USING MARKOV-SWITCHING VECTOR AUTOREGRESSIONS FOR MODELLING INTRAOPERATIVE HEMODYNAMICS

MSC THESIS REPORT

by

Rutger VAN BEEK

in partial fulfillment of the requirements for the degree of

Master of Science
in Applied Mathematics,

at the Delft University of Technology,

to be defended on the 5th July at 14:30

Student number:	4573145	
Supervisors:	Dr. N.Parolya	TU Delft, DIAM, Statistics
	Dr. F. van Lier	Erasmus MC, Anesthesiology
	Dr. S.E. Hoeks	Erasmus MC, Anesthesiology
Thesis committee:	Dr. M.B. van Gijzen	TU Delft, DIAM, Numerical Analysis

An electronic version of this thesis is available at http://repository.tudelft.nl/.

PREFACE

I vividly remember the moment I started with this thesis. The first paper I read had a title starting with 'troponin T'. And I remember just stopping with reading at that point and first finding out what troponin T was even about. For me, that is a large part of what *applied* mathematics is about. Understanding the context and modelling the situation mathematically. Only then turning to mathematical tools. The last step, the translation of mathematical results into practical, in this case medically relevant, results. I think this last step is very interesting and was very prominent in this research. A whole chapter in this thesis, chapter 6 is dedicated to this. I think this also makes applied mathematics interesting; how can we make the mathematical results meaningful?

This thesis marks the end of my studies at the TU Delft. 6 years ago, I started with the bachelor, to find there was a whole world of mathematics I had no idea about. I already turned my mathematical skills into practice during my year at Project March, which I definitely enjoyed and where I learned a lot besides mathematics. During my masters, I enjoyed going deep into my main interests, statistics, data, and machine learning.

This thesis incorporates 7 months of hard work, which I could not have done without help of course. First of all, I want to thank Sanne en Felix from the Erasmus MC. I want to thank you for providing the project including the interesting dataset, for explaining me the medical side of the project, and for the enthusiasm when I tried to explain the mathematics and the medical implications. And a special thanks for the day at the OR, where I experienced surgeries from close. This helped in interpreting the results, but also changed my look on surgery, which I noticed when I had surgery myself a month later. Next, I want to thank my TU Delft supervisor, Nestor, for helping me with the mathematical side of things. I remember asking about covariance shrinkage in weighted least squares and Nestor telling me about his own research on this specific subject.

Next to my supervisors, I want to thank all the people I studied with at the TU Delft during the past coming months. Some people on many days and some on less, but I want to thank you all: Roel, Olav, Joris, Björn, Maaike, Floortje, Sjoerd, Thomas, and Brian. I want to thank all these people for the discussions, lunch walks, and just being there. Especially Roel's disappointment every time I arrived later or left early to go sporting, probably positively influenced this thesis.

Rutger van Beek
Delft, 23 June 2022

ABSTRACT

Cardiac complications after surgery are common irrespective of the underlying condition. The postoperative level of troponin T is a good marker for cardiac complications. Little is known on the pathology of the release of troponin T in the blood, while a better understanding might provide the ability to reduce the complications. The goal of the thesis is to find patterns in intraoperative data that are related to the release of troponin T in the blood during surgery. The states resulting from estimating an MSVAR on intraoperative hemodynamic data were interpreted and related to postoperative troponin T measurements. The MSVAR was estimated in two ways: with the EM algorithm and in Bayesian fashion with the Gibbs sampler. Prior distributions were chosen and a Gibbs sampler was developed for estimating the MSVAR with these priors. The differences between the EM algorithm and the Gibbs sampler are mostly fundamental and not practical. Furthermore, the MSVAR is an appropriate model for modelling intraoperative hemodynamic data. The states of the MSVAR were related to various surgery variables, but did not have any prognostic value for predicting postoperative troponin T. The states related to the external shocks continuously given to the patient during surgery rather than the patient's state.

CONTENTS

List of Figures

List of Tables

List of Algorithms

1

INTRODUCTION

It is estimated that 4.2 million people die within 30 days after surgery each year [1]. Although surgeries have become much safer over the years, severe risks still exist. In Europe, the mortality rate for non-cardiac surgery is approximately 4% [2]. Surprisingly, many patients who undergo non-cardiac surgery end up with myocardial injury, which appears to be the main cause of postoperative mortality [3, 4, 5, 6]. How can it be that so many end up with cardiac problems after surgery?

1.1. CARDIAC PROBLEMS AFTER NON-CARDIAC SURGERY

Cardiac problems after non-cardiac surgery is a problem, having a large influence on the surgery outcome. Cardiovascular complications are generally not related to the patient's medical condition, but rather the body and especially the heart being under stress during surgery. In this section, a deeper look will be shed at cardiac complications from surgery.

1.1.1. STABILIZING PATIENTS DURING SURGERY

During surgery, general anaesthesia is commonly used. General anesthesia results in unconsciousness, paralysis, and a blunting of the stress response [7]. Due to anaesthesia, the body is not fully capable of controlling itself, especially when combined with the stress induced by surgery. Therefore, an anesthesiologist monitors the patient. For example, in the USA minimal monitoring includes electrocardiography (ECG), heart rate, blood pressure, inspired and expired gases, blood oxygen saturation (pulse oximetry), and temperature [8]. On the basis of these measurements, the anesthesiologist can supply medication. For example, by giving vasopressors, blood vessels narrow and blood pressure increases. However, the amount of myocardial injury after surgery tells that some patients are not stable, which is only observed postoperatively. Myocardial injury can be observed by measuring an elevated level of troponin T in the blood after the operation, which the next section will be about.

1.1.2. TROPONIN T

Troponin T (TnT) is part of the troponin complex, which is a combination of proteins important for contracting muscles [9]. Troponin T is present in the heart muscles. After myocardial injury, such as an heart attack, the troponin T is released into the blood-stream, where it can be measured with lab tests. TnT concentration in the blood begins to increase a few hours after the myocardial injury and remains high for a couple of days.

Troponin T is associated with major adverse cardiovascular events in the first year after surgery [10]. Additionally, Troponin T turns out to have a major prognostic value in predicting postoperative mortality [3, 5, 11, 12]. Troponin T is still a good predictor of mortality in the case of kidney injury, which has a major effect on troponin T values in the blood [13]. In [14] it is shown that troponin T is also a marker for non-cardiac complications and mortality not related to myocardial injury. One can conclude that troponin T has a good prognostic value for postoperative complications. However, little is known about the pathology of troponin T release during surgery [15].

1.1.3. PATHOLOGY OF INTRAOPERATIVE TROPONIN T RELEASE

In the previous section, troponin T was shown to have a major prognostic value in the prediction of postoperative complications, especially myocardial injury and mortality. It therefore seems to be a good indicator of the patients state after surgery as it is related to most major complications. Next to that, it is measured in almost all patients in the first three days after surgery [10]. However, little is known on the pathology of intra-operative troponin T release. The goal of this thesis is to find patterns in intraoperative data that are related to elevated postoperative troponin T measurements. These patterns can potentially be used to improve patient monitoring during surgery and establish the pathology of troponin T release.

1.2. DATABASE

To find patterns in data, one of course needs data. The database used in the thesis is provided by the Erasmus MC and includes data on Erasmus MC patients and surgeries performed at the Erasmus MC. The database is similar to the ones used in [10, 13, 15, 16]. The database includes patients who underwent intermediate or high-risk non-cardiac surgery at Erasmus MC between 1 July 2012 and 1 July 2017. Because on 1 July 2012, the postoperative troponin measurement protocol was started. Furthermore, only patients aged 60 or older are included as these are mainly at risk.

1.2.1. MEASURING TROPONIN T

As described before, since 1 July 2012 the troponin T is measured in each of the three postoperative days, unless the patient is discharged earlier. Troponin T is measured as high-sensitivity troponin T (hsTnT) using the Cobas e602 Troponin T hs STAT assay from Roch Diagnostics, Germany. This is the fifth generation assay, which has a higher prognostic value [17]. Preoperative measurements are generally only available for patients with known cardiovascular conditions. The peak value over three days of observations is considered the most relevant and is usually taken as the postoperative hsTnT value [10].

1.2.2. OTHER DATA

Next to hsTnT, other data in the database is obtained in the following way. Intraoperative data, such as heart rate and blood pressure, is recorded at irregular 1- to 5- minute intervals. The dataset also includes ST segments, which are derived from the ECG and recorded at 1- to 5- minute intervals for each of the leads. The patients had a preoperative screening that included a medical history check, physical examination, laboratory measurements, and electrocardiogram according to Erasmus MC policies. The preoperative screening, basic patient characteristics, intraoperative data, and preoperative use of medication can be retrieved from the electronic hospital patient information system. Survival status was checked with the civil registries.

