} denotes the filtration up to time $t-1$ where observations are made with a low frequency. (HF analogously is an abbreviation of high frequency). The distinction is the conditioning information set used in modelling the conditional variance of daily returns. Whereas HEAVY uses the lagged realized measure v_{t-1} to drive dynamics of h_t, GARCH uses the squared return. [Noureldin et al., 2012] develop a multivariate version by defining $R_{j,t}$ as a vector of returns at day t minute j and consequently the realized covariance matrix at dat t as:

$$RC_t = \sum_{j=1}^{m} R_{j,t} R_{j,t}' \tag{7.18}$$

And the HEAVY 2-equation model as:

$$\begin{aligned} E[P_t|F_{t-1}^{HF}] = E[R_t R_t'|F_{t-1}^{HF}] &:= H_t \\ E[V_t|F_{t-1}^{HF}] &:= M_t \end{aligned} \tag{7.19}$$

Or equivalently:

$$P_t = H_t^{1/2} \epsilon_t H_t^{1/2} V_t = M_t^{1/2} \eta_t M_t^{1/2} \tag{7.20}$$

Subsequently, many VARIMA parametrizations have been developed and estimated using quasi-maximum likelihood estimation methodologies with ϵ_t following a Wishart distribution.

7.2.2.2 Dynamic Correlation Multivariate Stocastic Volatility Models (DC MSV)

Previously the constant correlation multivariate stochastic volatility model (CC MSV), in order to adapt ARCH-based approaches to a multivariate setting, were introduced by [Harvey et al., 1994] (7.16). [Asai and McAleer, 2009] proposes an extension by directly modelling ϵ_t as a multivariate normal distribution $\epsilon_t \sim N(0, \Gamma_t)$:

$$\begin{aligned} \Gamma_t &= \tilde{Q}_t^{-1} Q_t \tilde{Q}_t^{-1\prime} \\ Q_{t+1} &= \Omega + \psi Q_t + \Xi_t \\ \Xi_t &\sim Wishart_k(v, \Lambda) \\ \tilde{Q}_t &= diagonal(Q_t)) \end{aligned} \tag{7.21}$$

Where the function "diagonal()" creates a diagonal matrix by setting the off-diagonal elements to be zero. The model is estimated by employing a Gibbs sampling scheme.

7.2.2.3 Multivariate Stochastic Volatility (MSVOL) models

[Philipov and Glickman, 2006] propose to use a Gibbs sampler to estimate a MSVOL model that is more flexible than previous multivariate SVOL models such as [Jacquier et al., 1994], [Harvey et al., 1994] and [Mahieu and Schotman, 1994]. Note that although [Harvey et al., 1994] was mentioned in the previous section as well, the emphasis here lies differently. The authors formulated it as:

$$\begin{aligned} y_t|\Sigma_t &\sim N(0,\Sigma_t) \\ \Sigma_t^{-1}|v,S_{t-1} &\sim Wishart_k(v,S_{t-1}) \\ S_t &= \frac{1}{v}\left(A^{1/2}\right)\left(\Sigma_t^{-1}\right)^d\left(A^{1/2}\right)' \end{aligned} \tag{7.22}$$

Whereas the matrix $\boldsymbol{A}$ defines the information about the intertemporal covariance relationships, d the overall strength of these relationships, k the size of the vectors and matrices and v the degrees of freedom for the Wishart distribution. As the Gibbs sampler was employed, priors for the hyperparemeters are necessary to be specified too.

7.2.2.4 Time Varying Realized Variance-Covariance matrices (RCOV)

[Jin and Maheu, 2009] propose a slightly modification of the two previous models and govern directly Σ_t rather than the precision, Σ_t^{-1}.

$$\begin{aligned} y_t|\Sigma_t &\sim N(0,\Sigma_t) \\ \Sigma_t|v,S_{t-1} &\sim Wishart_k(v,S_{t-1}) \\ S_t &= \frac{1}{v}\left(\Sigma_t^{d/2}\right)A\left(\Sigma_t^{d/2}\right)' \end{aligned} \tag{7.23}$$

The authors note however that the basic Wishart RCOV model as displayed has difficulty capturing persistency properties of speficially financial data. Inspired by the usage of the Heterogeneous AutoRegressive model (HAR) by [Corsi, 2009], amongst others, it develops the Wishart RCOV model with $K \geq 1$ components as:

$$\begin{aligned} \Sigma_t|v,S_{t-1} &\sim Wishart_k(v,S_{t-1}) \\ S_t &= \frac{1}{v}\Big[\prod_{j=K}^{1}\Gamma_{t,l_j}^{\frac{d_j}{2}}\Big]A\Big[\prod_{j=K}^{1}\Gamma_{t,l_j}^{\frac{d_j}{2}}\Big] \\ \Gamma_{t,l} &= \frac{1}{l}\sum_{i=0}^{l-1}\Sigma_{t-i} \\ 1 = l_1 &< ... < l_k \end{aligned} \tag{7.24}$$

The components are as such a sample average of past Σ_t raised to a different matrix power d_j. The component terms, $\Gamma_{t,l}$ allow for more persistence in the location of Σ_t while the d_j allow the effect thus to be amplified or dampenend.

7.2.2.5 The Wishart Autoregressive Process of Multivariate Stochastic Volatility

The Wishart Autoregressive (WAR) process is yet another methodology and framework that relies on more standard methods such as the method of moments and maximum likelihood.

The distribution of the process is defined by a markov process using the conditional Laplace transform that provides the conditional expectation of exponential affine transforms of elements of matrix Y_{t+1}, Ψ_t As such, the WAR(1)-process is given by

$$\Psi_t = E[expTr(\Gamma Y_{t+1}] = \frac{expTr[M'\Gamma(id - 2\Sigma\Gamma]^{-1}MY_t]}{[det(Id - 2\Sigma\Gamma]^k/2} \tag{7.25}$$

Where K denotes the degrees of freedom. It is right that the process can be extended to incorporate more lags of Y_t to a WAR(q) process. The paradigm differs of the others and is more closely related to continuous time implementations and stochastic calculus. It requires a large shift of paradigm with regard to the developed framework so far.

7.2.2.6 The Conditional Autoregressive Wishart model (CAW)

The conditional autoregressive Wishart model involves a central Wishart transition distribution with time-varying scale matrix that has a GARCH specification. It was first proposed by [Golosnoy et al., 2012] and recently generalized by [Yu et al., 2017]. The CAW(p,q)-model is described by:

$$\begin{aligned} R_t|F_{t-1} &\sim Wishart_k(v, S_t/v) \\ S_t &= CC' + \sum_{i=1}^{p} B_i S_{t-1} B_i' + \sum_{j=1}^{q} A_j R_{t-j} A_j' \end{aligned} \tag{7.26}$$

Note that the model thus can be seen as a state space model. Furthermore, the model as specified is unidentified and additional restrictions, similar to GARCH models as in [Bollerslev, 1986] require additional restrictions to allow the possibility of estimating the model. The general basic representation of the model allows for a multitude of extensions in the direction of GARCH-MIDAS (CAW-MIDAS), see [Engle et al., 2008], or HAR (CAW-HAR), see for example [Corsi, 2009]. These extensions are extensively discussed by [Golosnoy et al., 2012] as well. It estimates these models by employing a quasi-maximum likelihood methodology.

7.2.3 Decompositions

[Chiriac and Voev, 2011] decompose a series of covariance matrices into Cholesky factors before forecasting those Cholesky series using a suitable time-series model. From these models, the original covariance matrix can then be reconstructed. The cholesky decomposition of matrix A is given by:

$$A = LDL' \tag{7.27}$$

Where D is a diagonal matrix and L a lower triangular matrix. The reason why they do so is because it does not require imposing parameter restrictions on the model. Mos notably, it guarantees positive definiteness. But a nice implication is that covariance and correlation could be modelled independently in a sense.

The Cholesky decomposition is not the only used transformation of the covariance matrix in the literature. [Bauer and Vorkink, 2006] define a matrix-log-transformation and a matrix-exponential-transformation. They use these operations to be able to model the so-called log volatilities instead of the original ones.

7.3 How to proceed?

When opting for a certain model, there is thus enough to pick from. There exist extensions of the Dirichlet process into an additional time dimension, evolutionary clustering algorithms that minimize certain loss functions as well as econometric contributions in the form of multivariate stochastic volatility models, GARCH models, Wishart-dependent models and clever decompositions. This is however not an attempt to give an overview of all the methods that are out there. It is highly likely that there exists a large number of additional models ranging from submodels to completely unrelated models that could be used to describe the underlying dynamics at hand. It merely aims at giving an overview of my considerations that lead to the proposal of a new model as well as drawing inspiration from. That is, a model that allows fairly flexible for extending the Dirichlet process into the time dimension, without the necessity to completely overturn the inner workings of the clustering model as introduced in Chapter 5. A model that allows to take advantage of the collusive behaviour of underlying market forces by explicitly making use of the Dirichlet process, while adhering as much as possible to econometric and statistic theory as well as avoiding the trap of clustering subsequences of time series as described by [Keogh and Lin, 2005]. The next chapter develops an algorithm that attempts to do all of that. It is based on the framework introduced by [Windle et al., 2014], whereas the Dirichlet process is placed to fullfil the inner workings and implicitly apply a form of shrinking by clustering abstract dynamics.

8

Developing the Clustered Correlations Model

This chapter introduces the Clustered Correlations model. Whereas this thesis started out by exploring the possibilities of the Dirichlet process (Chapter 3) to model time dynamics in the form of the clustering model of Chapter 5, it appears that the results, as displayed in Chapter 6, are not promising. As such, Chapter 7 forms an exploration of the different directions that are out there that might reinvigorate the possibilities of the Dirichlet process as introduced in Chapter 3. Chapter 7 discusses several machine learning models that approach evolutionary clustering from a optimization perspective (7.1.2), as well as models that propose to augment the Dirichlet process of chapter 3 directly (7.1.3, 7.1.4). Finally, section 7.2 explains the reason for the variance-covariance approach and forms an overview of the models and developments of that field. These models are important to display what is out there, and where to build upon and implicitly set up criteria. As often stated, we are but dwarfs standing on the shoulders of giants.

8.1 The Variance Discounting Framework

One of the convenient things about the GARCH literature is that there is some sort of basic model representation that is used to derive further developments from. It is true that GARCH is well established and the literature surrounding this theme is vast as well. This common representation makes it easier to keep oversight of what is out there and to build upon that. Therefore it is important to first lay the foundations before building the model itself. This can be said as well for the structure of this chapter.

[Quintana and West, 1987] first introduced multivariate dynamic linear models that employed some form of variance discounting, but a rigorous justification in the form of Bayesian filtering for covariance matrices was given by the sister papers [Uhlig, 1994] and [Uhlig, 1997]. These models, and their relation to several models that were introduced in the previous section, are nicely generalized in [Windle et al., 2014]. Which I build upon to formalize a general framework. Denoting $\boldsymbol{r}_t$ as the vector of observations, I stick to the formulation of the state-space model in appendix A.1, but generalize these to the multivariate setting as in chapter 4 of [West, 1996],

among others:

$$\begin{aligned} \boldsymbol{r}_t &= \boldsymbol{Z}_t\boldsymbol{\alpha}_t + \boldsymbol{\epsilon}_t \qquad \text{with} \quad \boldsymbol{\epsilon_t} \sim g(\boldsymbol{H}_t) \\ \boldsymbol{\alpha}_{t+1} &= \boldsymbol{T}_t\boldsymbol{\alpha}_t + \boldsymbol{R}_t\boldsymbol{\eta}_t \quad \text{with} \quad \boldsymbol{\eta}_t \sim N(\boldsymbol{O}, \boldsymbol{Q}) \\ \boldsymbol{H}_t &= \boldsymbol{S}_t\boldsymbol{\Psi}_t\boldsymbol{S}_t' \qquad \text{where} \quad \boldsymbol{S}_t\boldsymbol{S}_t' \sim f(\boldsymbol{H}_{t-1}) \end{aligned} \tag{8.1}$$

Assuming $E[\boldsymbol{r}_t] = \boldsymbol{0}\ \forall t$. To obtain the model of [Uhlig, 1997], let g be a multivariate normal with covariance matrix $\boldsymbol{H}_t^{-1}$, f a deterministic function that assigns the upper cholesky factor of the matrix $\boldsymbol{H}_{t-1}$ to S_t (and the lower cholesky factor to $\boldsymbol{S}_t'$) and $\boldsymbol{\Psi}_t \sim \beta_m(n/2, 1/2)$ (where n is some integer which is at least equally large as the dimension of $\boldsymbol{r}_t$, m). The reason of the use of the multivariate beta-distribution lies in its well established conjugacy with the Wishart distribution. If we in turn start focusing on the squared observations instead, as well as leaving out the state space representation: $\boldsymbol{Y}_t = \boldsymbol{r}_t\boldsymbol{r}_t'$, the model becomes:

$$\begin{aligned} \boldsymbol{Y}_t &= \boldsymbol{\epsilon}_t\boldsymbol{\epsilon}_t' \qquad \text{with} \quad \boldsymbol{\epsilon_t} \sim g(\boldsymbol{H}_t) \\ \boldsymbol{H}_t &= \boldsymbol{S}_t\boldsymbol{\Psi}_t\boldsymbol{S}_t' \qquad \text{where} \quad \boldsymbol{S}_t\boldsymbol{S}_t' \sim f(\boldsymbol{H}_{t-1}) \end{aligned} \tag{8.2}$$

With g and f denoting some probability distribution. Assuming normality on $\boldsymbol{\epsilon}_t$ this could be extended to the model of [Harvey et al., 1994] and further to the model of [Asai and McAleer, 2009], as in the previous chapter.

Traditionally one would assume a normal-inverse-gamma distribution for $\boldsymbol{r}_t$. If I do the same here, the assumption would generalize the (inverse) gamma distribution for $\boldsymbol{\epsilon}$ to a (inverse) Wishart distribution:

$$\begin{aligned} \boldsymbol{Y}_t &\sim Wishart_m(k, (k\boldsymbol{H}_t)^{-1}) \\ \boldsymbol{H}_t &= \boldsymbol{S}_t\boldsymbol{\Psi}\boldsymbol{S}_t' \qquad \text{with} \quad \boldsymbol{S}_t\boldsymbol{S}_t^{-1} \sim f(\boldsymbol{H}_{t-1}) \end{aligned} \tag{8.3}$$

With m denoting the dimensionality of $\boldsymbol{Y}_t$ and k the degrees of freedom of the Wishart distribution. If we would let g be a multivariate normal with variance-covariance matrix $\boldsymbol{H}_t$, $f(\boldsymbol{H}_{t-1}) = \boldsymbol{A}^{1/2}\boldsymbol{H}_{t-1}^d\boldsymbol{A}^{1/2}$ and a Wishart distribution with a diagonal scale matrix for $\boldsymbol{\Psi}_t$ we obtain the model of [Philipov and Glickman, 2006]. Evidently, parametrization of the model above also allow us to obtain the other models that were introduced in the previous chapter. [Windle et al., 2014] derive the a convenient result that the model of [Uhlig, 1997] can be can be extended and derive closed form formulas for forward filtering, backward filtering and predicting one step in the future. They estimate the state space model:

$$\begin{aligned} \boldsymbol{Y}_t &\sim Wishart_m(k, (k\boldsymbol{H}_t)^{-1}) \\ \boldsymbol{H}_t &= \boldsymbol{S}_t\boldsymbol{\Psi}\boldsymbol{S}_t' \\ \boldsymbol{S}_{t-1} &= \text{upper cholesky factor } \boldsymbol{H}_{t-1} \\ \boldsymbol{\Psi}_t &\sim \beta_m(n/2, k/2) \end{aligned} \tag{8.4}$$

With n being a positive that is again at least as large as the dimensionality of $\boldsymbol{r}_t$.

8.2 The Clustered Correlations Model and its Estimation

8.2.1 Estimating the Correlation Matrices

Although the models are intrinsically intuitive and well defined on paper, most still struggle with scalability. Once it is required to model more than a handful of series using these models, they tend to break due to the vast amount of parameters associated with it. For example, as $\boldsymbol{Y}_t$ is a matrix of m by m variables, the total number of parameters associated with the covariance matrices $\boldsymbol{H}_t$ is $m \times m \times T$, with T denoting the number observations. In order to obtain a model with a significant smaller number of parameters, I propose to make use of a smart restriction mechanism, conform the previous chapters of this thesis. Instead of covariances directly, we can decompose the variance-covariance matrix into correlations and variances (somewhat similar to [Chiriac and Voev, 2011]). That is, into individual magnitudes and a measure of interconnectivity. As such, it is possible to model them independently. Conform the previous chapters, these correlations are assumed to be parametrized by a Dirichlet prior. In such a way we can cleverly shrink various correlations to be equal to each other and thus make sense of the parameter space. Define the model as:

$$\begin{aligned} \boldsymbol{H}_t &= diag(\sqrt{\boldsymbol{\Psi}_t})\boldsymbol{S}_t diag(\sqrt{\boldsymbol{\Psi}_t}) \\ \boldsymbol{S}_t &= DP\big(f_\alpha(\alpha), f_H(\boldsymbol{H}_{t-1})\big) \\ \boldsymbol{\Psi}_t &\sim igamma(n, n\sigma^2) \end{aligned} \tag{8.5}$$

n denotes the number of observations used to set up one time-window, whereas σ the sample variance of that set of observation entails. The literature is inconclusive with regard to the precise definition of those hyperparameters. It appears that they are often simply taken as being some fixed value and subsequently disregarded. However, I follow a common implementation that estimates those parameters from the data. These parametrizations require however some sort of realized variances approach. In practice, this implies that both the observations and the parameters of the inverse gamma distribution are constructed by using the information aggregated in one month. All the daily observations in that period of time are used to estimate the individual σ^2 and n is taken as the number of observations.