1.3. RELATION TO OTHER RESEARCH

Related research mainly includes several papers using the same database. In [13] the prognostic value of hsTnT in the presence of kidney disease was investigated. Furthermore, in [15] the database was used to research the association of postoperative hypotension and myocardial injury, which was extended with [16]. Finally, major adverse cardiovascular events after surgery were found to be related to postoperative hsTnT [10]. These researches focused mainly on the prognostic value of postoperative hsTnT and not on intraoperative data.

1.3.1. INTERNSHIP REPORTS

Several Applied Mathematics students from TU Delft have done internships at the Erasmus MC all using the same database. All used intraoperative data in the database as well. Thomas van der Jagt researched the relation between ST segments and mean arterial pressure [18]. Extending the modelling of the intraoperative time series data Daniël Hoonhout introduced VAR modelling [19] and Rissalah Abdellah extended this research by introducing VECM [20]. Using a different model for each surgery, both VAR and VECM can be used to model intraoperative hemodynamics. However, model validation hypothesis testing shows that these models are not generally appropriate. Finally, Laura Veerhoek and Antoine Pomari worked on summarizing the intraoperative data for prediction of hsTnT [21, 22]. The intraoperative data was summarized using statistical functions. In combination with preoperative characteristics, several statistical learning models were applied. The best model had a 0.75 ROC AUC score in predicting high(> 50 ng/L) peak posteropative hsTnT. In conclusion, several approaches for modelling intraoperative hemodynamics are already researched, however no consensus is reached.

1.3.2. PREDICTION FROM INTRAOPERATIVE DATA

Finally, there is some research on postoperative prediction from intraoperative data. In [23] it is stated that in the prediction of postoperative complications generally only preoperative features are used. It is shown that random forests trained on both pre- and intraoperative features have improved accuracy for predicting postoperative complications. In particular, the accuracy of predicting in-hospital mortality increased from 0.77 to 0.88. The time series was summarized by using statistical functions.

Similarly, research was conducted by Xue et al. [24]. Preoperative and intraoperative

data was used to identify the risks of postoperative complications, such as pneumonia, acute kidney injury, deep vein trombosis, pulmonary embolism, and delirium. These turned out to be quite predictable with ROC AUC scores between 0.831 and 0.905. The intraoperative data was summarized using normalized values of statistical functionals applied to the time series, similarly as in [21]. Many different machine learning models were considered. Gradient boosting trees and deep neural networks performed best.

Finally, in a series of articles a different approach for summarising time series was shown [25, 26, 27]. The data is not from surgeries, but from ICU monitoring, which is very similar. Here, the time series data is modelled with a Markov-Switching Vector Autoregression. The relative time in each of the states is used as predictive features for prediction mortality. This approach is more elaborately discussed in 3.1.2. It was shown that time in states has additional prognostic value in addition to preoperative characteristics, reaching a ROC AUC score of 0.68. The score increased to 0.83 for a small subgroup of patients, which had a higher incidence of mortality and were identified by intraoperative hypotension and vasopressor usage.

Research shows that intraoperative data has an additional prognostic value in predicting postoperative complications, including mortality. Several approaches for dealing with intraoperative data are available. The approach of using statistical functions to summarise intraoperative time series was found to have a ROC AUC score of 0.75 to predict elevated hsTnT on the first postoperative day.

1.4. CONTENTS

After the introduction this thesis starts with mathematical theory on modelling time series. Next, the methods used to obtain troponin predictions from these time series models are described. From chapter four, the result chapters follow. Chapter four presents the results for fitting an MSVAR to hemodynamic data with the EM algorithm. Next, chapter five is a collection of results from fitting an MSVAR in Bayesian fashion with the Gibbs sampler. The last result chapter, chapter six, is about interpreting the states resulting from the hierarchical Bayes MSVAR model fitted on hemodynamic data. Next, the results are interpreted and combined in the discussion. The thesis is ended with an conclusion and several appendices.

2

TIME SERIES MODELLING

A lot of intraoperative data is so-called time series data. A time series is a sequence of data that have been observed at different points in time. The idea is that the observations are related based on the observation time. "The obvious correlation introduced by the sampling of adjacent points in time can severely restrict the applicability of the many conventional statistical methods traditionally dependent on the assumption that these adjacent observations are independent and identically distributed." [28]. Examples of time series are stock prices, weather temperatures and the number of people infected with Covid per day. Most of the intraoperative data in the database are time series, for instance heart rate, blood pressures and ST segments. "The primary objective of time series analysis is to develop mathematical models that provide plausible descriptions for sample data" [28]. In this thesis, several known methods from time series modelling will be used for modelling the time series data. The methods and their relations will be discussed in this chapter. However, first some basic properties of time series will be introduced.

2.1. INTRODUCTION TO TIME SERIES ANALYSIS

This chapter begins with several characteristics of time series and the notion of stationarity, which is an essential concept in time series analysis. The definitions in this section are taken from the book of Shumway and Stoffer, more elaborate information can be retrieved there [28].

Three basic properties of time series are: the mean function, the autocovariance function and the autocorrelation function. The mean function is the expectation at any given time point. For a real-valued stochastic process $\{X_t\}$ with density f_t at each time point t is given by:

$$\mu_{x_t} = \mathbb{E}(x_t) = \int_{\infty}^{\infty} x f_t(t) dx \tag{2.1}$$

The autocovariance function gives the covariance between the distribution of two time

points in a stochastic process:

$$\gamma_X(s,t) = \text{cov}(X_s, X_t) = \mathbb{E}\left[(x_s) - \mu_s)(x_t - \mu_t)\right] \tag{2.2}$$

Using the standard normalizing transformation from covariance to correlation we have that the autocorrelation function is given by:

$$\rho(s,t) = \frac{\gamma(s,t)}{\sqrt{\gamma(s,s)\gamma(t,t)}} \tag{2.3}$$

There is a very particular group of time series, which will be used as a building block for complex models, which is a white-noise sequence. The basic white-noise sequence is a sequence of uncorrelated random variables w_t, with mean 0 and a finite variance σ_w^2, notated as $w_t \sim \text{wn}(0, \sigma_w^2)$. There are other forms of white noise, being independent white noise, where the variables are not just uncorrelated but also independent. Next to that, there is also Gaussian white-noise, where the variables are normally distributed with mean zero and a common variance. For a Gaussian white-noise sequence the independence is implied, because the variables are uncorrelated and normally distributed.

2.1.1. STATIONARITY

There are two notions of stationarity, being weak and strict stationarity. A time series is strictly stationary if for every collection of values $\{x_{t_1}, x_{t_2}, \ldots, x_{t_k}\}$ we have that

$$\mathbb{P}\{x_{t_1} \leq c_1, \ldots, x_{t_k} \leq c_k\} = \mathbb{P}\{x_{t_1+h} \leq c_1, \ldots, x_{t_k+h} \leq c_k\} \tag{2.4}$$

for all time points $t_1, \ldots, t_k$, numbers $c_1, \ldots, c_k$ and shifts $h = 0, \pm 1, \pm 2, \ldots$. Any joined distribution does not change when shifted in time for a strictly stationairy time series. This implies that the distribution of the observations including its dependencies on the past does not change over time.

An easier, and more generally used form of stationarity is weak stationarity. A stochastic process is weakly stationary if the mean does not depend on time, and the autocovariance function only depends on the time difference. A time series, $\{X_t\}$, is stationary if

$$\mu(t) = \mu, \quad \forall t \tag{2.5}$$

$$\gamma(s, s+h) = \gamma(t, t+h), \quad \forall t, s \tag{2.6}$$

Generally, with stationairy a weakly stationairy time series is meant. For a stationary time series we can redefine the mean function to be equal to a constant μ. Furthermore, the autocovariance and autocorrelation functions can be redefined, because the functions do not depend on specific times, but rather only on the time difference or lag, h:

$$\gamma(h) = \text{cov}(x_t, x_{t+h}) \tag{2.7}$$

$$\rho(h) = \rho(s, s+h) = \frac{\gamma(s, s+h)}{\sqrt{\gamma(s,s)\gamma(s+h, s+h)}} = \frac{\gamma(h)}{\gamma(0)} \tag{2.8}$$

TESTING FOR STATIONARITY

For an observed time series, being a realisation of a stochastic process, one can never determine with certainty whether the time series is stationairy, however one can do hypothesis testing for stationarity. The test statistic for this is the Dickey-Fuller test statistic or an alteration [29]. Here, the basic Dickey-Fuller test statistic will be described as it is the basis for almost all hypothesis testing for stationairity. The test statistic $n(\phi-1)$ is based on the assumption that the time series, can be written as $x_t+\phi x_t+w_t$, where w_t is a Gaussian white noise sequence. The test is whether ϕ would be exactly equal to one, in which case we would have a non-stationairy time series. The test statistic is then given by: :

$$n(\phi-1)=\frac{\frac{1}{n\sigma_w^2}\sum_{t=1}^{n}w_t x_{t-1}}{\frac{1}{n^2\sigma_2^2}\sum_{t=1}^{n}x_{t-1}^2}\xrightarrow{d}\frac{\frac{1}{2}(\chi_1^2-1)}{\int_0^1 W^2(t)\,dt} \tag{2.9}$$

where ϕ is the parameter of the model, n is the number of observations, σ_w is the standard deviation of the Gaussian white noise sequence, χ_1^2 is the chi-squared distribution with one degree of freedom and W is a standard Brownian motion. Many similar statistics can be derived based on another model, for instance by including an intercept term. There is no closed form for the distribution of the test statistic and it should be obtained by simulation.