The Dirichlet Process is used to make simultaneously sense of the time-dimensions as well as the shrinkage. I further develop $\boldsymbol{S}_t$ by defining f_α and f_H, analogously to 5.13, in the following way:

$$\begin{aligned} \boldsymbol{S}_t &= DP(\alpha_t, G_0) \\ \alpha_t &= \hat{\alpha}_t + \epsilon_t \\ \hat{\alpha}_{t+1} &= \hat{\alpha_t} + \eta_{t+1} \\ \epsilon_t &\sim N(0, A_\epsilon) \\ \eta_{t+1} &\sim N(0, A_\eta) \end{aligned} \tag{8.6}$$

As such, I attempt to describe the underlying tendency to cluster as a state space model. It simultaneously allows to take into account previous information as well as to form a different prior distribution for each individual time-slot. In theory, this thus seems as an interesting

candidate to model this complexity. But in practice it appeared to be easier said than done. In order to estimate this system appropriately, it is necessary to adhere to the derivation of the Kalman filter as in appendix A.1. The notational difference between the two might be confusing at first, but, in fact, equations A.7 and A.8 only need to be translated to this model. The update step, A.8, becomes:

$$\begin{aligned} E[\hat{\alpha}_{t+1}|\alpha_t] &= T_t\hat{\alpha}_t + T_tP_tZ_t'(Z_tP_tZ_t' + A_\epsilon)^{-1}(\alpha_t - \hat{\alpha}_t) \\ &= \hat{\alpha}_t + P_t(P_t + A_\epsilon)^{-1}(\alpha_t - \hat{\alpha}_t) \end{aligned} \tag{8.7}$$

As Z_t and T_t in equation A.8 is equal to 1 $\forall t$ in 8.6, and H_t in A.8 is replaced by A_ϵ in order to avoid confusion.

For each start of simulations for a new step in time, it is thus not too complex to fetch the expected value of the observation and use it to simulate the value of the observation:

$$\alpha_t \sim N(\hat{\alpha}_t, A_\epsilon) \tag{8.8}$$

It is however slightly more complex to perform the updating step, A.7:

$$\begin{aligned} E[\hat{\alpha}_t|\alpha_t] &= \hat{\alpha}_t + P_tZ_t'(Z_tP_tZ_t' + A_\epsilon)^{-1}(\alpha_t - \hat{\alpha_t}) \\ &= \hat{\alpha}_t + P_t(P_t + A_\epsilon)^{-1}(\alpha_t - \hat{\alpha}_t) \end{aligned} \tag{8.9}$$

The problem lies in the determination of the value of the observation α_t. Precisely because the value is not exactly observed but has actually to be determined from the Gibbs sampler. In order to partially circumvent this problem, I propose to weight the simulated α_t's by the stability of the simulation of the Gibbs Sampler that resulted due to that specific α_t. As such, the weight of that α_t becomes the inverse of the sum of standard deviations of the simulated $\boldsymbol{\mu}_{\cdot,t}$, irrespective of the cluster $y_{i,t}$ was allocated to during the sampling procedure. As such, it weights predictability of the cluster-formation process, according to the distribution of $\mu_{i,t}$. If the distribution of $\mu_{i,t}$ is not too informative with regard to $y_{i,t}$, its standard deviation will be larger and thus will have a lower weight. In that case, it is probable that the label of the cluster, and hence the location of the cluster $y_{i,t}$ might differ quite a lot between individual sampling rounds. Vica versa, if the distribution of $\mu_{i,t}$ is condensed around a certain point, we obtain more certainty with regard to the location of y_i and consequently want to attach a higher weight to this. The algorithm is summarized in algorithm 7, where $\boldsymbol{\omega}$ denotes the vector of weights and $\hat{\alpha}_t$ is updated by:

$$\hat{\alpha}_t = \boldsymbol{\omega}_t^T\boldsymbol{\alpha}_t \tag{8.10}$$

Algorithm 7 Gibbs Sampling Estimation with α state space Implementation, only Correlation-matrices

Let T denote the total number of time observations
Let $ITER_{\alpha}$ denote the number of simulations for each α_t
Let $ITER_{DP}$ the number of simulations for each clustering given α
State space representation:

$$\begin{aligned} \alpha_t &= \hat{\alpha}_t + \epsilon_t \\ \hat{\alpha}_{t+} &= \hat{\alpha}_t + \eta_{t+1} \\ \epsilon_t &\sim N(0, A_\epsilon) \\ \eta_{t+1} &\sim N(0, A_\eta) \end{aligned} \tag{8.11}$$

$\hat{\alpha}_0 = \bar{\alpha}$
for $\forall t \leq T$ **do**
  for $\forall j \leq ITER_{\alpha}$ **do**
    Generate: $\alpha_{t,j} \sim N(\hat{\alpha}_{t-1}, A_\epsilon)$
    Let $\boldsymbol{z}_{t,j,0,.}$ denote an initial partition of observations into clusters
    Let $\boldsymbol{\mu}_{t,j,0,.}$ denote a vector of initial cluster means for each observation
    Let $\boldsymbol{V}_{t,j,0,.}$ be a matrix of initial cluster variances and covariances for each observation
    for $\forall i \leq ITER_{DP}$ **do**
      $\boldsymbol{\mu}_{t,j,i,.} = \boldsymbol{\mu}_{t,j,i-1,.}; \boldsymbol{V}_{t,j,i,.} = \boldsymbol{V}_{t,j,i-1,.}; \boldsymbol{z}_{t,j,i,.} = \boldsymbol{z}_{t,j,i-1,.}$
      for $\forall i' \leq$ NCorrelations **do**
        Set Up Probabilities
          $p_k = \sum_{j':z_{t,j,i,j'}=k} q_{j'}(\boldsymbol{y}_{t,i'}; \boldsymbol{\mu}_{t,j,i,j'}; \boldsymbol{V}_{t,j,i,j'})$
          $p_0 = \alpha_{t,j} q_0(s; \boldsymbol{S}; \boldsymbol{y}_{t,i'}; \boldsymbol{\mu}_{t,j,i-1,i'}; \tau)$
          $[p_0; p_1; ...; p_K] = \frac{1}{\sum_k p_k + p_0}[p_0; p_1; ...; p_K]$
        Determine Transition
          $z_{t,j,i,i'} \sim$ Categorical$(p = [p_0; p_1; ...; p_K])$
        Update parameters $\mu_{t,j,i,i'}$, $V_{t,j,i,i'}$
          if $z_{t,j,i,i'} == 1$ **then** (Assign it to a new cluster)
            $V_{t,j,i,i'} = V$
            $\mu_{t,j,i,i'} | \boldsymbol{V}_{t,j,i,i'} \sim N(\boldsymbol{y}_{t,i'}, \frac{\tau}{1+\tau}\boldsymbol{V}_{t,j,i,i'})$
          else (Assign it to an existing cluster)
            $m_k = \frac{1}{n_k}\sum_{j':z_{j'}=k} y_{j'}$
            $V_{t,j,i,i'} = V$
            $\mu_{t,j,i,i'} | \boldsymbol{V}_{t,j,i,i'} \sim N(\boldsymbol{m}_k, \frac{1}{1+\tau}\boldsymbol{V}_{t,j,i,i'})$
      $\omega_{t,j} = \frac{1}{\sum_{\forall i'} \sqrt{VAR(\mu_{t,j,.,i'})}}$ Storing and calculating results
  $S_{t,j,i} = \text{matrix}(\mu_{t,j,i,.})$ [1]
  $\hat{\alpha}_t = \boldsymbol{\omega}_{t,.}^T \boldsymbol{\alpha}_{t,.}$

Output: S

[1] Where matrix reshapes the vector $\mu_{t,j,i,.}$ back into a upper triagonal matrix. S collects all those matrices, either in the form of a list or an array. This holds for algorithm 8 as well.

8.2.2 Complete Estimation Algorithm

Using a inverse gamma prior for $\boldsymbol{\psi}$ to model the magnitude that corresponds to the correlation matrix, e.g. the variances themselves, it is possible to now complete the model and obtain distributions for the individual elements of the variance covariance matrix. For each Gibbs sampling round, the results are then multiplied by the variances to form a distribution. It is necessary to repeat this process for each simulation of the variance. This might massively increase the required space for the algorithm to operate. Nevertheless, if enough space is available, the results consist of distributions that are able to capture the dynamics in a Bayesian fashion. An overview of the complete algorithm is depicted in algorithm 8.

Here St, j, i denotes the matrix with individual elements $\mu_{t,j,i,i'}$ arranged such that they form the correlation matrix again. The array $\boldsymbol{H}$ contains the variance-covariance matrices for all different timewindows and underlying parameter simulations. That is, for all time t ($t \leq T$), for all simulated $\psi_{t'}$ ($t' \leq ITER_{\psi}$), for all simulated $\alpha_{t,j}$ ($j \leq ITER_{\alpha}$) all Gibbs sampling round ($i \leq ITER_{DP}$) and, last but not least, for all individual correlations ($i' \leq$NCorrelations). The size of $\boldsymbol{H}$ is thus $T \times ITER_{\psi} \times ITER_{\alpha} \times ITER_{DP} \times$(nr of variables).

Note that those variables come from the correlation matrix, but are vectorized in order to speed up the performance of the algorithm as well as preserve simplicity within the code. After every round of sampling where each variable is sampled exactly once, the results are reformulated back into a matrix again in order to perform the matrix operations that yield an estimate of the variance-covariance matrix ($\boldsymbol{H}$).

Algorithm 8 can be described compactly by: "sampling the individual variances for all time windows, sampling the correlations for all time windows and α's, and obtaining the covariance matrix by applying a reverse of the variance-correlation decomposition on all obtained simulations of those two elements".

A discussion of the results can be found in the next chapter, whereas the results themselves are attached as appendix D.

Algorithm 8 Complete Estimation Algorithm Description

Let T denote the total number of time observations
Let $ITER_{\alpha}$ denote the number of simulations for each α_t
Let $ITER_{DP}$ denote the number of simulations for each clustering given α
Let $ITER_{\psi}$ denote the number of simulation for each time period for ψ
State space representation:

$$\begin{aligned} \alpha_t &= \hat{\alpha}_t + \epsilon_t & \hat{\alpha}_{t+} &= \hat{\alpha}_t + \eta_{t+1} \\ \epsilon_t &\sim N(0, A_\epsilon) & \eta_{t+1} &\sim N(0, A_\eta) \end{aligned} \tag{8.12}$$

$\hat{\alpha}_0 = \bar{\alpha}$
for $\forall t \leq T$ **do**
  Estimation of Variances
  for $\forall t' \leq ITER_{\psi}$ **do**
    Obtain $\boldsymbol{\psi}_{t,t'} \sim igamma(n, n\sigma^2)$

  Estimation of Correlations
  for $\forall j \leq ITER_{\alpha}$ **do**
    Generate: $\alpha_{t,j} \sim N(\hat{\alpha}_{t-1}, A_\epsilon)$
    Let $\boldsymbol{z}_{t,j,0,.}$ denote an initial partition of observations into clusters
    Let $\boldsymbol{\mu}_{t,j,0,.}$ denote a vector of initial cluster means for each observation
    Let $\boldsymbol{V}_{t,j,0,.}$ be a matrix of initial cluster variances and covariances for each observation
    for $\forall i \leq ITER_{DP}$ **do**
      $\boldsymbol{\mu}_{t,j,i,.} = \boldsymbol{\mu}_{t,j,i-1,.}; \boldsymbol{V}_{t,j,i,.} = \boldsymbol{V}_{t,j,i-1,.}; \boldsymbol{z}_{t,j,i,.} = \boldsymbol{z}_{t,j,i-1,.}$
      for $\forall i' \leq$ NCorrelations **do**
        Set Up Probabilities
          $p_k = \sum_{j':z_{t,j,i,j'}=k} q_{j'}(\boldsymbol{y}_{t,i'}; \boldsymbol{\mu}_{t,j,i,j'}; \boldsymbol{V}_{t,j,i,j'})$
          $p_0 = \alpha_{t,j} q_0(s; \boldsymbol{S}; \boldsymbol{y}_{t,i'}; \boldsymbol{\mu}_{t,j,i-1,i'}; \tau)$
          $[p_0; p_1; ...; p_K] = \frac{1}{\sum_k p_k + p_0}[p_0; p_1; ...; p_K]$
        Determine Transition
          $z_{t,j,i,i'} \sim$ Categorical$(p = [p_0; p_1; ...; p_K])$
        Update parameters $\mu_{t,j,i,i'}$, $V_{t,j,i,i'}$
          if $z_{t,j,i,i'} == 1$ **then** (Assign it to a new cluster)
            $V_{t,j,i,i'} = V$
            $\mu_{t,j,i,i'} | \boldsymbol{V}_{t,j,i,i'} \sim N(\boldsymbol{y}_{t,i'}, \frac{\tau}{1+\tau}\boldsymbol{V}_{t,j,i,i'})$
          else (Assign it to an existing cluster)
            $m_k = \frac{1}{n_k}\sum_{j':z_{j'}=k} y_{j'}$
            $V_{t,j,i,i'} = V$
            $\mu_{t,j,i,i'} | \boldsymbol{V}_{t,j,i,i'} \sim N(\boldsymbol{m}_k, \frac{1}{1+\tau}\boldsymbol{V}_{t,j,i,i'})$

        Storing and calculating results
        $S_{t,j,i} =$ matrix$(\mu_{t,j,i,.})$
        for $\forall t' \leq ITER_{\psi}$ **do**
          $H_{t,t',j,i} = diag(\boldsymbol{\psi}_{t,.}) S_{t,j,i} diag(\boldsymbol{\psi}_{t,.})$
      $\omega_{t,j} = \frac{1}{\sum_{\forall i'} \sqrt{VAR(\mu_{t,j,.,i'})}}$
  Update Parameters
  $\hat{\alpha}_t = \boldsymbol{\omega}_{t,.}^T \boldsymbol{\alpha}_{t,.}$

Output: $\boldsymbol{H}, \boldsymbol{S}$

9

Clustered Correlations Model Results

9.1 Data Correlation Clustering Model

In order to preserve non-singularity, which is necessary for the cholesky decomposition, it is more convenient to aggregate daily data per month and calculate take the input as monthly variance-covariance matrices. Secondly, because estimation of the model requires heavy simulation and consequently thus quite a long runtime, it is convenient to test the model on a lower number of variables. At this stage of the game, I am foremost interest in the performance of the model. Therefore I make use of the smaller industry dataset of Fama en French. This one contains only 5 industries, namely Consumer, Manufacturing, Hi-tech, Health and Other. The data ranges from Juli the 1st, 1926 up to the last trading day of March 2018. Several days are missing, because generally stock exchanges are not open seven days a week. In order to restrict the size, I take the segment from January 2005 up until today. These daily observation are then aggregated into monthly variance-covariance matrix. Implying that we are left with 159 variance-covariance matrices. See figure 9.1. Note how strong the tendency to move together for these variables can be at times.