TRANSFORMATION FOR STATIONARITY

As will be shown in the upcoming section, stationarity is often required for several models. Therefore, one often checks whether the time series is stationary, before applying most models. One can use transformations to obtain a stationairy time series from a non-stationairy time series.

One general approach is to use differencing. The backshift operator, B, is defined by $Bx_t=x_{t-1}$ and extended to $B^k x_t=B^{k-1}(Bx_t)=x_{t-k}$. The differences of a time series can be defined in terms of the backshift operator,

$$\nabla x_t=x_t-x_{t-1}=(1-B)x_t \tag{2.10}$$

and similarly the differences of order d are: $\nabla^d=(1-B)^d$. Any polynomial trend can be removed by differencing with the amount equal to the degree of the polynomial. A time series that is stationairy up to a polynomial trend will then become stationairy.

Sometimes a time series is not stationairy because the fluctuations vary in size. And in many of these cases the Box-Cox family of transformations can be used [30]. These transformations are generally used to get a more normally distributed sample, but can also make the time series more stationairy in some cases. The family of transformations is given by:

$$y_t=\begin{cases}(x_t^\lambda-1)/\lambda & \lambda\neq 0\\ \log x_t & \lambda=0\end{cases} \tag{2.11}$$

Especially the log-transformation is used often, mainly in finance. The transformations is appropriate when the error variance seems to depend on the time series values.

2.1.2. ARMA

The most well-known time series models are the ARMA models. These consist of two fundamental building blocks, the autoregressive(AR) and moving average(MA) models. In this section the time series is assumed to have mean zero. One could replace the time series, X_t with $\tilde{X}_t = X_t - \mu$ to have a mean zero time series and model that time series. Like last section, the definitions are taken from the book from Shumway and Stoffer [28].

The idea of an autoregressive model is to model the current value as a linear combination of the past values and some error term [31]. In an AR(p) model, the p last values are used in the regression, which could be written as:

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_i \tag{2.12}$$

The error ε_i is assumed to be a white-noise sequence. One can find the parameters ϕ by applying least squares on the lags. The model can be written as a polynomial of the backshift operator applied to the time series.

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_i = \left(\sum_{i=1}^{p} \phi_i B^i X_t + \varepsilon_i = \sum_{i=1}^{p} \phi_i B^i \right) X_t + \varepsilon_i \tag{2.13}$$

Putting the error terms on one side and the time series on the other the following is obtained:

$$\phi(B) X_t = \left(1 - \sum_{i=1}^{p} \phi_i B^i \right) X_t = \varepsilon_t \tag{2.14}$$

$\Phi(B)$ is called the autoregressive operator.

The moving average model is similarly defined, again assuming a mean-zero time series. The model depends linearly on the past errors. MA(q) is short for a moving average model of order q, meaning the the last q error terms are used.

$$X_t = \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t = \sum_{i=1}^{q} \theta_i B^i \varepsilon_t + \varepsilon_t = \left(1 + \sum_{i=1}^{q} \theta_i B^i \right) \varepsilon_t = \theta(B) \varepsilon_t \tag{2.15}$$

Here, again the model is also written as a polynomial in terms of the backshift operator. $\Theta(B)$ is called the moving average operator. ε_t is a white-noise sequence as usual.

An autoregressive moving average model (ARMA), as the name suggests, is a combination of these two [31]. The autoregressive model can be viewed as applying a backshift polynomial to the time series values, whereas the moving average can be viewed as applying a similar polynomial to the error terms. The ARMA(p, q), meaning p autoregressive and q moving average terms, can than be viewed as equating these.

$$\phi(B) X_t = \left(1 - \sum_{i=1}^{p} \phi_i B^i \right) X_t = \left(1 + \sum_{i=1}^{q} \theta_i B^i \right) \varepsilon_t = \theta(B) \varepsilon_t \tag{2.16}$$

2.2. MULTI-DIMENSIONAL TIME SERIES

The idea of a time series can easily be extended to multiple dimensions. Instead of mapping a value to every time point, a time series can also map a vector of values to every

time point. One could apply a single dimensional time series model to each of the vector values, however such a model would not take into account any interaction effects. Think, for instance, about modelling a blood pressure and heart frequency together. These two variables clearly interact and should be modelled accordingly. In this section, the vector autoregressive (VAR) is explained. The definitions are taken from the book of Lütkepohl, which is an extensive book on multivariate time series analysis [32].

2.2.1. VAR

One easy extensions of the univariate time series models, is the vector autoregressive model. This is the vector extension of the autoregressive model. One still models the current value as a linear regression on past values. However, the past values of the other time series are also used as regressors, modelling the influence of other time series. If $\mathbf{y}_t$ is a vector of size K, it is modelled by a VAR(p) processes as:

$$\mathbf{y}_t = \nu + A_1\mathbf{y}_{t-1} + \ldots + A_p\mathbf{y}_{t-p} + \mathbf{e}_t \tag{2.17}$$

Here, ν is a vector of constants of size K, the A_i are $K \times K$-matrices of coefficients. $\mathbf{e}_t$ is the vector of error terms that should satisfy three conditions:

- $\mathbb{E}(\mathbf{e}_t) = \mathbf{0}$, similarly as in a white-noise sequence.
- $\mathbb{E}(\mathbf{e}_t\mathbf{e}_t^T) = \Sigma$, that is the covariance-matrix does not depend on time.
- $\mathbb{E}(\mathbf{e}_t\mathbf{e}_{t-k}^T) = 0, \quad k \neq 0$, meaning that there is no autocorrelation in the errors.

The defining equation can be concisely written in terms of matrices by defining:

$$Y := \begin{bmatrix} \mathbf{y}_0, & \mathbf{y}_1, & \ldots, & \mathbf{y}_T \end{bmatrix}, \tag{2.18}$$

$$B := \begin{bmatrix} \nu, & A_1 & \ldots & A_p \end{bmatrix}, \tag{2.19}$$

$$Z := \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \mathbf{y}_0 & \mathbf{y}_1 & \cdots & \mathbf{y}_{T-1} \\ \mathbf{y}_{-1} & \mathbf{y}_0 & \cdots & \mathbf{y}_{T-2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}_{-p} & \mathbf{y}_{-p+1} & \cdots & \mathbf{y}_{T-p} \end{bmatrix}, \tag{2.20}$$

$$U := \begin{bmatrix} \mathbf{e}_0 & \mathbf{e}_1 & \ldots & \mathbf{e}_T \end{bmatrix} \tag{2.21}$$

Using the matrix notation, the VAR(p) can be written as $Y = BZ + U$.

2.2.2. VAR ESTIMATION

Optimal estimation of VAR models, consists of two parts: estimation for a given lag and determining the optimal lag. Given a lag, one can view the problem as a linear regression. It turns out that the maximum likelihood estimator of the coefficients coincides with multiple least squares.

$$\hat{B} = YZ^T(ZZ^T)^{-1} \tag{2.22}$$

The maximum likelihood estimator for the covariance matrix is estimated as the MLE of the expectation, $\mathbb{E}(\mathbf{e}_t\mathbf{e}_t^T)$, where we use the residuals as estimators for the error terms,

which is $\hat{\Sigma}_{\text{MLE}} = \frac{1}{T}\sum_{t=1}^{T}\hat{\epsilon}_t\hat{\epsilon}_t^T$. This estimator is biased. An unbiased estimator can be obtained by compensating for the degrees of freedom of the model.

$$\hat{\Sigma} = \frac{1}{T - np - 1}\left(Y - \hat{B}Z\right)\left(Y - \hat{B}Z\right)^T \tag{2.23}$$

2.3. MODELS FOR MULTIPLE TIME SERIES

For each surgery there is a multidimensional time series available. Although each patient and situation are different, there might be many similarities. Therefore, models dealing with several independent time series are discussed here. The basic idea is the pooled VAR, which is extended with the panel VAR. Both these models and their benefits are discussed in [33].

2.3.1. POOLED VAR

The idea of a pooled VAR is to estimate a standard VAR as discussed in the previous section. However, we pool all the coefficients, meaning that the coefficients are required to be the same for every time series (or surgery). This means that it is assumed that each patient behaves the same. A normal VAR can be estimated with ordinary least squares. As these time series are independent, one can extract the regression equations for each time step and combine them. This results in a least-squares problem again. Using the concise matrix notation used from equation 2.18 , the surgeries are combined as follows:

$$\tilde{Y} = \begin{bmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{k-1} \\ Y_K \end{bmatrix}, \qquad \tilde{Z} = \begin{bmatrix} Z_0 \\ Z_1 \\ \vdots \\ Z_{k-1} \\ Z_K \end{bmatrix} \tag{2.24}$$

Because of independence we can estimate the pooled VAR(p) coefficients, $\hat{B}$ again with:

$$\hat{B} = YZ^T(ZZ^T)^{-1} \tag{2.25}$$

This results in a VAR model, fitted on many different surgeries.