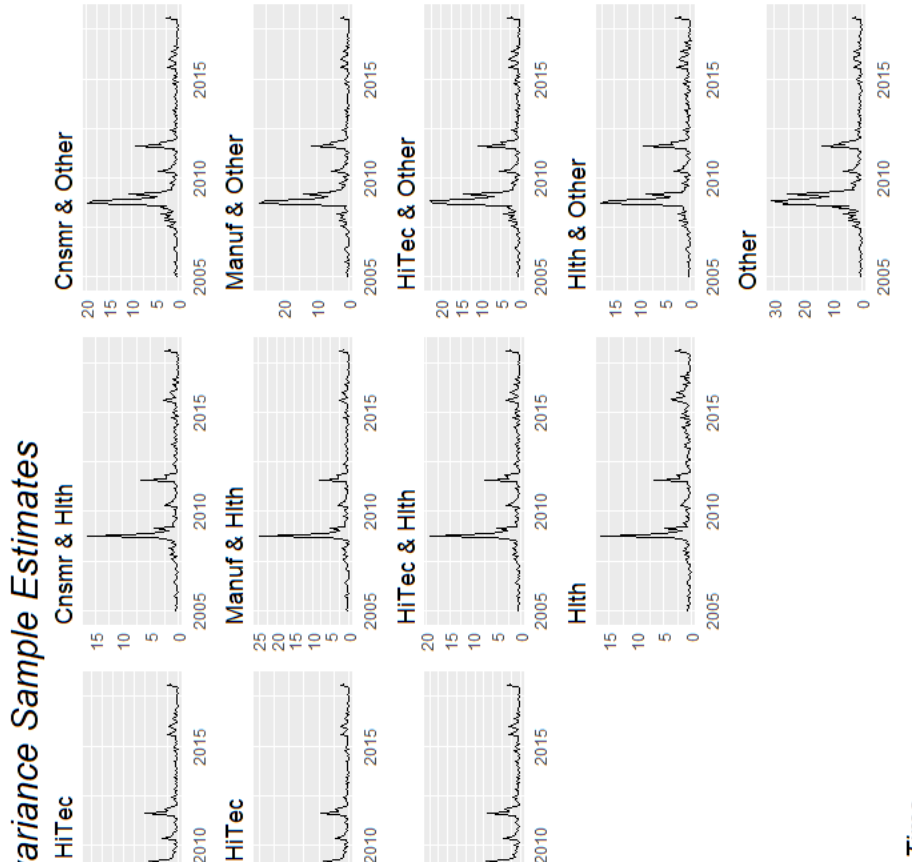

Fig. 9.1: Monthly Variance Covariance Matrix Estimates

9.2 Runtime of the Algorithm

The algorithm was run with the following settings for the parameters:

- $T = 159$
- $ITER_{\alpha} = 500$
- $ITER_{DP} = 500$
- $ITER_{\psi} = 500$
- 5 Variances
- $NCorrelations = 10$
- $A_{\epsilon} = 100$
- $A_{\eta} = 1$
- $\alpha_0 = 75$

On a laptop with a 2.8GHz CPU, 7.89 GB RAM a took the algorithm 5.35 days to run, while continuously using about 20% of the whole available capacity. Although such a runtime might be worrying, there is lot of efficiency in terms of the implementation to be gained. Therefore, the remaining part of section 10.2 discusses several ways to implement further parallelization as well as several ways to harness a greater amount of computing power.

9.3 Discussion of the Performance and Accuracy and Financial Interpretation

Appendix D documents both the estimated state space on α, as well as the upper triangular correlations matrix and the upper triangular covariance matrix. Theses results are the means of complex underlying distributions as estimated by algorithm 8. That is, the mean belief of each observation of the location of the cluster it is assigned to, aggregated over simulations of α and Gibbs Sampling rounds. When comparing the variance-covariance matrix results to the observations (the sample variance-covariance matrices), we see that the model overall tends to be a bit more modest (figure D3). The expectations tend to underestimate. Therefore, the prior distribution to estimate the individual variances might be a bit modest and probably performs better with appropriate retuning. However, due to the fact that magnitudes were seperated from interdynamics in this model, this should not have a profound effect on the estimated distributions dictating those correlations.

The correlations matrices over time are quite volatile, but tend to flock together nevertheless, implying that the Dirichlet process has had effect (figure D2). It is interesting to note that the results show that the consumer portfolio tends to show quite a high correlation with the other 4 portfolios, whereas the the rest of the correlations tend to be more modest. Floating around

zero and rarely showing a great tendency to correlate. The results do show however that the correlations between the portfolios went up in 2008, but in a explosive fashion. As such this does not support the hypothesis that correlations build up slowly over time. They respond quickly to perhaps unanticipated events that require traders and investor to rethink their own portfolios and strategies. The results do thus show that in 2008 correlations all went up, except the correlations between High-tech and Health, which is interesing. It is likely that Health portfolios were suddenly deemed to be more save than risky High-Tech portfolios. For the other correlations with health we see less of a pronounced effect of the crisis of 2008 as well. In terms of persistence of high correlations periods, it appears from the results that high volatilty, as well as high correlation tend to come in periods. Periods of strong volatility alternate with more calmer eras. A full inspection of the persistence of these series involves setting up a statistical test, nonetheless. See [Andersen and Bollerslev, 1997] for a detailed description of persistence. All in all, these results show that correlations theoretically could explain much more of the underlying dynamcis than covariance matrices alone. If there is one rule of thumb that can be derived from the results, would that be the observation that high correlations between portfolios go together with financial instability and could hence potentially be a strong indicator of bad weather ahead.

Finally, although A_ϵ is quite a lot larger than A_η, implying that the observation is far less stable than the actual observation, there does not seem to be an appropriate response to critical event in the state space model (figure D1). One would expect α and $\hat{\alpha}$ to decrease sufficiently to account for economies in recession in and after 2008. The common believe is that correlations tend towards 1 in times of crises, and this believe would imply thus that the tendency of clustering into few clusters large clusters would increase. Hence, one would expect the state space parameters to drop, as α dictates the likeliness that observations form their own cluster within the Gibbs sampling procedure. Note however that α does tend to increase significantly after 2010, indicating that the economy entered a more stable time where investors have more possibilities to diversify their portfolios. However, one would expect to see more 'action' in the graph around 2008. The magnitude of the effect is small nonetheless. This does not mean that the underlying distribution is completely off, but it does imply that the model could benefit from tuning of the underlying parameters. I expect that allowing more flexibility in higher layers, like the state space layer, while restricting the distributions of deeper layers could improve the effects. The choice to use a state space changes in α over time as well as to connect dynamics over time appeared to be not very satisfying.

It is very unfortunate that the runtime of the algorithm is compromised by the complexity of the model. In essence, one would want to run this model for several different parametrizations, as was done for the clustering algorithm before. The model contains various hyperparameters that ideally should be tested for other values as well. Next to that, it is not ideal in a Bayesian setting to be required to neglect large parts of the underlying distributions in order to visualize what is going on. The strongest part of this model might as well be those distributions, but that potential remains unharnessed.

10

Conclusion, Further Research and Further Development

10.1 Modelling and Quantification of the posterior

Although the Dirichlet process is promising for the development of Bayesian estimation of variance-covariance matrices, the results indicate that these models might be lacking in various aspects, including runtime, robustness and ease of visualisation. This is true for this model at least. The clustering part of the 2-step procedure (Chapter 5) shows that the Dirichlet might effectively assign observations to an unknown number of clusters, but its performance nevertheless strongly hinges upon perfectly-tuned parameters and hyperparameters. These problems translate directly to the Clustered Correlations model (Chapter 8). From the underlying distributions, one could in theory obtain all the information about the underlying complexity, but it is not straightforward to translate these complex distributions into useful insight. This is not necessary due to the model specification, but merely inherent in distributions over distributions. The information from the initial distributions is extremely large, but in order to gain an insight into the clustering labels that were applied during a full Gibbs sampling iteration, it was necessary to translate those simulated allocations into distributions of probabilities. Practitioners tend to use the last partition visited, but that neglects the whole simulation process. I use a Greedy MAP (section 5.3.3) translation mechanism, but that privileges the most likely allocations and hence disturbs the distribution as well. The difficulty to obtain clear characterizations of the resulting distributions make analyzing the performance, usefullness and robustness of both these two algorithms difficult as well. Interesting solutions for the formulation of MAPs from these posterior distribution lie in recent research like [Raykov et al., 2014] and [Broderick et al., 2013]. For the Clustered Correlations model, I therefore focus on the means of the cluster rather than the cluster label themselves. This already alleviates the problem, but does not necessarily make the distribution easier to visualize. In fact, in order to obtain a complete idea of the underlying distribution, it is necessary to inspect more profoundly the individual distributions for all in all timeslots, as well as their individual Gibbs sampling results. I have stored these in considerable pdfs entailing thousands of pages. They show that the simulations tend to be quite robust when times are stable, but become unpredictable once stock markets become more volatile. Naturally, these pdfs are available for further research and to practitioners interested in the inner workings of the model.

In general, the methodology as described in this thesis could benefit immensely from more efficient code and more computational power. In such settings, it becomes less time consuming to set the parameters by ratio or even trial and error. As such, although computational power has come a long way, there are still gains to be made. For the model to be practically relevant, it is even absolutely necessary that the runtime is decreased to a reasonable time.

Another promising direction for further research would be a better way to engineer the time-dimension. The Clustered Correlations Model (Chapter 8) implements α as a state space, but does use a shortcut to obtain updates of the observation. That is, a smart weighting of simulated αs according to the performance of the resulting simulation. It seems like a logical solution, but it is nevertheless not formulated or based on Bayesian statistics. An optimal solution should at least have (Bayesian) statistical validity. All in all, I hope that this thesis opens up new interesting research directions.

10.2 Estimation: Runtime Improvements

Because of the lack of robustness of the clustering algorithm introduced in chapter 5, it is likely that this true for Clustered Correlations model as well. As the (hyper)-parameterspace could be quite extensive, combined with the displayed subtlety required to setting hyperparameters for such models, for this model to be efficiency further investigated it is necessary to improve the runtime. I ran the algorithm thus basically under the standard R settings. At this stage of development of the algorithm, these statistical techniques and inspecting their usability, the algorithm was implemented with accuracy in mind rather than speed. As such, it is without question that the runtime could be improved in the future. First of all, the loop that employs the Gibbs sampling procedure should be parallelized. That is the loop starting at line 12 in algorithm 7 and line 18 of algorithm 8. The other parts of the algorithm require being processed sequentially. Without being able to estimate $\hat{\alpha}_t$ we cannot proceed to the next loop where that parameter is used to sample $\alpha_{t,j}$. However, given a certain $\hat{\alpha}_{t-1}$, observations $y_{t,.}$, simulated $\boldsymbol{\psi}$ and initial values for $\boldsymbol{z}$, $\boldsymbol{\mu}$ and $\boldsymbol{V}$, the separate Gibbs Sampling procedures do not depend on each other and could be executed in parallel. Note that the draws of $\boldsymbol{\psi}$ could also be executed in parallel for the individual elements, as well as α_t. Despite the fact that these particular parts could be performed in parallel, the algorithm requires that they are connected sequentially. An overview of the resulting pipeline, including the parts with appropriate parallelization, is given in appendix E.

The ever increasing demand for more computing power, fuelled by novel algorithms and enormous quantities of data, as well as the possibilities offered to consumers on terms of computing power, have given rise to a plethora of options to do so. Cloud computing services offered by Amazon (AWS), Microsoft (Azure) and Google (Google Cloud) offer massive parallelization using a multitude of cores. If saving money is an issue, one could also rely on excellent products that attempt to make the most out of your laptop. GPU's can switched on for certain tasks, either by simply augmenting the code of by clicking, installing and configurating the appropriate

software. For example, the laptop that I make use comes preinstalled with several utilies of the new NVIDIA CUDA packages. Next to that, it is known that C++, python and most notably Spark have much more ease and capabilities in terms of paralllel computating compared to R. Finally, there is some sort of middle ground where one has the possibility to make use of multiple cores without actually having to buy computational power from Amazon, Google or Microsoft. In such cases, it is possible to set up multiple environment where parts of the could run and an orchestrator that manages the available resources and environments. Nowadays one is not obliged to make use of virtual machines, but instead one can make use of what are called dockers to create mildly unstable but temporary virtual environments. Those dockers can then be created, managed and removed by applications such as Kubernetes, which came out of google's BORG project. Fortunately, it has been giving to the public as it is opensourced nowadays.

A

Proofs and Derivations

A.1 Kalman Filter

Lemma A.1.1 *Suppose:*

$$E\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$$
$$Var\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \Sigma_{x,x} & \Sigma_{x,y} \\ \Sigma_{x,y} & \Sigma_{y,y} \end{bmatrix}$$

Then:

$$E[x|y] = \mu_x + \Sigma_{x,y}\Sigma_{y,y}^{-1}(y - \mu_y)$$
$$Var[x|y] = \Sigma_{x,x} - \Sigma_{x,y}\Sigma_{y,y}^{-1}\Sigma_{x,y}$$

Lemma A.1.2 *These results hold whether (x, y) is normally distributed or not. (MVLUE: minimal variance linear unbiased estimator property)*

Lemma A.1.3 *The posterior mean and variance matrix follow similarly:*

$$P[x|y] = \frac{P(x,y)}{P(y)} = \frac{P[y|x]P[x]}{P[y]} \tag{A.1}$$

Lemma A.1.4 *The estimate $\hat{x}|y$ is a minimum variance linear posterior mean estimate (MVLPME)*

Define the state space model as:

$$\begin{aligned} y_t &= Z_t\alpha_t + e_t \\ a_{t+1} &= T_t\alpha_t + R_t\eta_t \\ e_t &\sim N(0, H) \\ \eta_t &\sim N(0, Q) \\ \alpha_t &\sim N(a_1, P_1) \end{aligned} \tag{A.2}$$

With Notation:

$$\begin{aligned} a_{t|t} &= E[\alpha_t|y_t] \\ a_{t+1} &= E[\alpha_{t+1}|y_t] \\ P_{t|t} &= Var[\alpha_t|y_t] \\ P_{t+1} &= Var[\alpha_{t+1}|y_t] \\ v_t &= y_t - E[Z_t\alpha_t + \epsilon_t|y_{t-1}] \end{aligned} \tag{A.3}$$

Applying Lemma A.1.1, we can derive the update step:

$$\begin{aligned} a_{t|t} &= E[\alpha_t|Y_{t-1}] + cov(\alpha_t, v_t)[var(v_t)]^{-1}v_t \\ P_{t|t} &= var[\alpha_t] + cov(\alpha_t, v_t)\left[var(v_t)\right]^{-1}cov(\alpha_t, v_t) \end{aligned} \tag{A.4}$$

With:

$$\begin{aligned} cov(\alpha_t, v_t) &= E[\alpha_t(y_t - Z_t a_t)'|y_{t-1}] \\ &= E[\alpha_t(Z_t\alpha_t + \epsilon_t - Z_t a_t)|y_{t-1}] \\ &= Z_t E[\alpha_t^2 - \alpha_t a_t]Z_t' \\ &= P_t Z_t' \end{aligned} \tag{A.5}$$

$$Var[v_t] = Var[Z_t\alpha_t + \epsilon_t - Z_t a_t] = Z_t P_t Z_t' + H_t = F_t \tag{A.6}$$

Thus:

$$\begin{aligned} a_{t|t} &= a_t + P_t Z_t'(Z_t P_t Z_t' + H_t)^{-1}v_t \\ P_{t|t} &= P_t - P_t Z_t'[Z_t P_t Z_t' + H_t]^{-1}Z_t P_t' \end{aligned} \tag{A.7}$$

Then, Lemma A.1.2 states that this is the best estimate of a_t and P_t given the information available up to time t. Next, we can use this to derive the forecast step:

$$\begin{aligned} E[\alpha_{t+1}|y_t] &= T_t E[\alpha_t|y_t] + R_t E[\eta_t] \\ &= T_t E[\alpha_t|y_t] \\ &= T_t a_t + T_t P_t Z_t'[Z_t P_t Z_t' + H_t]^{-1}v_t \\ &= T_t a_t + K_t v_t \end{aligned} \tag{A.8}$$

Where $K_t = T_t P_t Z_t' F_t^{-1}$ is the Kalman gain and $F_t^{-1} = [Z_t P_t Z_t' + H_t]^{-1}$

$$\begin{aligned} P_{t+1} &= Var[T_t\alpha_t + R_t\eta_t|y_t] \\ &= T_t P_{t|t} T_t' + R_t Q_t R_t' \\ &= T_t(P_t - P_t Z_t' F_t^{-1} Z_t P_t)T_t' + R_t Q_t R_t' \\ &= T_t P_t(T_t - K_t Z_t)' + R_t Q_t R_t \end{aligned} \tag{A.9}$$

From Lemma 3 and 4 we can than deduce that these results hold under non-normality of (α_t, v_t) as well.