2.3.2. PANEL VAR

In a pooled VAR each surgery is modelled with exactly the same model. This might be too far stretching; this can be relaxed a bit by introducing surgery-specific terms in the equations. In a fixed effects panel VAR a constant term per surgery is added, such that each surgery can be modelled with an individual mean. This model can be estimated by adding dummy variables to the MLS in the previous section. Let D be the matrix of these dummies. It has a column per surgery. In each row there is exactly one one in the

corresponding surgery column.

$$D = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \vdots & \vdots & 0 & 1 \end{bmatrix} \tag{2.26}$$

Then Z can be updated to $\tilde{Z} = \begin{bmatrix} Z & D \end{bmatrix}$ and the solution becomes again the following.

$$\hat{B} = Y\tilde{Z}^T(\tilde{Z}\tilde{Z}^T)^{-1} \tag{2.27}$$

2.4. MSVAR

Another idea is to not have a VAR per surgery, but a VAR per patient state. One can imagine that patients are in different states during a surgery, i.e. in pain, in stress, stable etc.. The assumption is that each patient can be modelled similarly as long as they are in the same state. However, this implies that the patient state should also be modelled. A general set of such time series models are the state-space models. A state-space model consists of two parts, a hidden state process $\{X_t, t = 0, 1, \dots\}$ and a model for each of the states [28]. A known model is a Markov-Switching Vector Autoregression (MSVAR) [34], which will be elaborated in the upcoming section. In an MSVAR the states are modelled as an Hidden Markov Model. The model in each state is equal to a Vector Autoregression as in section 2.2.1. First the Markov chain and the Hidden Markov Model are introduced. Next, their combination is discussed, the MSVAR, is discussed and finally two methods for estimation are explained.

2.4.1. Markov chains

In an MSVAR, the state is modelled as a Markov chain. A Markov chain is a stochastic model, describing a sequence of possible events with the Markov property [35]. The Markov property in a sequence of stochastic events means that the future events only depend on the current state and not on the past. Markov chains can be discrete or continuous in time and also in state space. In an MSVAR a discrete time Markov chain will be used, because the VAR also operates on discrete time events. Furthermore, a discrete state space will be used to have a finite number of VAR models. Given a finite amount, S, of possible states and a stochastic sequence of discrete-time events, $\{X_t\}_{n\in\mathbb{N}}, X_i \in \{1, \dots, S\}$. This sequence is a Markov chain if

$$\mathbb{P}(X_{n+1} = x_{n+1} | X_1 = x_1, \dots, X_n = x_n) = \mathbb{P}(X_{n+1} = x_{n+1} | X_n = x_n), \quad \forall x_1, \dots, x_n, x_{n+1} \in \{1, \dots, S\} \tag{2.28}$$

With a finite number of states this probability can be modelled with a transition matrix, P, with

$$P_{ij} = \mathbb{P}(X_{n+1} = j | X_n = i). \quad \forall i, j \in \{1, \ldots, S\} \tag{2.29}$$

In addition to a transition matrix, a Markov chain also has an initial distribution, often called π. Then $\mathbb{P}(X_0 = i) = \pi_i$. Generally, the uniform distribution over the states is chosen as the initial distribution. The transition matrix and initial distribution together, fully define a Markov chain.

2.4.2. HIDDEN MARKOV MODELS

Hidden Markov Models are a way of modelling a Markov chain in which the states are not directly observed. For instance, one can not directly observe whether a patient is stable, one can only observe things like heart rate and blood pressure, which are dependent on the patient state. One can use a Hidden Markov Model to model a Markov chain, without directly the observing it, but rather inferring it from observed variables with a dependence on the Markov chain. A Hidden Markov Model consists of a Markov chain with a transition matrix, P, and an initial distribution, π, a sequence of observations $y_0, y_1, \ldots$, an unobserved sequence of states $X_0, X_1, \ldots$, and finally an emission model, which is a probabilistic model for observing y_t in state i, notated as $b_i(y_t)$.

The standard method for estimating a Hidden Markov model is the Baum-Welch algorithm [36]. This is an Expectation-Maximization (EM) algorithm specifically for hidden Markov models. An EM algorithm is an iterative method for finding a (local) maximum likelihood estimate for parameters, where the model depends on unobserved latent variables [37]. There are many special cases of EM algorithms. Generally, the optimal parameters can be found given the latent variables and the most likely latent variables can be found given the parameters. After random initialisation of the parameters each iteration consists of two steps. First, the E-step, in which the latent variables given the parameters are estimated. Next, the M-step, in which the maximum likelihood estimators of the parameters given the latent variables are calculated. Whenever only MLE's are used the likelihood is proved to be monotonically increasing, as there always is a maximum likelihood for a probabilistic model, the algorithm converges [38]. However, it does not necessarily converge to the maximum; it can also converge to a local maximum.

BAUM-WELCH ALGORITHM

In the Baum-Welch algorithm, the E-step is to estimate the distribution over the states for each time step given the observations and the parameters of the Markov chain. The Baum-Welch algorithm uses the forward-backward algorithm for filtering and smoothing the likelihoods of the observations to get the distributions over the states. The forward-backward algorithm consists of two steps. First, the joint distribution over the states and the past and current observations is calculated. This is the forward step. The forward probability of being in state i at time t, $\alpha_i(t) = \mathbb{P}(Y_1 = y_1, \ldots, Y_t = y_t, X_t = i|\theta) =$ is calculated recursively:

1. Base case:

$$\alpha_i(0) = \pi_i b_i(y_0) \tag{2.30}$$

2. Recursive update:

$$\alpha_i(t+1) = b_i(y_{t+1}) \sum_{j=1}^{N} \alpha_j(t) P_{ji}, \quad t = 0, \ldots, T-1 \tag{2.31}$$

In the backward step, a similar calculation is done. The backward probability, $\beta_i(t) = \mathbb{P}(Y_{t+1} = y_{t+1}, \ldots, Y_T = y_T | X_t = i, \theta)$ is the probability of observing the future observations given the parameters and the current state being equal to i. These can be calculated with a backward recursion.

1. Base case:

$$\beta_i(T) = 1 \tag{2.32}$$

2. Backwards recursive update:

$$\beta_i(t) = \sum_{j=1}^{N} \beta_j(t+1) P_{ij} b_j(y_{t+1}), \quad t = T-1, \ldots, 1 \tag{2.33}$$

The results of the filtering and smoothing is a distribution over the states at every time step given all observations and parameters.

$$\gamma_i(t) = \mathbb{P}(X_t = i | Y, \theta) = \frac{\mathbb{P}(X_t = 1, Y|\theta)}{\mathbb{P}(Y|\theta)} = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^{N} \alpha_j(t)\beta_j(t)} \tag{2.34}$$

In the M-step, the parameters of the Markov chain, the initial probabilities, transition matrix and the emission model, are updated using the MLE. The empirical probability of the transition from state i to state j between time steps t and $t+1$ is equal to:

$$\xi_{ij}(t) = \mathbb{P}(X_t = 1, X_{t+1} = j | Y, \theta) = \frac{\mathbb{P}(X_t = i, X_{t+1} = j, Y|\theta)}{\mathbb{P}(Y|\theta)} = \frac{\alpha_i(t) a_{ij} \beta_j(t) b_j(y_{t+1})}{\sum_{k=1}^{N} \sum_{w=1}^{N} \alpha_k(t) a_{kw} \beta_w(t+1) b_w(y_{t+1})} \tag{2.35}$$

Given the empirical forward, backward, and transition probabilities, the necessary estimators can be formed.

- The initial probability can be estimated as the distribution over the states at the first time step: $\pi^* = \gamma(0)$.
- The estimator for the transition probabilities is observed number of transitions normalized with the number the transition was possible: $P^*_{ij} = \frac{\sum_{t=1}^{T-1} \xi_{ij}(t)}{\sum_{t=1}^{T-1} \gamma_i(t)}$.
- The emission model is estimated as the expected number of times the output was seen, normalized by the expected amount of time in the state: $b^*_i(v_k) = \frac{\sum_{t=1}^{T} \mathbb{1}_{y_t = v_k}}{\sum_{t=1}^{T} \gamma_i(t)}$

Given these specific E and M steps the Baum-Welch algorithm is just an EM-algorithm.

2.4.3. MSVAR

A Markov-Switching Vector Autoregression essentially is a Hidden Markov Model, where the emission model consists of a Vector Autoregression per state. The model was introduced in 1989 by Hamilton [34]. He modelled macro-economic variables and the states tended to coincide with different macro-economic regimes. A MSVAR thus consists of a Markov chain with an initial distribution, π and a transition matrix P. In addition, the model consists of a state space, S and a vector autoregressive model per state. All parameters including the transition matrix and initial distribution of the Markov chain are gathered in a vector θ

2.4.4. ESTIMATION WITH EM ALGORITHM

One can estimate an MSVAR with an EM-algorithm very similar to the Baum-Welch algorithm. Such an algorithm is presented in [39]. First, the weights of the Vector autoregression are randomly initialized. Next, the EM iteration is started. The probabilities are estimated similarly as in the Baum-Welch algorithm. First, the Hamilton filter is applied, where we estimate the distribution over the states of the observations y_t being generated by a particular state, which is denoted by ξ_t. If only the current observation is used, $\xi_{t|t}$ is obtained. It can be obtained by normalising the conditional likelihood. The conditional likelihood is the likelihood of an observation given a specific vector autoregression. Gaussian white-noise is assumed to have a probabilistic model.

$$\xi_{t|t}(i) = \mathbb{P}(X_t = i|Y_t = y_t,\theta) = \frac{\mathbb{P}(Y_t = y_t|X_t = i,\theta)}{\sum_{j=1}^{|S|}\mathbb{P}(Y_t = y_t|X_t = j,\theta)} = \frac{\eta_t(i}{\sum_{j=1}^{|S|} eta_t(j} \tag{2.36}$$

$\eta_t(i) = \mathbb{P}(Y_t = y_t|X_t = i,\theta)$ can be found by using the corresponding VAR model. However, the states are not independent, but modelled by a Markov chain. This means that past and future observations should also be taken into account. This can be done by applying the Hamilton filter [39]. For $t = 1,\dots T$ the following iteration should be done:

- Updating the current value with the information from the observation: $\hat{\xi}_{t|t}(i) = \frac{\hat{\xi}_{t|t}(i)\cdot\eta_t(i)}{\sum_{j=1}^{S}\hat{\xi}_{t|t}(j)\cdot\eta_t(j)}$ for $i = 1,\dots,S$
- Predicting the next distribution: $\hat{\xi}_{t+1|t} = P\hat{\xi}_{t|t}(i)$

This gives the inference for the state distribution given the current and past observations. However, future observations should also be taken into account. This is known as smoothing. For this specific problem the Kim smoother can be used[40]. Again there is an equation per time step, but this time the iteration is backward, thus for $t = T-1,\dots,1$ the calculation is:

$$\hat{\xi}_{t|T}(i) = \hat{\xi}_{t|t}(i)\cdot\left(P_i\cdot\frac{\hat{\xi}_{t+1|T}(i)}{\hat{\xi}_{t+1|t}(i)}\right), \quad i = 1,\dots,S \tag{2.37}$$

This is thus similar as in the forward-backward algorithm, where a forward pass and backward pass are combined.

In the M-step, θ is updated using MLE estimates. First the parameters of the VAR models per state are updated. However, the data points should be weighted with the

probability that of being in that state at that time. The maximum likelihood estimator for the weighted vector autoregression is the weighted multiple least squares solution of the same matrix. For each state there is a weight matrix, W_i, with the probabilities of being in state i on the diagonal.

$$W_i = \begin{bmatrix} \hat{\xi}_{0|T}(i) & 0 & \dots & \dots & 0 \\ 0 & \hat{\xi}_{1|T}(i) & 0 & \dots & 0 \\ \vdots & 0 & \ddots & 0 & 0 \\ 0 & \dots & 0 & \hat{\xi}_{T-1|T}(i) & 0 \\ 0 & \dots & \dots & 0 & \hat{\xi}_{T|T}(i) \end{bmatrix} \tag{2.38}$$

So we can define $\tilde{Y} = WY$ and $\tilde{Z} = WZ$, to get again the solution $\hat{B} = \tilde{Y}\tilde{Z}^T(\tilde{Z}\tilde{Z}^T)^{-1} = YWZ^T\left(ZWZ^T\right)^{-1}$. Furthermore, covariance shrinkage should be added, this generally gives a better estimates. Next to that, it also convenient that the estimated covariance matrices are certainly invertible.

COVARIANCE SHRINKAGE FOR WEIGHED MLS

Given a regression, with residuals $Y - XB$, with n observations and p dimensions, we have the (unweighted) sample covariance: $\tilde{\sigma} = \frac{1}{n}(Y - XB)(Y - XB)^T = \Sigma_{\text{MLE}}$. Given a weighted matrix W, which is diagonal, we get a weighted MLE: $\Sigma_{\text{WMLE}} = \frac{1}{n}(Y-XB)W(Y-XB)^T = S_n$. We want to shrink the covariance matrix with by adding to the diagonal, $\hat{\Sigma} = \hat{\alpha}^*\Sigma_{\text{WMLE}} + \hat{\beta}^* I$. The optimal estimators are as follows as obtained from [41], which is a generalization of [42].

$$\hat{\beta}^* = \frac{\text{Tr}(S_n)}{p\,\text{Tr}(W)}\left(1 - \hat{\alpha}^*\right)$$

$$\hat{\alpha}^* = 1 - \frac{p\,\text{Tr}\left(W^2\right)\left(\frac{\text{Tr}(S_n)}{\text{Tr}(W)}\right)^2}{p\left(\frac{\|S_n\|_F^2}{\text{Tr}(W)^2\,n} - \frac{n^2\|W\|_F^2 \cdot \text{Tr}(S_n)^2}{\text{Tr}(W)^2} + \|W\|_F^2\left(\frac{\text{Tr}(S_n)}{\text{Tr}(W)}\right)^2\right) - \left(\frac{\text{Tr}(S_n)}{\text{Tr}(W)}\right)^2\|W\|_F^2}$$

2.5. BAYESIAN ESTIMATION OF MSVAR

In the previous section, the Markov-Switching Vector Autoregression (MSVAR) was introduced. In addition, an algorithm for finding the maximum likelihood solution was described. The EM algorithm is a classical approach to finding an optimal MSVAR. However, there is another approach to statistics called Bayesian statistics. First, Bayesian statistics in general will be discussed. The section is continued with the Bayesian estimation of the MSVAR.

2.5.1. BAYESIAN ESTIMATION

The Bayesian interpretation of statistics is different from the classical interpretation. In Bayesian statistics, a probability expresses a degree of belief in an event. This probability is updated when new information is available. Given two events A and B, one has a prior belief that the event A happened, $\mathbb{P}(A)$, and given that A happened, one knows the influence on the event B, $\mathbb{P}(B|A)$. One can update the prior, $\mathbb{P}(A)$ with the new information

on the event B by using Bayes' theorem:

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(B|A)\mathbb{P}(A)}{\mathbb{P}(B)} \propto \mathbb{P}(B|A)\mathbb{P}(A) \tag{2.39}$$

The updated probability on the event A, $\mathbb{P}(A|B)$, is called the posterior. This can be especially applied to the estimation of model parameters. Given a random variable Y with observations y and a model for Y, $f(Y = y|\Theta = \theta)$, which depends on some parameter vector θ. The Θ is the random variable corresponding to the model parameters, as these are not observed. And in Bayesian statistics all unobserved variables are treated as random variables. For an MSVAR, the y are the observed variables. The θ consists of the model parameters, which are the Markov chain parameters, the initial distribution and transition matrix, and the VAR coefficients for each state. On the parameter random variable Θ one should have a prior distribution $g(\Theta)$.

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_0 & \mathbf{y}_1 & \cdots & \mathbf{y}_T \end{bmatrix}^T, \tag{2.40}$$

$$\theta = \begin{bmatrix} \text{vec}(\pi_0)^T & \text{vec}(P)^T & \text{vec}\left(A_1^{(1)}\right)^T & \cdots & \text{vec}\left(A_p^{(S)}\right)^T & \text{vec}\left(\Sigma^{(1)}\right)^T & \cdots & \text{vec}\left(\Sigma^{(S)}\right)^T \end{bmatrix}^T \tag{2.41}$$

where the subscripts are for either timestamp or lag, the superscripts are for the different states. vec$(\cdot)$ is the operator that stacks the columns of a matrix in a vector. One wishes to obtain the distribution on the parameters given the observations.

$$g\left(\Theta|Y = \mathbf{y}\right) = \frac{f\left((Y = \mathbf{y}|\Theta = \theta\right) g\left(\Theta\right)}{f\left(Y = \mathbf{y}\right)} \propto f\left(Y = \mathbf{y}|\Theta = \theta\right) g\left(\Theta\right) \tag{2.42}$$

There are several problems in the case of an MSVAR. The model is a multivariate normal (if Gaussian noise is assumed), but only given the latent states and past observations.

$$f(Y_t = y_t|\Theta = \theta, X_t = j, Y_{1:t-1} = \mathbf{y}_{1:t-1}) = \mathcal{N}\left(\sum_{i=1}^{p} A_i^{(j)} y_{t-i}, \Sigma^{(j)}\right) \tag{2.43}$$

finding the distribution $f\left((Y = \mathbf{y}|\Theta = \theta\right)$ is generally not tractable. The same holds for $f\left(Y = \mathbf{y}\right)$, for which one needs to integrate the prior distribution out of the conditional distribution, which results in a very high-dimensional integral. There are better methods that directly sample from the posterior distribution, $g\left(\Theta|Y = \mathbf{y}\right)$. In the next subsection, these methods will be explored in general. Subsequently, these methods will be applied to Bayesian estimation of MSVAR.