A.2 Deriving the Dirichlet Process

Both [Ferguson, 1973] and [Blackwell and MacQueen, 1973] show that the Dirichlet process arises naturally from a finite mixture model characterized by the Dirichlet distribution. The Dirichlet distribution with K dimensions is characterized by probability density function:

$$f(x_1, .., x_K; \alpha_1, ..., \alpha_K) = \frac{1}{B(\boldsymbol{\alpha}} \prod_{i=1}^{K} x_i^{\alpha_i - 1} \tag{A.10}$$

With:

$$B(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)} \tag{A.11}$$

With α_i as the mixing proportions, which are naturally constrained to be positive and sum to 1. A nice property of the Dirichlet distribution is the fact that it is the conjugate prior of the categorical distribution and multinomial distribution. Denote N as the total number of observations and K as the total number of clusters:

$$\begin{aligned} P[\boldsymbol{\theta}|\boldsymbol{\alpha}] &\sim Dirichlet(\alpha) \\ P[z_i|\boldsymbol{\theta}] &\sim multinomial(\boldsymbol{\theta}) \end{aligned} \tag{A.12}$$

Then the joint distribution is:

$$\begin{aligned} P[\boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{\alpha}] &= P[\boldsymbol{\theta}|\alpha] \times \prod_{i=1}^{n} P[z_i|\boldsymbol{\theta}] \\ &= \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \times \frac{N!}{\prod_{k=1}^{K} n_k!} \prod_{k=1}^{K} \theta_k^{n_k} \end{aligned} \tag{A.13}$$

Where:

$$n_k = \sum_{i=1}^{N} \mathbb{1}(z_i = k) \tag{A.14}$$

Then we may integrate out the mixing proportions and obtain the marginal of $\boldsymbol{z}$ given $\boldsymbol{alpha}$:

$$\begin{aligned} P[\boldsymbol{z}|\boldsymbol{\alpha}] &= \int p[\boldsymbol{z}|\boldsymbol{\theta}] P[\boldsymbol{\theta}|\boldsymbol{\alpha}] d\boldsymbol{\theta} \\ &= \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \frac{N!}{\prod_{k=1}^{K} n_k!} \int \prod_{k=1}^{K} \theta_k^{n_k + \alpha_k - 1} d\theta_k \\ &= \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \frac{N!}{\prod_{k=1}^{K} n_k!} \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + n_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k + n_k)} \end{aligned} \tag{A.15}$$

To see this, note that we can use the fact that each proper distribution sums to 1 by definition:

$$\begin{aligned} \int P[\boldsymbol{\theta}|\boldsymbol{\alpha}] &= 1 \\ \int \frac{\Gamma(\sum_{k=1}^{K} \alpha_k + n_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} d\boldsymbol{\theta} &= 1 \\ \int \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} d\boldsymbol{\theta} &= \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k)} \end{aligned} \tag{A.16}$$

Then simply replace the parameters α_k by $n_k + \alpha_k$

However, a slightly different view on the same model is derived from the categorical distribution. The difference is slight, but very outspoken. To derive its properties, slightly rewrite the model as:

$$\begin{aligned} P[\boldsymbol{\theta}|\boldsymbol{\alpha}] &\sim Dirichlet(\alpha) \\ P[z_i|\boldsymbol{\theta}] &\sim categorical(\boldsymbol{\theta}) \end{aligned} \tag{A.17}$$

Then the joint distribution is:

$$\begin{aligned} P[\boldsymbol{\theta}, \boldsymbol{z}|\boldsymbol{\alpha}] &= P[\boldsymbol{\theta}|\alpha] \times \prod_{i=1}^{n} P[z_i|\boldsymbol{\theta}] \\ &= \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \times \prod_{k=1}^{K} \theta_k^{n_k} \end{aligned} \tag{A.18}$$

It follows that the marginal of $\boldsymbol{z}$ given $\boldsymbol{\alpha}$ analogously is:

$$P[\boldsymbol{z}|\boldsymbol{\alpha}] = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + n_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k + n_k)} \tag{A.19}$$

This can be used to derive mathematically the convenience of the Dirichlet process. To demonstrate the further line of thought, take the probability that $z_N = 1$. E.g. observation N belongs to cluster $k = 1$. Condition $z_N = 1$ on the other z_i s.t. $i < N$. We can calculate this by:

$$P[z_N = 1|z_1, ..., z_{N-1}, \boldsymbol{\alpha}] = \frac{P[z_1, ..., z_N = 1|\boldsymbol{\alpha}]}{P[z_1, ..., z_{N-1}|\boldsymbol{\alpha}]} \tag{A.20}$$

Then we obtain:

$$\begin{aligned} P[z_N = 1|z_1, ..., z_{N-1}, \boldsymbol{\alpha}] &= \frac{\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + n_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k + n_k)}}{\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \frac{\Big(\prod_{k \neq 1} \Gamma(\alpha_k + n_k)\Big) \Gamma(\alpha_1 + n_1 - 1)}{\Gamma\Big((\sum_{k \neq 1} \alpha_k + n_k) + \alpha_1 + n_1 - 1\Big)}} \\ &= \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + n_k)}{\Big(\prod_{k \neq 1} \Gamma(\alpha_k + n_k)\Big) \Gamma(\alpha_1 + n_1 - 1)} \frac{\Gamma\Big((\sum_{k \neq 1} \alpha_k + n_k) + \alpha_1 + n_1 - 1\Big)}{\Gamma(\sum_{k=1}^{K} \alpha_k + n_k)} \\ &= \frac{\Gamma(\alpha_1 + n_1)}{\Gamma(\alpha_1 + n_1 - 1)} \frac{\Gamma(N + \sum_{k=1}^{K} \alpha_k - 1)}{\Gamma(N + \sum_{k=1}^{K} \alpha_k)} \\ &= \frac{\alpha_1 + n_1}{N + \sum_{k=1}^{K} \alpha_k} \end{aligned} \tag{A.21}$$

Taking into account that after fixing z_N, it is no longer stochastic variable. This implies that we should replace N by $N - 1$ in the equation. From here it should be clear that:

$$P[z_N = k|z_1, ..., z_{N-1}, \boldsymbol{\alpha}] = \frac{\alpha_j + n_k}{N + \sum_{k=1}^{K} \alpha_k - 1} \tag{A.22}$$

Now, if we condition this $\alpha_i = \alpha_j = \frac{\sum_{k=1}^{K} \alpha_k}{K} \; \forall i, j$.Then I obtain:

$$P[z_N = k | z_1, ..., z_n, \boldsymbol{\alpha}] = \frac{\frac{\sum_{k=1}^{K} \alpha_k}{K} + n_k}{N + \sum_{k=1}^{K} \alpha_k - 1} \tag{A.23}$$

Then, following [Rasmussen, 2000], let $K \to \infty$:

$$P[z_N = k : n_K \geq 1 | z_1, ..., z_n, \boldsymbol{\alpha}] = \frac{n_k}{N + \sum_{k=1}^{K} \alpha_k - 1} \tag{A.24}$$

From the law total probability it has to be that:

$$P[z_n = k : n_k = 0 | z_1, ..., z_n, \boldsymbol{\alpha}] = \frac{\sum_{k:n_k=0} \frac{\sum_{k=1}^{K} \alpha_k}{K}}{N + \sum_{k=1}^{K} \alpha_k - 1} = \frac{\tilde{\alpha}}{N + \sum_{k=1}^{K} \alpha_k - 1} \tag{A.25}$$

At first, it seems counter-intuitive that one is able to let the fraction $\frac{\sum_{k=1}^{K} \alpha_k}{K} \to 0$ for any $k : n_K \geq 1$. However, if we fix N, it will always be the case that the number of unoccupied clusters will grow. Even more, the number of occupied clusters is bounded above by N and the ratio of occupied clusters over unoccupied clusters will tend to zero:

$$\lim_{K \to \infty} \frac{N}{K} \to 0 \tag{A.26}$$

The same reasoning also implies that:

$$lim_{k \to \infty} \tilde{\alpha} \to \sum_{k=1}^{K} \alpha_k \tag{A.27}$$

As such, to fully derive the Dirichlet process we simply need to complete the derivation above by a distribution that defines a cluster when it is occupied, G_0. Then we obtain:

$$\begin{aligned} P[y_N | z_N] P[z_N | \boldsymbol{z}_{-N}, \boldsymbol{\alpha}] &= P[y_N | z_N = k : n_k = 0] P[z_N = k : n_k = 0 | \boldsymbol{z}_{-N}, \boldsymbol{\alpha}] \\ &+ P[y_N | z_N = k : n_K \geq 1] P[z_N = k : n_K \geq 1 | \boldsymbol{z}_{-N}, \boldsymbol{\alpha}] \\ &= \frac{\tilde{\alpha}}{N + \sum_{k=1}^{K} \alpha_k - 1} G_0 + \sum_{k=1}^{K} \frac{n_k}{N + \sum_{k=1}^{K} \alpha_k - 1} G_k \end{aligned} \tag{A.28}$$

Which in turns allows for a more dense representation as:

$$P[y_N | z_N] P[z_N | \boldsymbol{z}_{-N}, \boldsymbol{\alpha}] \propto \bar{\alpha} G_0 + \sum_{k=1}^{K} n_k G_k \tag{A.29}$$

In order to complement the literature, compare this for example to equation 5) in [Escobar and West, 1995]:

$$(Y_{n+1} | \boldsymbol{\pi}) \sim \alpha a_n T_s(m, M) + a_n \sum_{j=1}^{k} n_j N(\mu_j^*, V_j^*) \tag{A.30}$$

Where:

$$a_r = \frac{1}{\alpha + r} \tag{A.31}$$

A.3 Mixture Proportions Conditionals

Finally, we can make use of the marginal in A.16 to derive the conditional $P[\boldsymbol{\theta}|\boldsymbol{z},\boldsymbol{\alpha}]$, for the categorical model with Dirichlet prior. Rewrite:

$$\frac{\prod_k \Gamma(\alpha_k)}{\Gamma(\sum_k \alpha_k)} = P[\boldsymbol{z}|\boldsymbol{\alpha}]\frac{\prod_k \Gamma(\alpha_k + n_k)}{\Gamma(\sum_k \alpha_k + n_k)} \tag{A.32}$$

And replace:

$$\begin{aligned} P[\boldsymbol{\theta},\boldsymbol{z}|\boldsymbol{\alpha}] &= P[\boldsymbol{z}|\boldsymbol{\alpha}]P[\boldsymbol{\theta}|\boldsymbol{z},\boldsymbol{\alpha}] \\ &= P[\boldsymbol{z}|\boldsymbol{\alpha}]\frac{\Gamma(\sum_k \alpha_k + n_k)}{\prod_k \Gamma(\alpha_k + n_k)}\prod_{k=1}^{K}\theta_k^{\alpha_k - 1} \times \prod_{k=1}^{K}\theta_k^{n_k} \end{aligned} \tag{A.33}$$

Rewriting the products then gives the conditional distribution as:

$$P[\boldsymbol{\theta}|\boldsymbol{z},\boldsymbol{\alpha}] = \frac{\Gamma(\sum_k \alpha_k + n_k)}{\prod_k \Gamma(\alpha_k + n_k)}\prod_{k=1}^{K}\theta_k^{\alpha_k + n_k - 1} \tag{A.34}$$

Which follows a Dirichlet$(\alpha_1 + n_1, ..., \alpha_K + n_K)$-distribution. Now in order to obtain the conditional probability for a single θ_i, first break it down to a lower-dimensional setting. Take $K = 3$:

$$P[\theta_1,\theta_2,\theta_3|\boldsymbol{z},\boldsymbol{\alpha}] = \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + N)}{\prod_{k=1}^{3}\Gamma(\alpha_k + n_k)}\left(\theta_1^{\alpha_1 + n_1 - 1}\theta_2^{\alpha_2 + n_2 - 1}\theta_3^{\alpha_3 + n_3 - 1}\right) \tag{A.35}$$

Then in order to obtain the conditional of θ_1:

$$\begin{aligned} &P[\theta_1|\theta_2,\theta_3,\boldsymbol{\alpha},\boldsymbol{z}] = \int_0^1\int_0^1 P[\theta_1,\theta_2,\theta_3|\boldsymbol{\alpha},\boldsymbol{z}]d\theta_2 d\theta_3 \\ &= \int_0^{1-\theta_1}\frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + N)}{\prod_{k=1}^{3}\Gamma(\alpha_k + n_k)}\left(\theta_1^{\alpha_1 + n_1 - 1}\theta_2^{\alpha_2 + n_2 - 1}(1 - \theta_2 - \theta_2)^{\alpha_3 + n_3 - 1}\right)d\theta_2 \\ &= \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + N)}{\prod_{k=1}^{3}\Gamma(\alpha_k + n_k)}\theta_1^{\alpha_1 + n_1 - 1}\int_0^{1-\theta_1}\theta_2^{\alpha_2 + n_2 - 1}(1 - \theta_1 - \theta_2)^{\alpha_3 + n_3 - 1}d\theta_2 \\ &\qquad (\text{Set: } \theta_2 = (1 - \theta_1)u) \\ &= \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + N)}{\prod_{k=1}^{3}\Gamma(\alpha_k + n_k)}\theta_1^{\alpha_1 + n_1 - 1}\left(1 - \theta_1^{\alpha_2 + \alpha_3 + n_2 + n_3 - 1}\right)\int_0^1 u^{\alpha_2 + n_2 - 1}(1 - u)^{\alpha_3 + n_3 - 1}du \\ &= \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + n_1 + n_2 + n_3)}{\Gamma(\alpha_1 + n_1)\Gamma(\alpha_2 + n_2)\Gamma(\alpha_3 + n_3)}\frac{\Gamma(\alpha_2 + n_2)\Gamma(\alpha_3 + n_3)}{\Gamma(\alpha_2 + \alpha_3 + n_2 + n_3)}\theta_1^{\alpha_1 + n_1 - 1}(1 - \theta_1)^{\alpha_2 + \alpha_3 + n_2 + n_3 - 1} \\ &= \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + n_1 + n_2 + n_3)}{\Gamma(\alpha_1 + n_1)\Gamma(\alpha_2 + \alpha_3 + n_2 + n_3)}\theta_1^{\alpha_1 + n_1 - 1}(1 - \theta_1)^{\alpha_2 + \alpha_3 + n_2 + n_3 - 1} \end{aligned} \tag{A.36}$$

Where I used that the beta function can be transformed to an integral over polynomials ([Krantz, 2012]):

$$B(m,n) = \frac{(m-1)!(n-1)!}{(m+n-1)!} = \int_0^1 u^{m-1}(1-u)^{n-1}du \tag{A.37}$$

It follows that:

$$P[\theta_1|\theta_2,\theta_3,\boldsymbol{\alpha},\boldsymbol{z}] \sim \text{Beta}(\alpha_1 + n_1, \alpha_2 + \alpha_3 + n_2 + n_3) \tag{A.38}$$

Furthermore, this result can be extended to higher dimensions such that the conditional becomes:

$$P[\theta_i|\boldsymbol{\theta}_{-i}, \boldsymbol{\alpha}, \boldsymbol{z}] \sim \text{Beta}(\alpha_1 + n_1, \sum_{j \neq i} \alpha_j + n_j) \tag{A.39}$$

Using these results for the conditionals, we are then able to implement the Gibbs sampling methodology.