2.5.2. MARKOV CHAIN MONTE CARLO

Markov Chain Monte Carlo(MCMC) is a class of algorithms to sample from a posterior distribution in Bayesian inference. A Monte Carlo method is a method for stochastic simulation. In a Markov Chain Monte Carlo method a Markov chain is simulated. However, the Markov chain is chosen such that the stationairy distribution is equal to the posterior distribution [43]. The Markov chain viewed as a sample will converge in probability to the posteriord distribution.

A Markov chain may reach a stationairy distribution π^*. A Markov chain has a stationairy distribution if it is irreducible and aperiodic [43]. There is a unique stationairy distribution, when the detailed balance equation holds.

$$P(j,k)\pi_j^* = P(k,j)\pi_k^* \text{ for } j,k \in \{1,\ldots,S\} \tag{2.44}$$

METROPOLIS-HASTINGS ALGORITHM

The basic MCMC method is the Metropolis-Hastings algorithm, which assumes that one knows the posterior distribution up to a constant, which is the case whenever one knows the model and the prior [44, 45] (see equation 2.42) . It is then not necessary to calculate the normalization constant, which usually is a very high-dimensional integral. Thus one wishes to draw from a distribution $p(x) = \frac{f(x)}{C}$, where C is unknown. One should have a proposal distribution $q(x_1, x_2)$, which is a Markov kernel. Then one can apply the Metropolis-Hastings algorithm as depicted in Algorithm 1.

Algorithm 1 Metropolis-Hastings algorithm

Choose an arbitrary point x_0 with $f(x_0) > 0$
for i=1, ..., N **do**
  Draw a candidate point x^* from the proposal distribution, given x_0;
  Calculate the acceptance probability $\alpha \leftarrow \min\left(\frac{f(x^*)q(x^*,x_{t-1})}{f(x_{t-1})q(x_{t-1},x^*)}\right)$
  Draw a random number, u, from the standard uniform distribution
  if $u \leq \alpha$ **then**
    $u \leftarrow x^*$;
  else
    $u \leftarrow x_{i-1}$;
  end if
end for

By choosing α this way, the kernel q is corrected to have p as a stationairy distribution. This can be shown by evaluating the detailed balance equation. In calculating α it is used that when one divides two evaluations of p that the unknown constant can be left out. Generally, the first part of the resulting chain largely depends upon the starting value x_0, whereas it takes time to converge to the stationairy distribution. It is therefore common to throw away the first few samples. This is called the burn-in period.

For the proposal kernel several options are available, like independent from the previous value or a normally distributed around the previous value. A good proposal kernel finds a balance between exploration of the entire distribution space and a high acceptance rate.

The Metropolis-Hastings algorithm assumes the posterior distribution to be known up to a constant. However, in the case of a MSVAR this distribution is only known by adding more conditional dependencies. The Metropolis-Hastings is thus not appropriate for Bayesian estimation of a MSVAR. The Gibbs sampler is and it will be discussed next.

GIBBS SAMPLER

Another MCMC method is the Gibbs Sampler [46]. This method creates a Markov chain, where each draw consists of several partial updates of the sample. For each draw, all the partitions of the parameter set are updated sequentially, but conditional on the other parameters. This is beneficial when the distributions are known conditional on subsets of parameters. The Gibbs sampler is a special case of the Metropolis-Hastings algorithm, where the acceptance probability is always one. Given partitions of the parameters set $\theta = [\theta^{(1)}, \ldots, \theta^{(n)}]$ the Gibbs sampler works as in Algorithm 2

Algorithm 2 Gibbs sampling algorithm

Get an initial value θ_0 with $f(\theta_0) > 0$
for i=1, ..., N **do**
  for j=1, ..., n **do**
    $\theta_i^{(j)} \sim p\left(\Theta^{(j)} \middle| Y = \mathbf{y}, \Theta^{(-j)} = \left[\theta_i^{(1:j-1)}, \theta_{i-1}^{(j+1:n)}\right]\right)$
  end for
end for

For each of the partitions one can have a different method for obtaining a sample. One can have a conjugate prior just for a specific partition or one can use a single Metropolis-Hastings step for a partition, when one has the conditional distribution for that partition up to a constant. At the end of this section it will be shown that a MSVAR can be estimated with a Gibbs sampler, by adding the states as extra parameters to be estimated. The problem will be decomposed into three components: sampling the states from the Markov chain, sampling new VAR coefficients and sampling the properties of the Markov chain.

2.5.3. SAMPLING STATES

As discussed in the previous section the state variable, $\mathbf{x}$ will be added to the parameter space. In a Gibbs sampler iteration it is needed to sample all the states given the other parameters and the observations. [47] Mentions several methods for sampling states. Forward-filtering-backward-sampling(FFBS) seems to have the best theoretic properties in general, thus it is used in this research. FFBS is a form of multi-move sampling. It is similar to the forward-backward algorithm as discussed in 2.4.2, but instead of updating a distribution in the backward iteration, one samples a state. For iteration m of the Gibbs sampler, one should first run a filter, to obtain a distribution over the state space at each time given the past observations and current parameter estimates, $\mathbb{P}(x_t = j|\mathbf{y}_{1:t}, \theta)$. Next, sample $x_T^{(m)}$ the final state of the Markov chain, from the filtered probabilities.. Then for $t = T-1, T-2, \ldots$, sample $x_t^{(m)}$ from the conditional distribution

$$\mathbb{P}\left(x_t = j|S_{t+1}^{(m)}, \mathbf{y}_{1:t}, \theta\right) = \frac{P_{j,l}^{(m)} \mathbb{P}\left(x_t = j|\mathbf{y}_{1:t}, \theta\right)}{\sum_{i=1}^{S} P_{i,l}^{(m)} \mathbb{P}\left(x_t = i|\mathbf{y}_{1:t}, \theta\right)} \tag{2.45}$$

$\mathbb{P}\left(x_t = i|\mathbf{y}_{1:t}, \theta\right)$ are the filtered state probabilities calculated at the start of the iteration. This algorithm provides a way of sampling the states given the parameters and the observations. The prior is implicitly set to the uniform distribution on the states. There

is no reason to have a prior preference for any of the states. In an algorithm one would sample the states at iteration m of a Gibbs sampler as in Algorithm 3

Algorithm 3 Sampling states with FFBS

for m=1, ... M **do**
  $\eta_t(i) = \mathbb{P}(Y_t = y_t | X_t = i, \theta^{(m-1)})$ for $i = 1, \dots S$
  ▷ Run Hamilton filter
  for t=1, ..., T **do**
    $\hat{\xi}_{t|1:t}(i) = \frac{\hat{\xi}_{t|t}(i) \cdot \eta_t(i)}{\sum_{j=1}^{S} \hat{\xi}_{t|t}(j) \cdot \eta_t(j)}$ for $i = 1, \dots, S$
    $\hat{\xi}_{t+1|1:t} = P\hat{\xi}_{t|1:t}$
  end for
  $x_T^{(m)} \sim \hat{\xi}_{T|1:T}$
  for t=T-1, ..., 1 **do**
    $x_t^{(m)} \sim \mathbb{P}\left(S_t = j | S_{t+1}^{(m)=l}, \mathbf{y}_{1:t}, \theta^{(m-1)}\right) = \frac{P_{j,l}^{(m)} \mathbb{P}\left(S_t = j | \mathbf{y}_{1:t}, \theta^{(m-1)}\right)}{\sum_{i=1}^{S} P_{i,l}^{(m)} \mathbb{P}\left(S_t = i | \mathbf{y}_{1:t}, \theta^{(m-1)}\right)}$
  end for
end for

2.5.4. SAMPLING VAR COEFFICIENTS

Each VAR can be separated into two sets of parameters, the coefficients, $A = [A_1, \dots, A_p]$, and covariance of the noise, Σ. There are four suitable priors for a VAR [48]:

- A normal prior for the A and Σ fixed
- A non-informative prior for A and a non-informative prior for Σ.
- A normal prior for A and a non-informative prior Σ.
- A conditionally conjugate (normal) prior for A and a Inverse-Wishart prior for Σ

The last option will be used. The normal prior is conditionally on Σ conjugate for the VAR coefficients. First, one should choose a prior mean, $\bar{A}$ and covariance Σ_A, such that $\alpha \sim \mathcal{N}(\bar{A}, \Sigma_A)$. A special case for this specification is the Minnesota prior [49]. In the Minnesota prior the process is assumed to be like a random walk. The mean of the multivariate normal is 1 for the coefficients at lag one of the same variable and zero for the others. In expectation this results in a VAR that predicts the previous values as the next value. The covariance matrix of the prior is chosen to be diagonal. Given the parameters ϕ_0, ϕ_1, ϕ_2 and ϕ_3, the lag function is $h(l) = l^{\phi_3}$, the diagonal elements of σ_A with i the equation number and l the lag of the dependent variable, j, are as follows:

$$\sigma_{i,j,l} = \begin{cases} \frac{\phi_0}{h(l)} & \text{if i=j} \\ \phi_0 \cdot \frac{\phi_1}{h(l)} \left(\frac{sigma_j}{\sigma_i} \right)^2 & \text{if } i \neq j \text{,but endogenous} \\ \phi_0 \phi_2 & \text{if } j \text{ exogenous} \end{cases} \tag{2.46}$$

with σ_k the sample standard deviation of variable k. The lag function describes how the freedom changes with the lags. ϕ_0 describes the general prior strictness. ϕ_1 describes the relative strictness for endogenous coefficients and ϕ_2 describes the relative strictness for exogenous coefficients.

The Inverse-Wishart distribution is the standard prior for the covariance matrix of a multivariate normal distribution. The Inverse-Wishart distribution has two parameters, ν, the degrees of freedom and Φ, the scale matrix. For $\nu \geq= p$ the draw from the distribution is invertible almost surely, whereas for for $\nu < p$ that is not guaranteed. The mean of the Inverse-Wishart distribution is $\frac{\Phi}{\nu-p-1}$, given a prior guess of the covariance matrix Σ_0, Φ should be set to $\Phi = (\nu - p - 1)\Sigma_0$.

For both these priors are (conditionally) conjugate their updates can be derived and are as follows: If $g(\alpha = A) = \mathcal{N}(\bar{A}, \Sigma_A)$. Define

$$\tilde{\alpha} = \left[\tilde{\Sigma}_A^{-1} + \left(\Sigma^{-1} \bigotimes X'X\right)\right]^{-1} \left[\tilde{\Sigma}_A^{-1}\bar{A} + \left(\Sigma^{-1} \bigotimes X\right)^T y\right] \tag{2.47}$$

here $\tilde{\Sigma}_A = \left[\Sigma_A^{-1} + \left(\Sigma^{-1} \otimes X^T X\right)\right]^{-1}$, and than the normal distribution $\mathcal{N}(\tilde{\alpha}, \tilde{\Sigma}_A)$ is the posterior. For the inverse Wishart, we have $\tilde{\nu} = n + \nu$ and $\tilde{\Phi} = \Phi^{-1} + \text{SSE}$, leading to a new Wishart distribution, from which a covariance matrix can be sampled.

In conclusion, the following hyperparameters should be chosen:

ϕ_0 the strictness of the Minnesota prior for the VAR coefficients,

ϕ_1 the relative strictness for endogenous coefficients,

ϕ_2 the relative strictness for exogenous coefficients,

ϕ_3 the power of the lag function in the Minnesota prior,

ν the freedom of the Inverse-Wishart prior, measures the confidence in prior information,

Σ_0 the base VAR covariance.

2.5.5. SAMPLING MARKOV CHAINS

The last set of parameters that should be sampled in a Gibbs sampler for MSVAR are the properties of the Markov chain. In [47] section 11.5.5 two cases of estimation are discussed for stationairy and non-stationary Markov chains. The state changes during a surgery seem to form a non-stationary Markov chain, because it does not seem the cases that states are equally likely at the start and end of a surgery. The rows of the transition matrix, P, are a multinomial distribution for the next state, given a current state, and are independent. The Dirichlet distribution is a conjugate distribution for the multinomial distribution. Given a matrix of prior weights e, where e_{ij} is the prior weight for P_{ij} the posterior distribution is

$$\tilde{P}_j \sim \mathcal{D}\left(e_{j1} + N_{j1}(\mathbf{x}), \ldots, e_{jS} + N_{jS}(\mathbf{x})\right), \qquad j = 1, \ldots, S \tag{2.48}$$

where N_{ij} is the function that counts the number of transitions from state i, to j, $N_{ij}(\mathbf{x}) = \#\{x_{t-1} = i, x_t = j, t = 1, \ldots, T-1\}$. The initial distribution is also a multinomial distribution and can be handled similarly to a row of the transition matrix. Thus, one needs to

determine the weights for the transition matrix and initial distribution prior Dirichlet distributions. There is no reason for any prior preference for any of the states, as these are abstract. This has two consequences. First, the weights for the initial distribution should all be equal, with only a hyperparameter for the total weight, i.e. the confidence. Another consequence is that for the transition matrix, there are only two types of coefficients, diagonal, called e_1, and off-diagonal, e_2. These two hyperparameters will be reparameterized into more informative hyperparameters. Given a transition matrix, P, the expected consecutive time in a state i is $\frac{1}{1-p_{ii}}$ as the expectation from a geometric distribution. The expectation of a Dirichlet distribution is equal to the normalised weights. If a certain expected consecutive time (ect) is desired, it can be enforced by setting:

$$e_1 = \text{confidence} \cdot (1 - \frac{1}{\text{ect}}) \tag{2.49}$$

$$e_2 = \text{confidence} \cdot (\frac{1}{\text{ect} \cdot |S|}) \tag{2.50}$$

This leaves three hyperparameters for determining the priors of the Markov chain:

- the expected consecutive time in the same state,
- the confidence in the transition matrix prior,
- the confidence in the initial distribution prior.

2.5.6. GIBBS SAMPLER FOR MSVAR

The previous sections include all the necessary ingredients for the Bayesian estimation of a Markov-switching vector autoregression. The idea is to make a Gibbs sampler, where each of the substeps: sampling states, sampling VAR coefficients, and sampling Markov chain parameters are used.

Suppose the following setting:

- Hyperparameters for all the priors.
- Given a sequence of observed vectors $\mathbf{y}_t$, $t = 1, \ldots, T$, of length d, the dimension of the time series.
- a prior mean, $\bar{\alpha}_j$ and covariance $\Sigma_{\alpha,j}$, for the VAR coefficients for the states $j = 1, \ldots, S$,
- a initial estimate for the VAR covariance $\Sigma_{0,j}$ for each state $j = 1, \ldots, S$, leading to a Wishart parameter $V_j = n^{-1}\Sigma_{0,j}^{-1}$,
- a matrix e of weights for the prior Dirichlet distributions of the transition matrix and a vector $\mathbf{e}_0$ of weights for the prior Dirichlet distribution of the initial probability,
- an initial sample of $\Theta = \theta$, the vector with all parameters of the MSVAR model, with prior probability greater than zero,

- a burn-in period M_0 and a requested number of samples M,

then the following one can apply the specific Gibbs sampler as developed in the past sections for the estimation of an MSVAR (see Algorithm 4). This algorithm is a Gibbs sampler (Algorithm 2) with in each iteration the FFBS algorithm (Algorithm 3), sampling of the VAR coefficients, and sampling of the Markov chain parameters.

Algorithm 4 Gibbs sampler for MSVAR

for $m = 1, \ldots, M + M_0$ **do**
  $\eta_t(i) = \mathbb{P}(Y_t = y_t | X_t = i, \theta^{(m-1)})$ for $i = 1, \ldots S$
  ▷ Run Hamilton filter
  for t=1, . . . , T **do**
    $\hat{\xi}_{t|1:t}(i) = \frac{\hat{\xi}_{t|t}(i) \cdot \eta_t(i)}{\sum_{j=1}^{S} \hat{\xi}_{t|t}(j) \cdot \eta_t(j)}$ for $i = 1, \ldots, S$
    $\hat{\xi}_{t+1|1:t} = P\hat{\xi}_{t|1:t}$
  end for
  $x_T^{(m)} \sim \hat{\xi}_{T|1:T}$
  for t=T-1, . . . , 1 **do**
    $x_t^{(m)} \sim \mathbb{P}\left(S_t = j | S_{t+1}^{(m)=l}, \mathbf{y}_{1:t}, \theta^{(m-1)}\right) = \frac{P_{j,l}^{(m)} \mathbb{P}\left(S_t = j | \mathbf{y}_{1:t}, \theta^{(m-1)}\right)}{\sum_{i=1}^{S} P_{i,l}^{(m)} \mathbb{P}\left(S_t = i | \mathbf{y}_{1:t}, \theta^{(m-1)}\right)}$
  end for
  for j=1, . . . , S **do**
    $\tilde{\Sigma}_{\alpha,j}^{-1} = \left[\Sigma_{\alpha,j}^{-1} + \left(\left(\Sigma_j^{-1}\right)^{(m-1)} \otimes X^T X\right)\right]$
    $\tilde{\alpha}_j = \left[\tilde{\Sigma}_{\alpha,j}^{-1} + \left(\left(\Sigma^{-1}\right)^{(m-1)} \otimes X'X\right)\right]^{-1} \left[\tilde{\Sigma}_{\alpha,j} \bar{A} + \left(\left(\Sigma^{-1}\right)^{(m-1)} \otimes X\right)^T y\right]$
    $A_j^{(m)} \sim \mathcal{N}\left(\alpha_j, \tilde{\Sigma}_{\alpha,j}\right)$
    $\tilde{n} = n_j + \nu$, $\nu = |\mathbf{y}|$
    $\tilde{V} = \left(V_j^{-1} + SSE\left(\theta^{(m)}, S_{1:T}\right)\right)^{-1}$
    $\left(\Sigma_j^{-1}\right)^{(m)} \sim \mathcal{W}_d(\tilde{V}, \tilde{n})$, where SSE computes the sum of squared errors given the appropriate VAR model for each observation.
  end for
  $\pi_0^{(m)} \sim \mathcal{D}(e_1 + N_1(x_0), \ldots, e_S + N_S(x_0)$
  $P_j^{(m)} \sim \mathcal{D}\left(e_{j1} + N_{j1}(x_0), \ldots, e_{jS} + N_{jS}(x_0)\right)$ for $j = 1, \ldots S$
end for