Note that the results are again easily extendable to the multinomial distribution with a Dirichlet prior. Then take again:

$$P[\boldsymbol{\theta}|\boldsymbol{z}, \boldsymbol{\alpha}] = \frac{\Gamma(\sum_k \alpha_k + n_k)}{\prod_k \Gamma(\alpha_k + n_k)} \frac{N!}{\prod_{k=1}^K n_k!} \prod_{k=1}^K \theta_k^{\alpha_k + n_k - 1} \tag{A.40}$$

Evidently, for $K = 3$:

$$P[\theta_1|\theta_2, \theta_3, \boldsymbol{z}, \boldsymbol{\alpha}] = \frac{N!}{\prod_{k=1}^3 n_k!} \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3 + n_1 + n_2 + n_3)}{\Gamma(\alpha_1 + n_1)\Gamma(\alpha_2 + \alpha_3 + n_2 + n_3)} \theta_1^{\alpha_1 + n_1 - 1}(1 - \theta_1)^{\alpha_2 + \alpha_3 + n_2 + n_3 - 1} \tag{A.41}$$

A.4 Bayesian Estimation of the Kalman Filter

[Geweke and Tanizaki, 2001] proposes a intuitive and concise way to estimate state-space models following the Bayesian paradigm. In order to describe how to go about this, I use the notation introduced in section A.1 and follow the steps in [Geweke and Tanizaki, 2001]. First define $A_t = (\alpha_0, ..., \alpha_t)$, let T denote the total number of timesteps. Denote $Y_t = (y_1, ..., y_t)$. Finally, denote the parameters with a tilde in a similar fasion like $\tilde{Z}_t = (Z_1, ..., Z_t)$.The joint distribution of the measurements and the states are then:

$$P[A_T, Y_T|\tilde{Z}_T, \tilde{T}_T, \tilde{H}_T, \tilde{Q}_T, \tilde{R}_T] = P_y[Y_T|A_T, \tilde{Z}_T, \tilde{H}_T]P_\alpha[A_T|\tilde{T}_T, \tilde{R}_T, \tilde{Q}_T]P[\tilde{Z}_T, \tilde{T}_T, \tilde{H}_T, \tilde{Q}_T, \tilde{R}_T] \tag{A.42}$$

For simplicity, take the hyperparameters as being fixed for the moment over time and deterministic. Because the number of time steps was defined to be T, denote a fixed and deterministic matrix $\tilde{T}_T$ as $\bar{T}$ Then the joint distribution collapses into:

$$P[A_T, Y_T|\bar{Z}, \bar{T}, \bar{H}, \bar{Q}, \bar{R}] = P_y[Y_T|A_T, \bar{Z}, \bar{H}]P_\alpha[A_T|\bar{T}, \bar{R}, \bar{Q}] \tag{A.43}$$

Where the two densities of the right hand are represented by:

$$\begin{aligned} P_\alpha[A_T|\bar{T}, \bar{R}] &= \begin{cases} P_\alpha[\alpha_0|\bar{T}, \bar{R}] \prod_{t=1}^{T} P_\alpha[\alpha_t|\alpha_{t-1}, \bar{T}, \bar{R}] & \text{if } \alpha_0 \text{ is stochastic} \\ \prod_{t=1}^{T} P_\alpha[\alpha_t|\alpha_{t-1}, \tilde{T}_T, \tilde{R}_T] & \text{otherwise} \end{cases} \\ P_y[Y_T|A_T, \bar{Z}, \bar{H}] &= \prod_{t=1}^{T} P_y[y_t|\alpha_t, \bar{Z}, \bar{H}] \end{aligned} \tag{A.44}$$

For the model as defined in equation A.2 these probabilities are defined as:

$$\begin{aligned} P_\alpha[\alpha_t|\alpha_{t-1}, \bar{T}, \bar{R}] &= (2\pi)^{-k_\alpha/2}|\bar{T}P_{t-1}\bar{T}' + \bar{R}\bar{Q}\bar{R}'|^{-1} e^{-\frac{1}{2}\left(\alpha_t - \bar{T}\alpha_{t-1}\right)'\left((\bar{T}P_{t-1}\bar{T}' + \bar{R}\bar{Q}\bar{R}')^{-1}\right)\left(\alpha_t - \bar{T}\alpha_{t-1}\right)} \\ P_y[y_t|\alpha_t, \bar{Z}, \bar{H}] &= (2\pi)^{-k_y/2}|\bar{Z}P_{t-1}\bar{Z}' + \bar{H}|^{-1} e^{-\frac{1}{2}\left(y_t - \bar{Z}\alpha_t\right)'\left((\bar{Z}P_{t-1}\bar{Z}' + \bar{H})^{-1}\right)\left(y_t - \bar{Z}\alpha_t\right)} \end{aligned} \tag{A.45}$$

With k_α and k_y denoting the dimensionality of α and y, respectively. However, it is known that in case of fixed and deterministic hyperparameters, P_t quickly converges to a stable fixed value. Suppose we know that value and set our initial value P_0 to it, these equations become more handlable as:

$$\begin{aligned} P_\alpha[\alpha_t|\alpha_{t-1}, \bar{T}, \bar{R}] &= (2\pi)^{-k_\alpha/2}|\bar{T}P_0\bar{T}' + \bar{R}\bar{Q}\bar{R}'|^{-1} e^{-\frac{1}{2}\left(\alpha_t - \bar{T}\alpha_{t-1}\right)'\left((\bar{T}P_0\bar{T}' + \bar{R}\bar{Q}\bar{R}')^{-1}\right)\left(\alpha_t - \bar{T}\alpha_{t-1}\right)} \\ P_y[y_t|\alpha_t, \bar{Z}, \bar{H}] &= (2\pi)^{-k_y/2}|\bar{Z}P_0\bar{Z}' + \bar{H}|^{-1} e^{-\frac{1}{2}\left(y_t - \bar{Z}\alpha_t\right)'\left((\bar{Z}P_0\bar{Z}' + \bar{H})^{-1}\right)\left(y_t - \bar{Z}\alpha_t\right)} \end{aligned} \tag{A.46}$$

Then A.43 comes down to:

$$
\begin{aligned}
P_y[Y_T|A_T,\bar{Z},\bar{H}]P_\alpha[A_T|\bar{T},\bar{R},\bar{Q}] &= \prod_{t=1}^{T}\Bigg[(2\pi)^{-k_\alpha/2}|\bar{T}P_0\bar{T}'+\bar{R}\bar{Q}\bar{R}'|^{-1} \\
&\times \exp\Big\{-\frac{1}{2}\big(\alpha_t-\bar{T}\alpha_{t-1}\big)'\big((\bar{T}P_0\bar{T}'+\bar{R}\bar{Q}\bar{R}')^{-1}\big)\big(\alpha_t-\bar{T}\alpha_{t-1}\big)\Big\} \\
&\times (2\pi)^{-k_y/2}|\bar{Z}P_0\bar{Z}'+\bar{H}|^{-1}\exp\Big\{-\frac{1}{2}\big(y_t-\bar{Z}\alpha_t\big)'\big((\bar{Z}P_0\bar{Z}'+\bar{H})^{-1}\big)\big(y_t-\bar{Z}\alpha_t\big)\Big\}\Bigg] \qquad \text{(A.47)} \\
&= (2\pi)^{-\frac{T(k_\alpha+k_y)}{2}}|\bar{T}P_0\bar{T}'+\bar{R}\bar{Q}\bar{R}'|^{-T}|\bar{Z}P_0\bar{Z}'+\bar{H}|^{-T} \\
&\times \exp\Bigg\{-\frac{1}{2}\Bigg(\sum_{t=1}^{T}\frac{\big(\alpha_t-\bar{T}\alpha_{t-1}\big)'\big(\alpha_t-\bar{T}\alpha_{t-1}\big)}{(\bar{T}P_0\bar{T}'+\bar{R}\bar{Q}\bar{R}')}+\sum_{t=1}^{T}\frac{\big(y_t-\bar{Z}\alpha_t\big)'\big(y_t-\bar{Z}\alpha_t\big)}{(\bar{Z}P_0\bar{Z}'+\bar{H})}\Bigg)\Bigg\}
\end{aligned}
$$

Although the function appears intractable at first, note that the conditionals for y_t and α_t are straightforward to derive. Simply rewrite the exponential. Thus this can be estimated by a Gibbs Sampler. In [Geweke and Tanizaki, 2001] the hyperparameters are defined as stochastic and prior distributions are defined. In such a case, it is very likely that deriving conditionals for these parameters above is intractable. The authors then propose to make use of the Metropolis-Hastings algorithm within each Gibbs sampling step.

A.5 Normal-Inverse-Gamma Distribution

The Gamma distribution is given either by a shape and scale (s, S) parametrization or a shape and rate parametrization (α, β):

$$\begin{aligned} G(x|s,S) &= \frac{x^{s-1}}{S^s\Gamma(s)}exp\Big(-\frac{x}{S}\Big) \\ G(x|\alpha,\beta) &= \frac{\beta^\alpha x^{\alpha-1}}{\Gamma(\alpha)}exp\Big(-\beta x\Big) \end{aligned} \tag{A.48}$$

They are transformed into one another by the relations: $\alpha = s$ and $S = \frac{1}{\beta}$.

The relation between the inverse-gamma and the gamma is quantified by:

$$X \sim G(\alpha,\beta) \qquad \text{then} \qquad \frac{1}{X} \sim IG(\alpha,\beta) \tag{A.49}$$

Thus:

$$IG(x|\alpha,\beta) = \frac{\beta^\alpha x^{-\alpha+1}}{\Gamma(\alpha)}exp\Big(-\frac{\beta}{x}\Big) \tag{A.50}$$

The normal-inverse gamma is then given by:

$$\begin{aligned} \sigma^2 &\sim IG(\alpha,\beta) \\ \mu|\sigma^2 &\sim N(\mu_0,\sigma^2) \\ P[x|\mu,\sigma] &\sim \frac{1}{\sigma\sqrt{2\pi}}\frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{1}{\sigma^2}\right)^{\alpha-1}exp\left(-\frac{2\beta+(x-\mu)^2}{2\sigma^2}\right) \end{aligned} \tag{A.51}$$

A.6 Multivariate versions of Normal-Inverse-Gamma distribution

Although the normal distribution is quite straightforwardly scaled into a n-dimensional version of itself, this is not the case for the gamma and the inverse gamma distribution. Interestingly, there exist several forms in the literature, but commonly the Wishart distribution is chosen for Bayesian models. The normal-inverse-Wishart prior is given by:

$$\begin{aligned} \Sigma \sim& IW(\Lambda_0^{-1},v_0) \\ \mu|\Sigma \sim& N(\mu_0,\Sigma/\kappa_0) \\ P[\mu,\Sigma] =& NIW(\mu_0,\kappa_0,\Lambda_0,v_0) \\ =& \frac{|\Lambda_0|^{v_0/2}}{2^{v_0 d/2}\Gamma_d(v_0/2)(2\pi/\kappa_0)^{d/2}}exp\left(-\frac{1}{2}tr\Big(\Lambda_0\Sigma^{-1}\Big)-\frac{\kappa_0}{2}(\mu-\mu_0)^T\Sigma^{-1}(\mu-\mu_0)\right) \end{aligned} \tag{A.52}$$

B

Fama and French Portfolios Dataset

Starting Month	Type	Starting Month	Type
192607	Agric	192607	Ships
192607	Food	196307	Guns
196307	Soda	196307	Gold
192607	Beer	192607	Mines
192607	Smoke	192607	Coal
192607	Toys	192607	Oil
192607	Fun	192607	Util
192607	Books	192607	Telcm
192607	Hshld	192707	PerSv
192607	Clths	192607	BusSv
196907	Hlth	192607	Hardw
192607	MedEq	196507	Softw
192607	Drugs	192607	Chips
192607	Chems	192607	LabEq
193107	Rubbr	193004	Paper
192607	Txtls	192607	Boxes
192607	BldMt	192607	Trans
192607	Cnstr	192607	Whlsl
192607	Steel	192607	Rtail
196307	FabPr	192607	Meals
192607	Mach	192607	Banks
192607	ElcEq	192607	Insur
192607	Autos	192607	RlEst
192607	Aero	192607	Fin
192607	Other		

Table B.1: Portfolios and starting dates

C

Results 2 Step Procedure

C.1 Default Setting

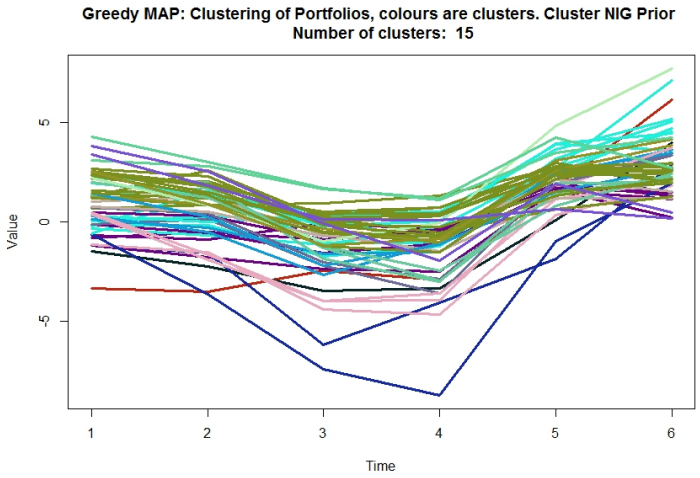

Fig. C.1

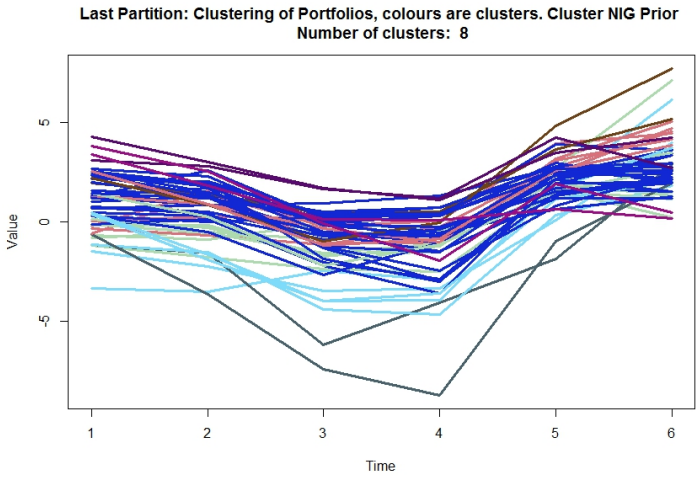

Fig. C.2

C.2 Varying α

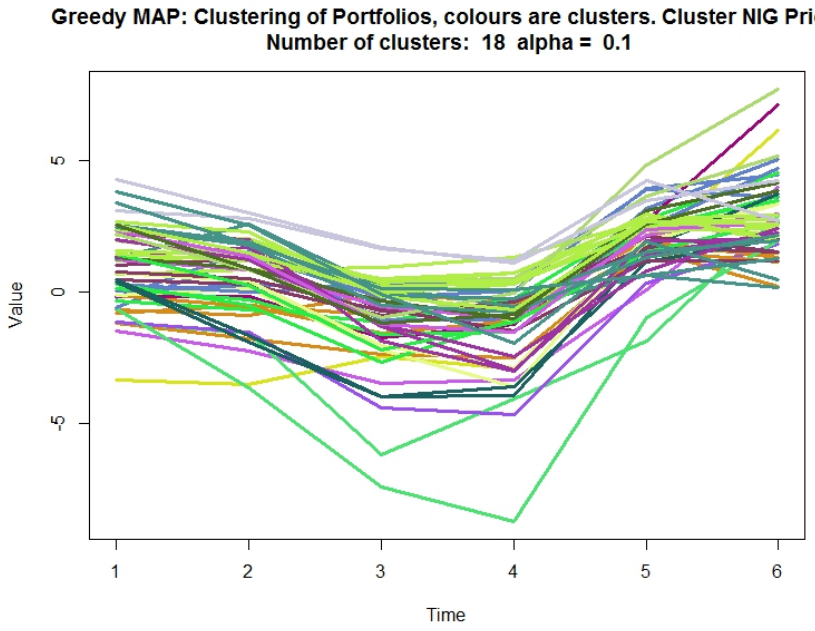

Fig. C.3

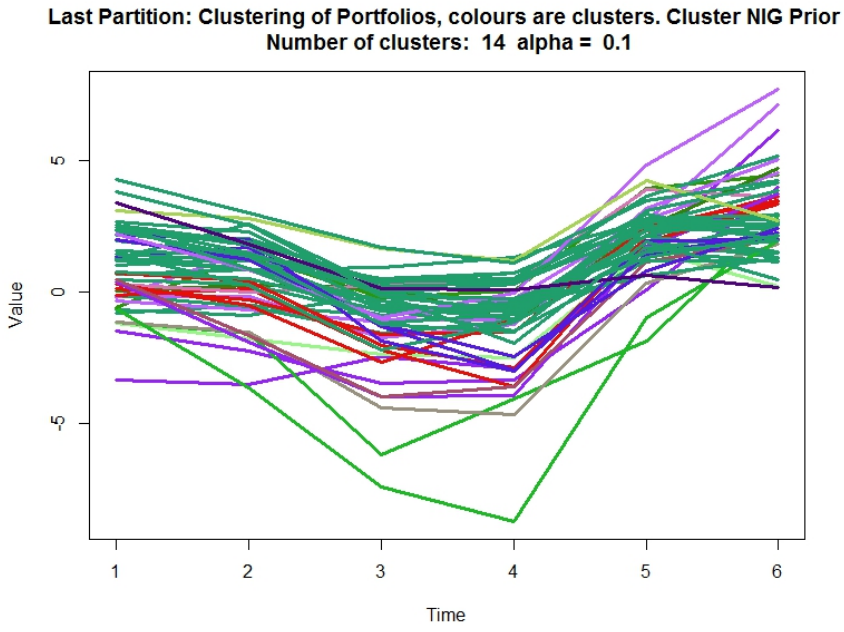

Fig. C.4

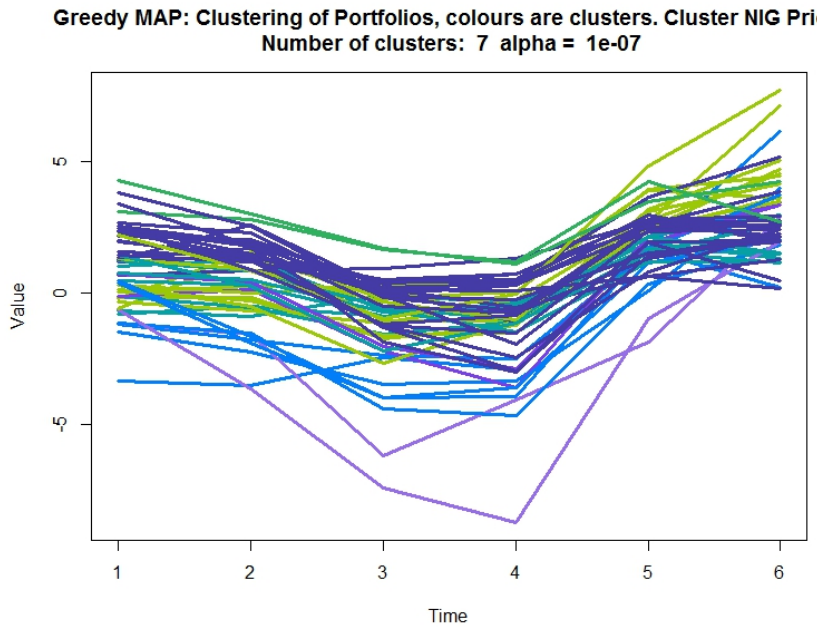
Greedy MAP: Clustering of Portfolios, colours are clusters. Cluster NIG Prior
Number of clusters: 7 alpha = 1e-07
Value
5
0
-5
1
2
3
4
5
6
Time

Fig. C.5

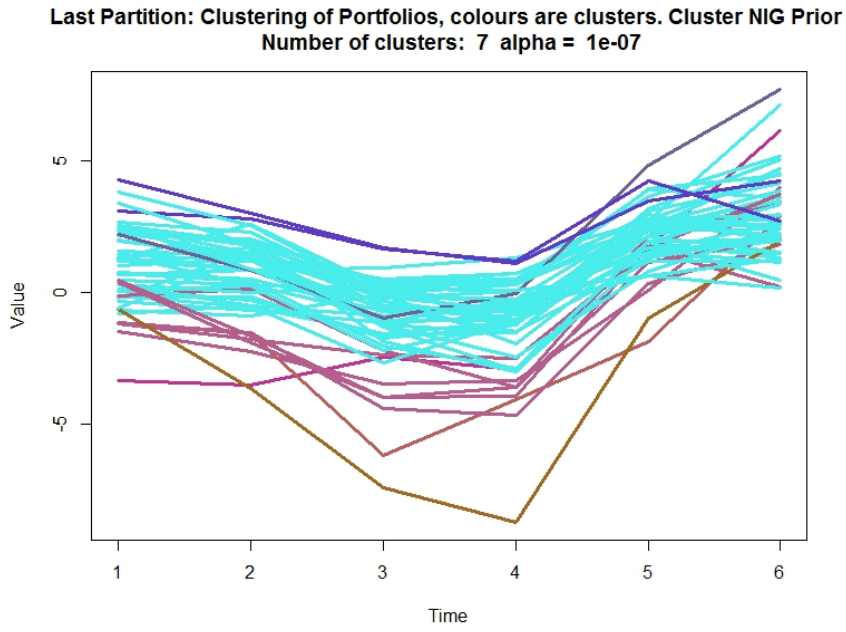
Last Partition: Clustering of Portfolios, colours are clusters. Cluster NIG Prior
Number of clusters: 7 alpha = 1e-07
Value
5
0
-5
1
2
3
4
5
6
Time

Fig. C.6

C.3 Varying s

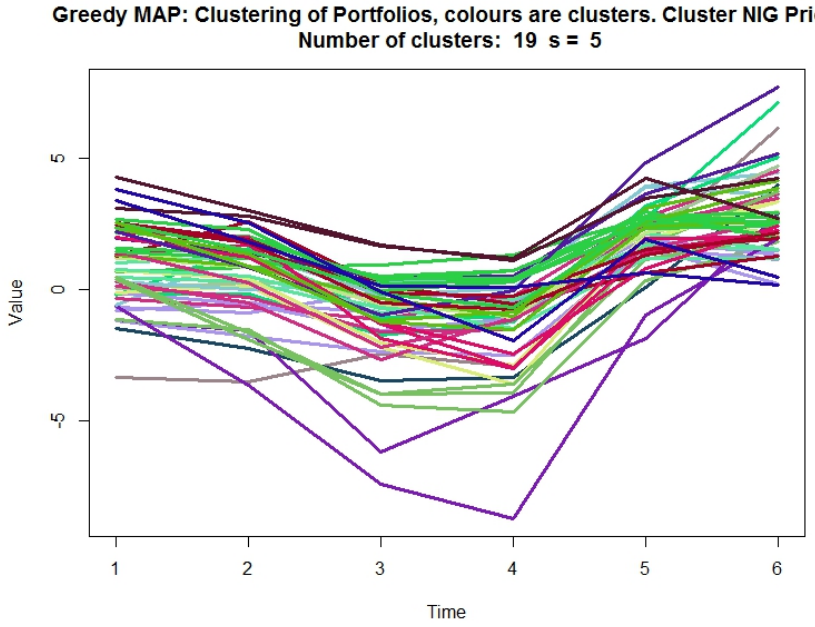

Fig. C.7

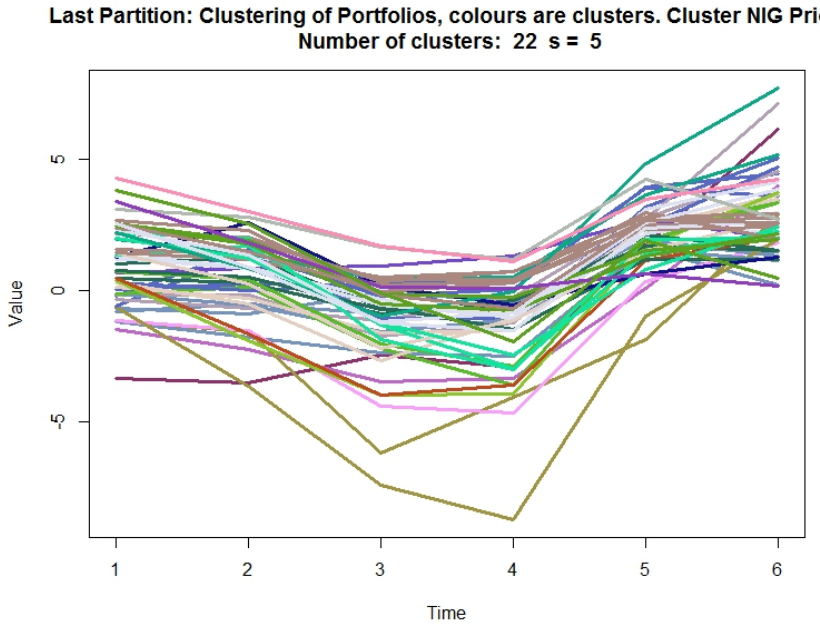

Fig. C.8

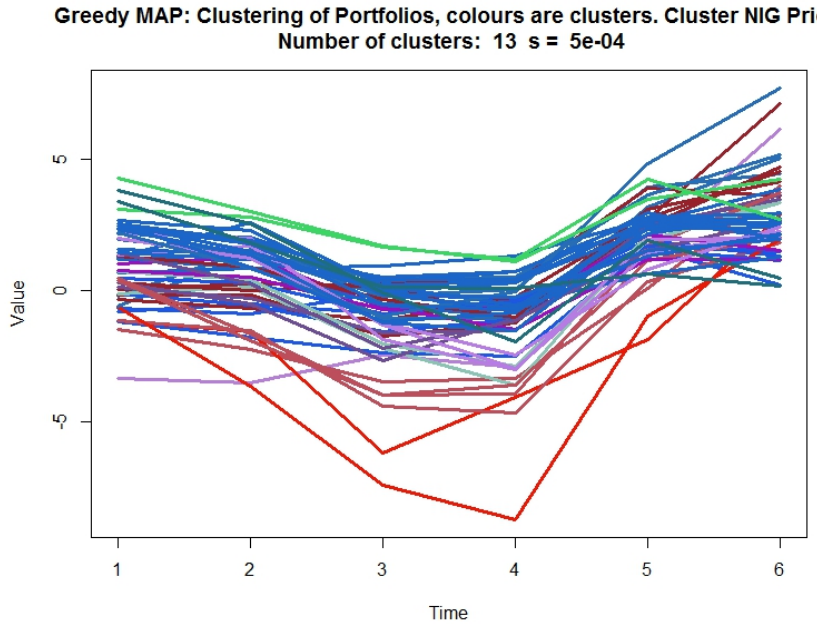

Fig. C.9

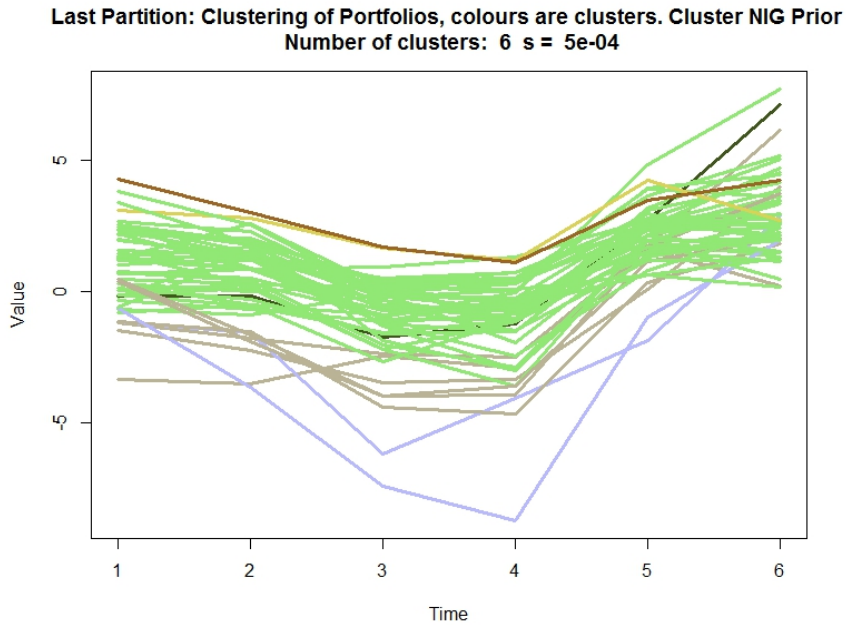

Fig. C.10

C.4 Varying S

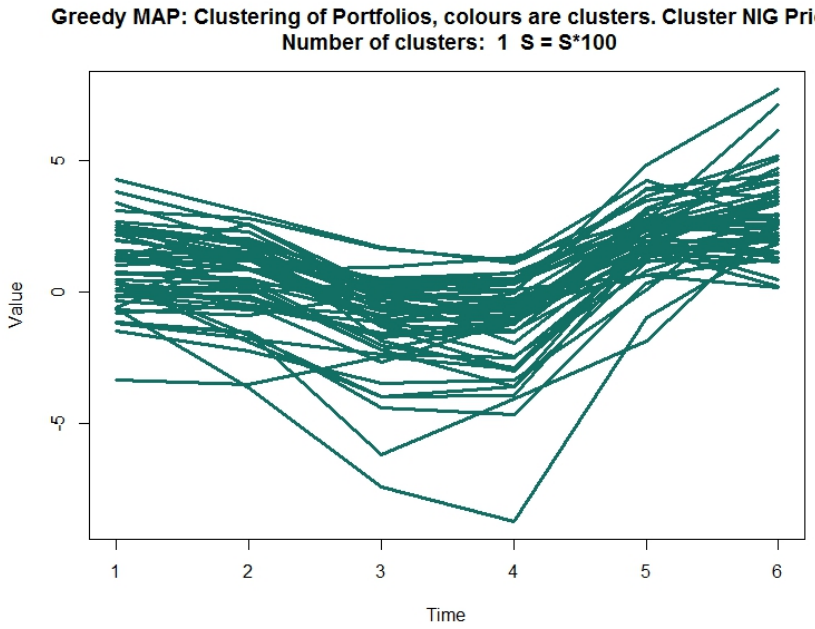

Fig. C.11

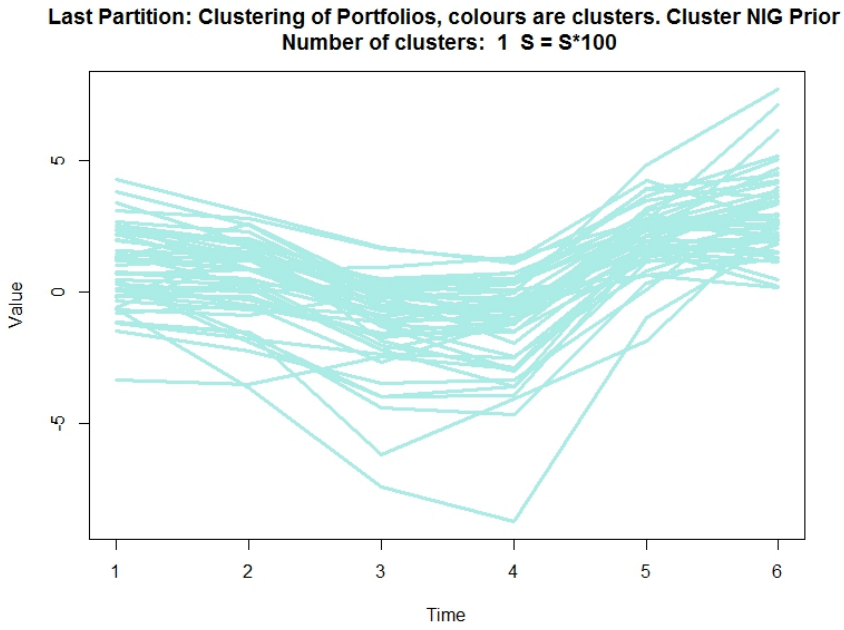

Fig. C.12

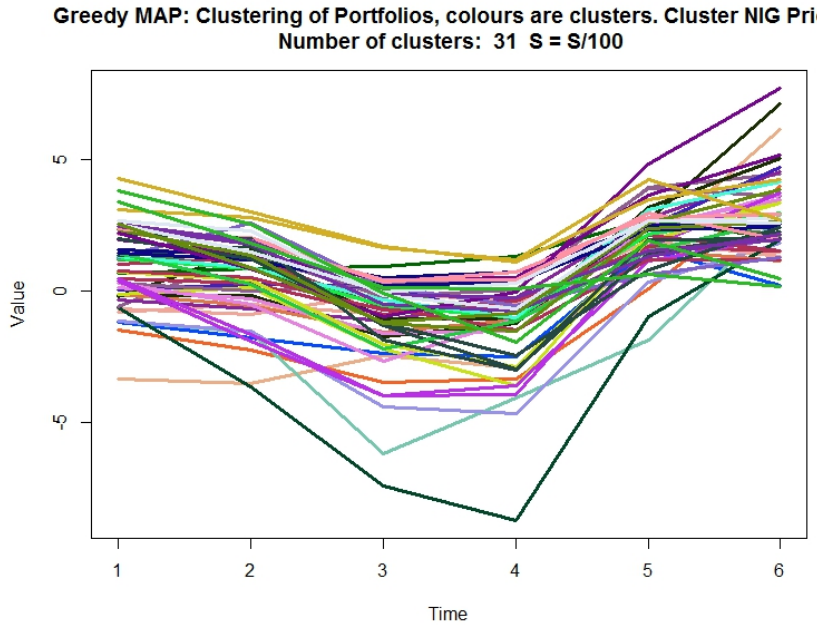

Fig. C.13

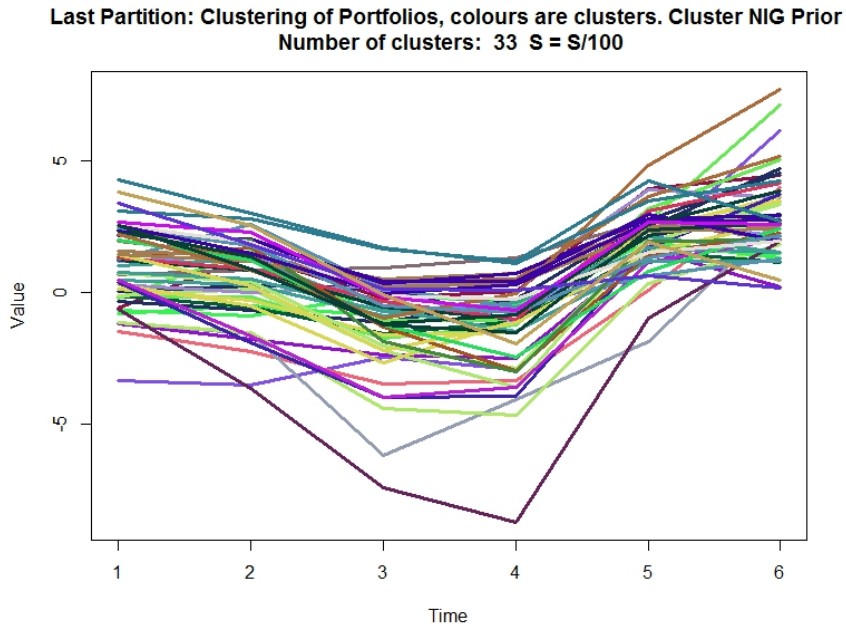

Fig. C.14

C.5 Varying τ

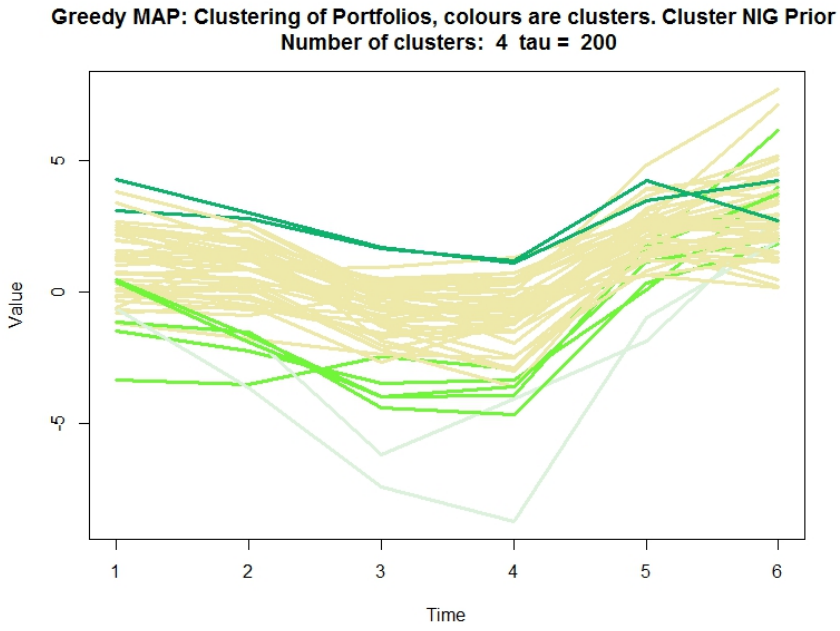

Fig. C.15

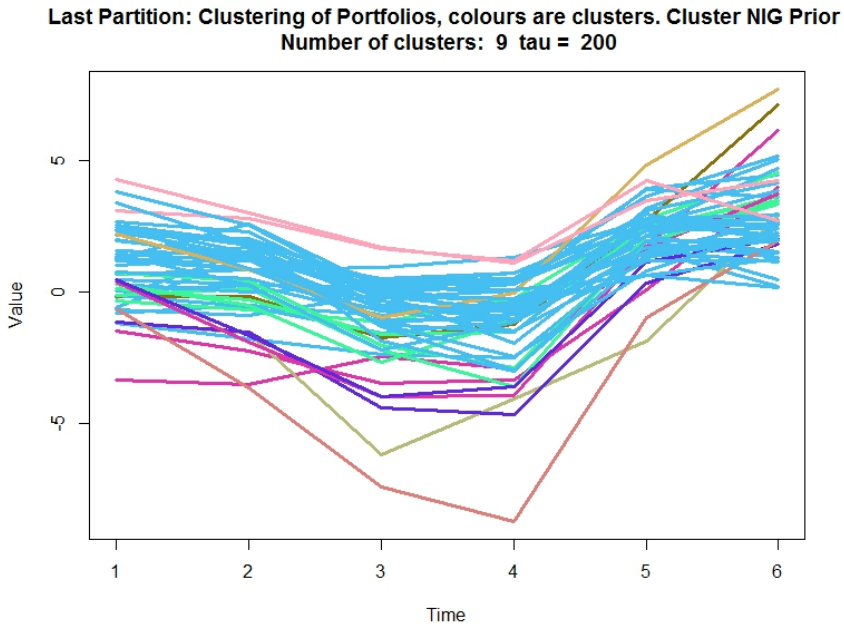

Fig. C.16

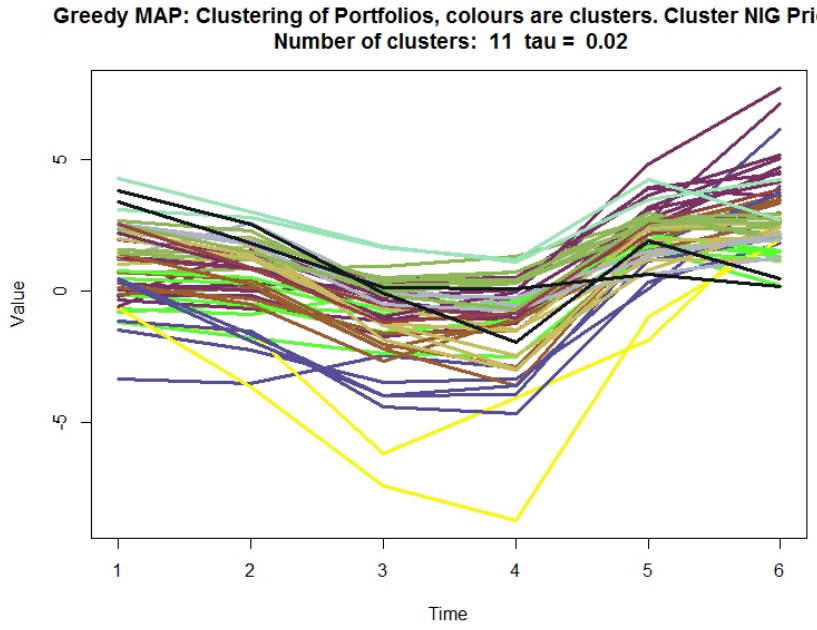

Fig. C.17

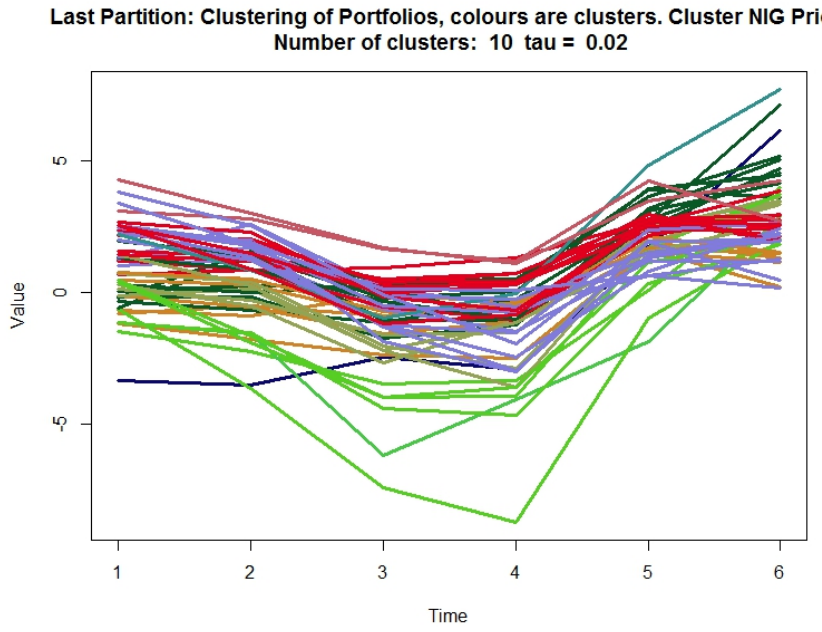

Fig. C.18

C.6 Without V being fixed

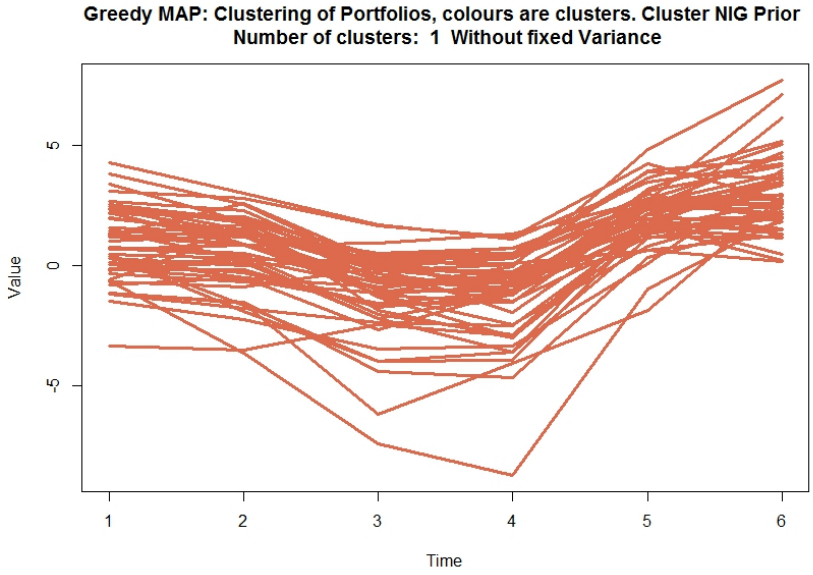

Fig. C.19

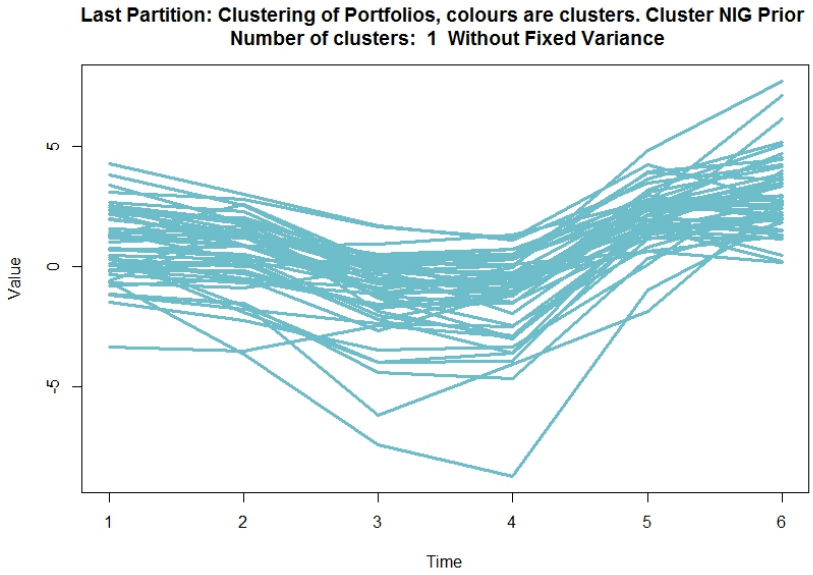

Fig. C.20

D

Results Clustered Correlations Model

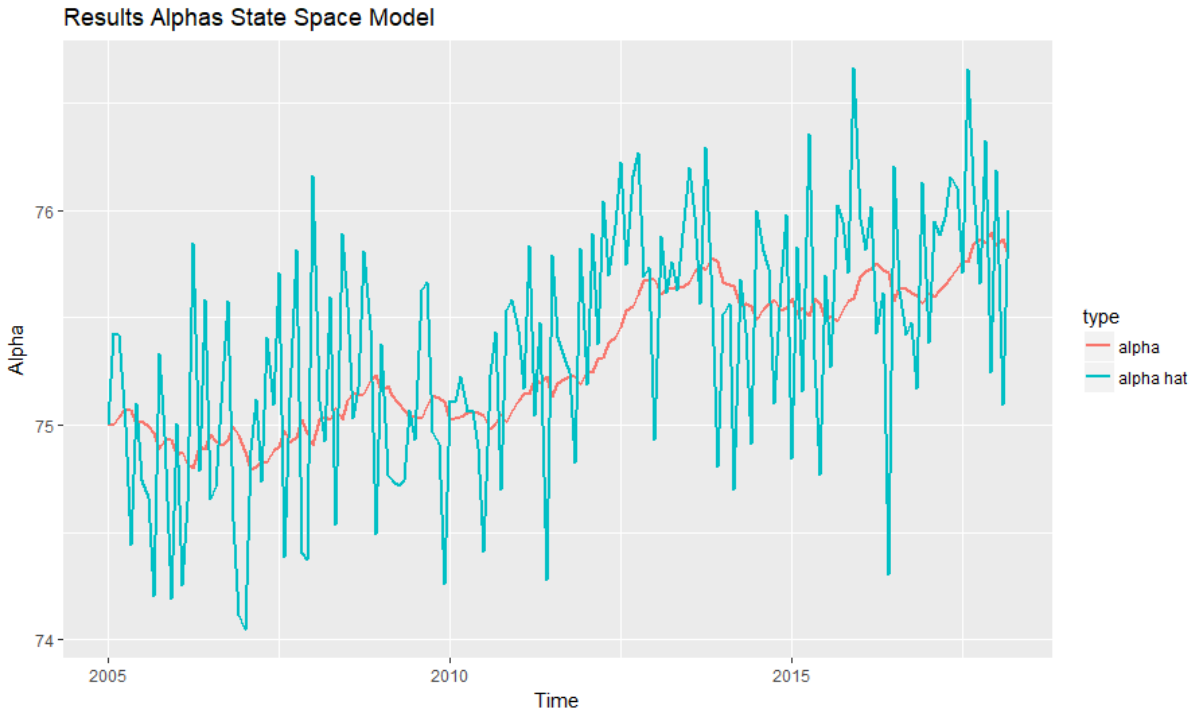

Fig. D.1

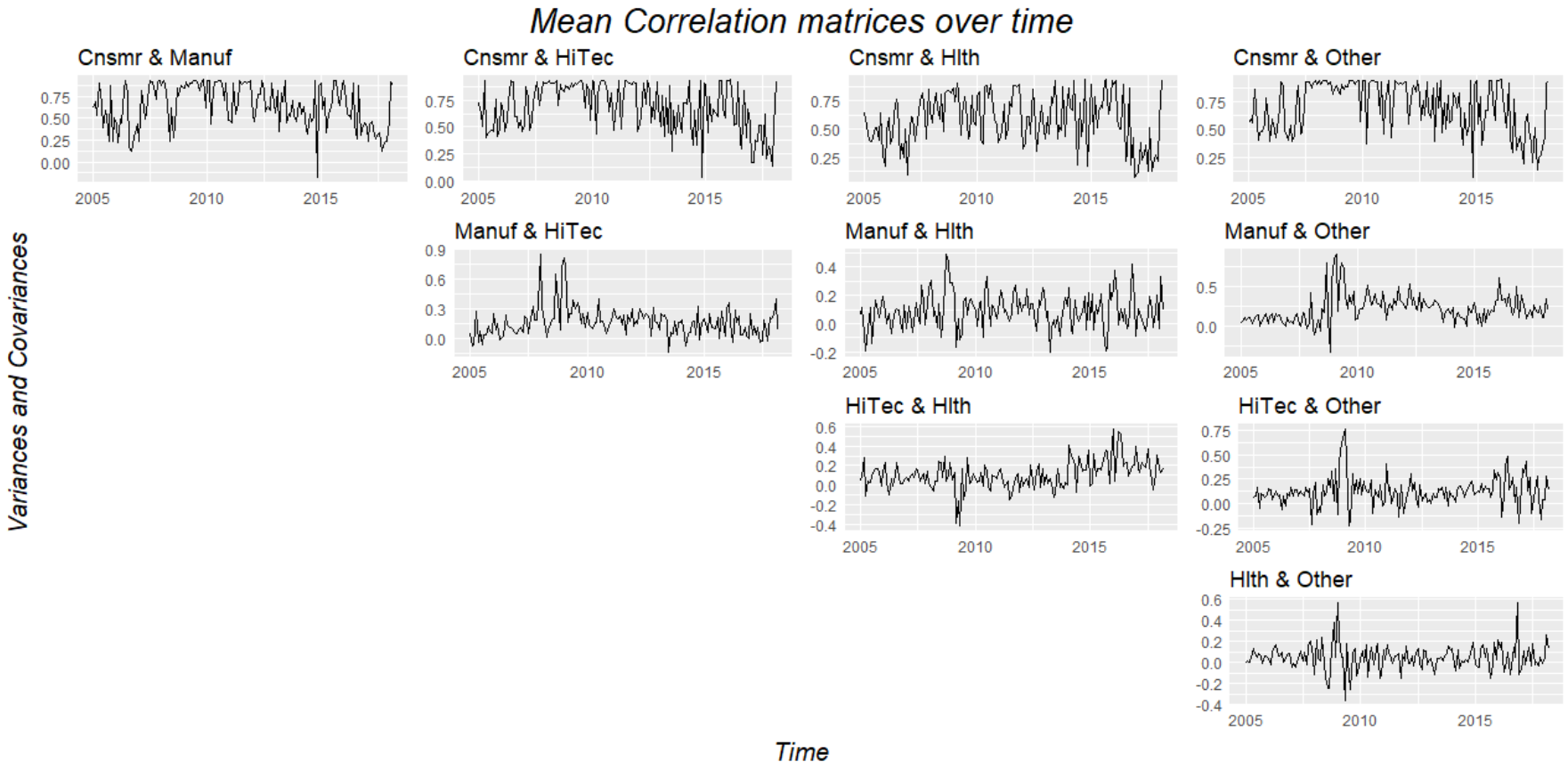

Fig. D.2: Clustered Correlations Model Results: Correlations Matrix

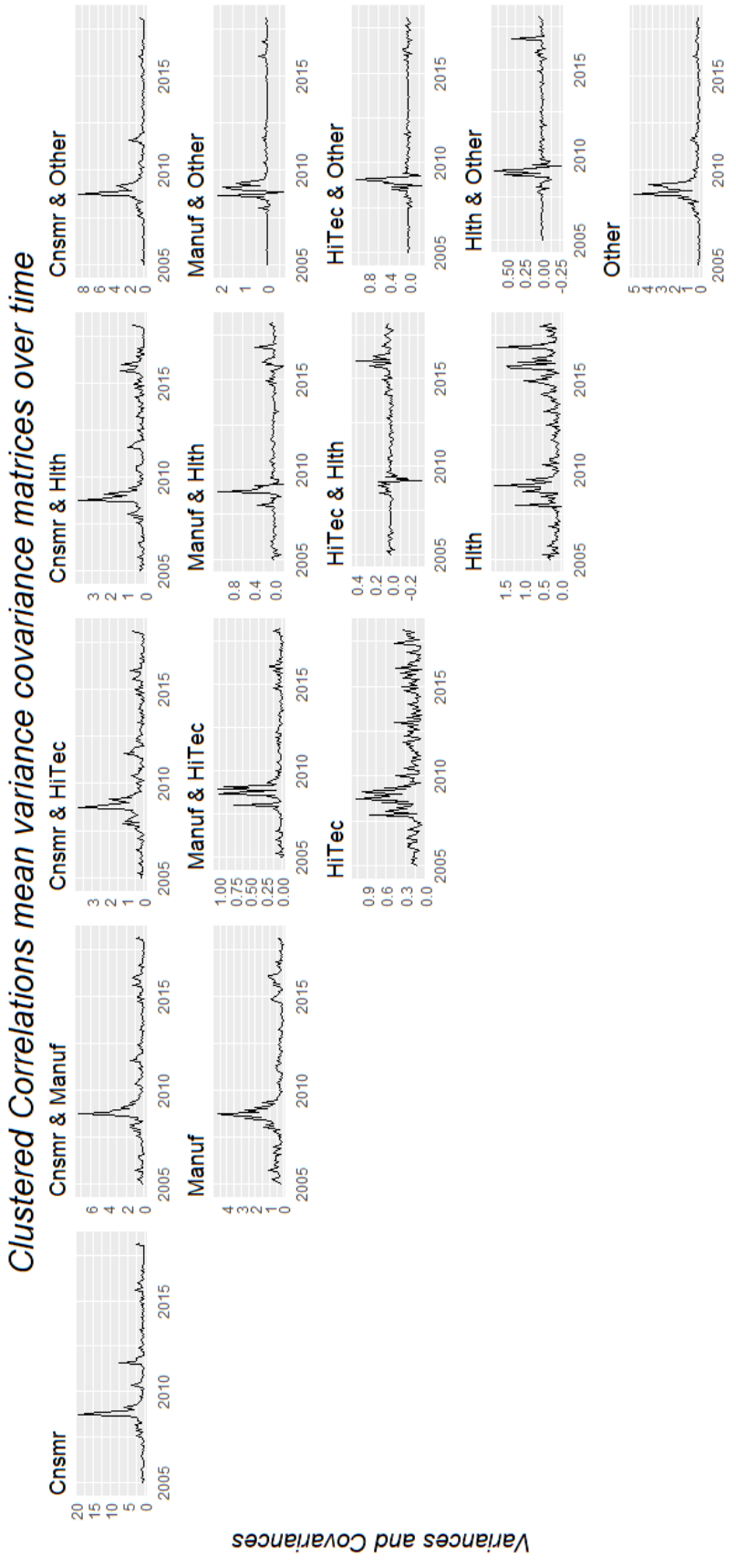

Fig. D.3: Clustered Correlations Model Results: Variance Covariance Matrix

E

Parallelizable parts of the Clustered Correlations Algorithm

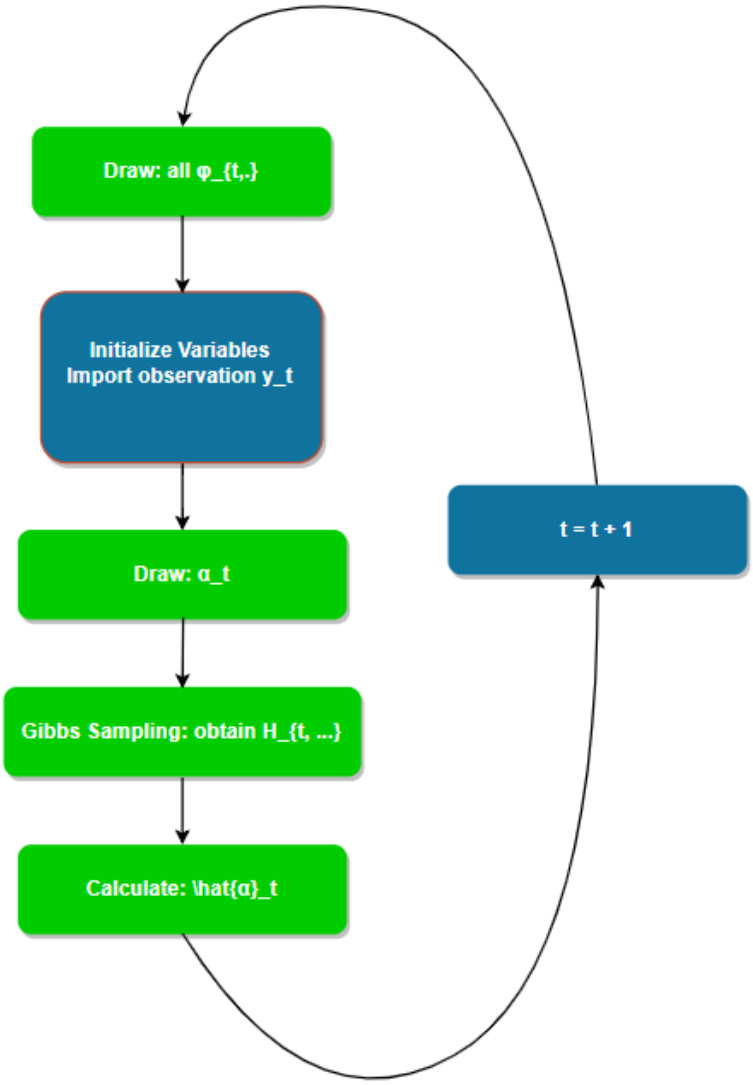

Fig. E.1: Parallelizable parts (green) of the Clustered Correlations Algorithm

Bibliography

[Ahmed and Xing, 2012] Ahmed, A. and Xing, E. P. (2012). Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream. *arXiv preprint arXiv:1203.3463*.

[Aldous, 1985] Aldous, D. J. (1985). Exchangeability and related topics. In *École d'Été de Probabilités de Saint-Flour XIII—1983*, pages 1–198. Springer.

[Andersen and Bollerslev, 1997] Andersen, T. G. and Bollerslev, T. (1997). Intraday periodicity and volatility persistence in financial markets. *Journal of empirical finance*, 4(2-3):115–158.

[Andersen et al., 2007] Andersen, T. G., Bollerslev, T., Christoffersen, P., and Diebold, F. X. (2007). Practical volatility and correlation modeling for financial market risk management. In *The risks of financial institutions*, pages 513–548. University of Chicago Press.

[Antoniak, 1974] Antoniak, C. E. (1974). Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The annals of statistics*, pages 1152–1174.

[Asai and McAleer, 2009] Asai, M. and McAleer, M. (2009). The structure of dynamic correlations in multivariate stochastic volatility models. *Journal of Econometrics*, 150(2):182–192.

[Asai et al., 2006] Asai, M., McAleer, M., and Yu, J. (2006). Multivariate stochastic volatility: a review. *Econometric Reviews*, 25(2-3):145–175.

[Bauer and Vorkink, 2006] Bauer, G. and Vorkink, K. (2006). Multivariate realized stock market volatility.

[Blackwell and MacQueen, 1973] Blackwell, D. and MacQueen, J. B. (1973). Ferguson distributions via pólya urn schemes. *The annals of statistics*, pages 353–355.

[Blei and Lafferty, 2006] Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM.

[Bollerslev, 1986] Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327.

[Bollerslev, 1990] Bollerslev, T. (1990). Modelling the coherence in short-run nominal exchange rates: a multivariate generalized arch model. *The review of economics and statistics*, pages 498–505.

[Bollerslev et al., 1994] Bollerslev, T., Engle, R. F., and Nelson, D. B. (1994). Arch models. *Handbook of econometrics*, 4:2959–3038.

[Broderick et al., 2013] Broderick, T., Kulis, B., and Jordan, M. (2013). Mad-bayes: Map-based asymptotic derivations from bayes. In *International Conference on Machine Learning*, pages 226–234.

[Bródka et al., 2012] Bródka, P., Kazienko, P., and Kołoszczyk, B. (2012). Predicting group evolution in the social network. In *International Conference on Social Informatics*, pages 54–67. Springer.

[Chiriac and Voev, 2011] Chiriac, R. and Voev, V. (2011). Modelling and forecasting multivariate realized volatility. *Journal of Applied Econometrics*, 26(6):922–947.

[Corsi, 2009] Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics*, 7(2):174–196.

[Engle et al., 2008] Engle, R. F., Ghysels, E., and Sohn, B. (2008). On the economic sources of stock market volatility.

[Escobar, 1994] Escobar, M. D. (1994). Estimating normal means with a dirichlet process prior. *Journal of the American Statistical Association*, 89(425):268–277.

[Escobar and West, 1995] Escobar, M. D. and West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, 90(430):577–588.

[Ferguson, 1973] Ferguson, T. S. (1973). A bayesian analysis of some nonparametric problems. *The annals of statistics*, pages 209–230.

[Gelfand and Smith, 1990] Gelfand, A. E. and Smith, A. F. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, 85(410):398–409.

[Geweke, 2005] Geweke, J. (2005). *Contemporary Bayesian econometrics and statistics*, volume 537. John Wiley & Sons.

[Geweke and Tanizaki, 2001] Geweke, J. and Tanizaki, H. (2001). Bayesian estimation of state-space models using the metropolis–hastings algorithm within gibbs sampling. *Computational Statistics & Data Analysis*, 37(2):151–170.

[Golosnoy et al., 2012] Golosnoy, V., Gribisch, B., and Liesenfeld, R. (2012). The conditional autoregressive wishart model for multivariate stock market volatility. *Journal of Econometrics*, 167(1):211–223.

[Hansen and Yu, 2001] Hansen, M. H. and Yu, B. (2001). Model selection and the principle of minimum description length. *Journal of the American Statistical Association*, 96(454):746–774.

[Hartig et al., 2011] Hartig, F., Calabrese, J. M., Reineking, B., Wiegand, T., and Huth, A. (2011). Statistical inference for stochastic simulation models–theory and application. *Ecology letters*, 14(8):816–827.

[Harvey et al., 1994] Harvey, A., Ruiz, E., and Shephard, N. (1994). Multivariate stochastic variance models. *The Review of Economic Studies*, 61(2):247–264.

[Hinton and Roweis, 2003] Hinton, G. E. and Roweis, S. T. (2003). Stochastic neighbor embedding. In *Advances in neural information processing systems*, pages 857–864.

[Jacquier et al., 1994] Jacquier, E., Polson, N. G., and Rossi, P. E. (1994). Bayesian analysis of stochastic volatility models. *Journal of Business & Economic Statistics*, 12(4):371–389.

[Jeffreys, 1946] Jeffreys, H. (1946). An invariant form for the prior probability in estimation problems. In *Proceedings of the Royal Society of London a: mathematical, physical and engineering sciences*, volume 186, pages 453–461. The Royal Society.

[Jin and Maheu, 2009] Jin, X. and Maheu, J. M. (2009). Modelling realized covariances. *No. tecipa-382*.

[Kass, 2011] Kass, R. E. (2011). Statistical inference: The big picture. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 26(1):1.

[Keogh and Lin, 2005] Keogh, E. and Lin, J. (2005). Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and information systems*, 8(2):154–177.

[Kohonen, 1998] Kohonen, T. (1998). The self-organizing map. *Neurocomputing*, 21(1-3):1–6.

[Kotz and Nadarajah, 2004] Kotz, S. and Nadarajah, S. (2004). *Multivariate t-distributions and their applications*. Cambridge University Press.

[Krantz, 2012] Krantz, S. G. (2012). *Handbook of complex variables*. Springer Science & Business Media.

[Ledoit and Wolf, 2003] Ledoit, O. and Wolf, M. (2003). Honey, i shrunk the sample covariance matrix.

[Lienou et al., 2010] Lienou, M., Maitre, H., and Datcu, M. (2010). Semantic annotation of satellite images using latent dirichlet allocation. *IEEE Geoscience and Remote Sensing Letters*, 7(1):28–32.

[Maaten and Hinton, 2008] Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

[Mahieu and Schotman, 1994] Mahieu, R. and Schotman, P. (1994). *Stochastic volatility and the distribution of exchange rate news*. Institute for Empirical Macroeconomics, Federal Reserve Bank of Minneapolis.

[Noureldin et al., 2012] Noureldin, D., Shephard, N., and Sheppard, K. (2012). Multivariate high-frequency-based volatility (heavy) models. *Journal of Applied Econometrics*, 27(6):907–933.

[Palm, 1996] Palm, F. C. (1996). 7 garch models of volatility. *Handbook of statistics*, 14:209–240.

[Philipov and Glickman, 2006] Philipov, A. and Glickman, M. E. (2006). Multivariate stochastic volatility via wishart processes. *Journal of Business & Economic Statistics*, 24(3):313–328.

[Quintana and West, 1987] Quintana, J. M. and West, M. (1987). An analysis of international exchange rates using multivariate dlm's. *The Statistician*, pages 275–281.

[Ramanathan and Guan, 2006] Ramanathan, K. and Guan, S. U. (2006). Recursive self organizing maps with hybrid clustering. In *Cybernetics and Intelligent Systems, 2006 IEEE Conference on*, pages 1–6. IEEE.

[Ramanathan and Wechsler, 2013] Ramanathan, V. and Wechsler, H. (2013). Phishing detection and impersonated entity discovery using conditional random field and latent dirichlet allocation. *Computers & Security*, 34:123–139.

[Rasmussen, 2000] Rasmussen, C. E. (2000). The infinite gaussian mixture model. In *Advances in neural information processing systems*, pages 554–560.

[Raykov et al., 2014] Raykov, Y. P., Boukouvalas, A., and Little, M. A. (2014). Simple approximate map inference for dirichlet processes. *arXiv preprint arXiv:1411.0939*.

[Sethuraman, 1994] Sethuraman, J. (1994). A constructive definition of dirichlet priors. *Statistica sinica*, pages 639–650.

[Shephard and Sheppard, 2010] Shephard, N. and Sheppard, K. (2010). Realising the future: forecasting with high-frequency-based volatility (heavy) models. *Journal of Applied Econometrics*, 25(2):197–231.

[Teh et al., 2005] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2005). Sharing clusters among related groups: Hierarchical dirichlet processes. In *Advances in neural information processing systems*, pages 1385–1392.

[Tibshirani et al., 2001] Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

[Uhlig, 1994] Uhlig, H. (1994). On singular wishart and singular multivariate beta distributions. *The Annals of Statistics*, pages 395–405.

[Uhlig, 1997] Uhlig, H. (1997). Bayesian vector autoregressions with stochastic volatility. *Econometrica: Journal of the Econometric Society*, pages 59–73.

[Van Laarhoven and Aarts, 1987] Van Laarhoven, P. J. and Aarts, E. H. (1987). Simulated annealing. In *Simulated annealing: Theory and applications*, pages 7–15. Springer.

[West, 1996] West, M. (1996). *Bayesian forecasting*. Wiley Online Library.

[Windle et al., 2014] Windle, J., Carvalho, C. M., et al. (2014). A tractable state-space model for symmetric positive-definite matrices. *Bayesian Analysis*, 9(4):759–792.

[Xing and Girolami, 2007] Xing, D. and Girolami, M. (2007). Employing latent dirichlet allocation for fraud detection in telecommunications. *Pattern Recognition Letters*, 28(13):1727–1734.

[Xu et al., 2014] Xu, K. S., Kliger, M., and Hero Iii, A. O. (2014). Adaptive evolutionary clustering. *Data Mining and Knowledge Discovery*, 28(2):304–336.

[Xu et al., 2008a] Xu, T., Zhang, Z., Philip, S. Y., and Long, B. (2008a). Dirichlet process based evolutionary clustering. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 648–657. IEEE.

[Xu et al., 2008b] Xu, T., Zhang, Z., Philip, S. Y., and Long, B. (2008b). Evolutionary clustering by hierarchical dirichlet process with hidden markov state. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 658–667. IEEE.

[Yang, 1993] Yang, M.-S. (1993). A survey of fuzzy clustering. *Mathematical and Computer modelling*, 18(11):1–16.

[Yu et al., 2017] Yu, P. L., Li, W., and Ng, F. (2017). The generalized conditional autoregressive wishart model for multivariate realized volatility. *Journal of Business & Economic Statistics*, 35(4):513–527.