2.6. EVALUATION OF TIME SERIES MODELS

In most cases, time series models are fitted using maximum likelihood estimation (MLE). The likelihood function is the joint probability distribution of the data given parameters, which is thus a function of the parameters. In maximum likelihood estimation the maximum of this function is found and the corresponding parameters are considered the best estimates for the parameters. The likelihood of a given model is the joint probability of all the data given the model. The likelihood is a measure of how well the model fits the

data. Usually, the log of the likelihood is taken because it is easier to compute and the monotone transformation makes sure that the order is maintained.

2.6.1. PENALIZED LIKELIHOOD

The (log)likelihood is calculated on the same data is that was used for model estimation. This is especially a problem when using the maximum likelihood estimator, as the models is chosen to optimize the likelihood, which makes the likelihood less suitable as a measure of the model performance. When adding extra parameters or variability to the model, the likelihood will always increase when using the MLE. However, this easily leads to overfitting. It is therefore either good to split the data in fitting and evaluation parts to get out of sample estimates of model performance. Or, as is more common in time series analysis, use the likelihood, but compensate for the amount of parameters. AIC and BIC are two of such generally used measures.

The AIC is an estimator of the prediction error based on information theory [50]. The result is the likelihood penalised for the number of free parameters.

$$\text{AIC} = 2k - 2\log\left(\hat{L}\right) \tag{2.51}$$

where k is the number of free parameters and $\hat{L}$ is the maximum likelihood value. Similarly, an information criterion can be established based on Bayesian arguments [51].

$$\text{BIC} = k\log(n) - 2\log\left(\hat{L}\right) \tag{2.52}$$

again k is the number of free parameters, $\hat{L}$ is the maximum value of the likelihood, and n is the number of observations. Both these information criterion are suitable for comparing model performance among a set of different models on the same data.

3

METHOD

The goal of the thesis is to predict postoperative hsTnT. In this chapter, it will be shown how the theory in the previous chapter can be applied to the Erasmus MC database to obtain troponin T predictions. First, it will be discussed how to get from time series data to features for each surgery. It will turn out that some preprocessing is necessary and that will be discussed next. Finally, some models for actually predicting troponin from features and their properties will be shown.

3.1. FEATURES FROM TIME SERIES

In the previous chapter a lot of time series models were discussed. However, the main idea is to get predictors from the time series to predict the hsTnT. This is often not trivial. Time series usually do not have the same length, because surgeries have different lengths for instance. Next to that, inputting the raw time series values in a model disregards their interactions and time dependence. The values mainly mean something with regard to their timestamp. One wants time series that are similar to lead to similar predictions. However, similarity of time series can be interpreted in a lot of different ways. One wants to find a measure of similarity such that time series with similar corresponding troponin T values are seen as similar. Here, three approaches are discussed for generating predictors from time series. All these methods map the time series to a finite-dimensional vector space, where time series that are close in some sense are also close in the new space. Next, a novel approach used in similar context is discussed and finally it is shown how this approach can be used in the context of predicting troponin.

3.1.1. KNOWN APPROACHES FOR EXTRACTING FEATURES FROM TIME SERIES

The first approach is applying statistical functionals to the time series. The idea is to apply functions like, mean, max and min to each of the time series and stack the values in a vector. This leads to a vector representation of the time series. There are many more functions and these can also be domain specific. This method was applied for predicting

troponin in [21]. Such a domain-specific function is the mean value of one time series while the other was under a relevant medical threshold. This approach does not model the time series with some model, but uses the raw values directly. In [21] this approach was used to predict troponin T from time series consisting of heart rate, blood pressure and ST segments. An ROC AUC score in predicting high troponin T (>50 ng/L) of about 0.75 was reached.

Another approach is to use the predictions made by any time series model for an unknown future as features. For example, in [20], a time series with heart frequency, blood pressure, and ST segments was modelled with a VECM. Then, predictions were made for each of the 5 minutes following surgery. In predicting high troponin T(> 50 ng/L) a balanced accuracy of 58% was achieved. This is only marginally better than random guessing. Instead of a VECM, any model can be used and any number of predictions. However, often predictions will be very correlated. Next to that, there are many cases were patients are observed longer than the surgery. Building a model for prediction does not seem necessary then.

The final approach that will be discussed here is similar to the previous in that a model is fitted to the time series. Instead of using predictions, one uses the fitted coefficients of the model as a vector representation of the time series. Again the features will be very correlated. This approach was successfully employed in [52]. There principal component analysis was used to reduce the correlation among features [53]. Especially combined with principal component analysis the features are hard to interpret.

In the setting of predictions from hemodynamic modelling another approach was published quite recently and it will be extensively discussed next.

3.1.2. STATE TIMES FROM MSVAR

In a series of papers Lehman et al. discussed the usage of Markov-switching Vector Autoregression in hemodynamic modelling and predicting mortality from these models [25, 26, 27], these papers will be referred to as the ICU-MSVAR papers. Their approach on ICU data is transferable to the surgery data and is therefore discussed here.

All there work describes a similar procedure, which will be viewed in general here. The authors extracted minute-by-minute heart rate and blood pressure data from 453 patients while on the intensive care unit (ICU) from MIMIC II [54]. Next, an Markov-Switching Vector Autoregression, see section 2.4, was fitted to the data. Then, the relative time each patient was in each state was extracted from the model fit. These relative times were used as features in predicting 30-day all-cause mortality with logistic regression. Next to that, they were able to use this method to generate a live risk quantification. Their 30-day all-cause mortality estimate was more accurate than standard ICU risk score [25] and was shown to have additional prognostic value [27]. From the fitted logistic regression one can derive which states are low-risk (decrease in mortality probability) and which are high-risk (increase in mortality probability), making this a good interpretable approach.

The authors claim their success is because the hemodynamics have rich dynamical structures, as it is part of a feedback control system. The MSVAR is able to capture the complexity. Next to that, patients can be compared by their shared states, meaning that their hemodynamics can be modelled with the same Vector Autoregression.

3.1.3. DIFFERENCES WITH ICU-MSVAR PAPERS

The approach of extracting state time from an MSVAR in hemodynamic modelling is very relevant, however not the same. In this section the differences will be discussed. First of all, ICU data was used, while in the troponin prediction cases there is surgery data. In both cases patients are strictly monitored and often under a lot of stress. It can be argued that patients in the ICU are generally in a more stressful state. Secondly, the authors selected their patients to have at least 24 hours of hemodynamic monitoring data. Surgeries generally do not last that long, thus the available time series will be shorter in general. Next to that, minute-by-minute data was used. There is no minute-by-minute heart rate and blood pressure data. It is measured irregularly at intervals of a few minutes. As noted in [19] one needs time series data, which is equidistant for time series modelling anyway and the easiest distance to use is one minute. In the next section, it will be shown how interpolation can be used to get a minute-by-minute time series. Finally, the authors use the relative state times as features for predicting mortality, whereas here the goal is to predict troponin T. That is mainly the case because luckily the mortality rate after surgery is not high enough to have enough relevant data. The authors themselves note that next to predicting mortality many other variables like the event of hemodialysis, severe sepsis and readmission can be used as endpoints [26]. Tropinin T measurements are another event that is happening after the surgery and might therefore be predicted from the time series with this approach.

To summarize, this approach can be adapted for predicting troponin T. After preprocessing minute-by-minute hemodynamic time series can be obtained. This time series can be modelled with a MSVAR. Next, the proportion of time each patient was in a certain state can be extracted from the model. This maps the time series to a vector space from which one can try to predict tropnin T.

3.2. INTRAOPERATIVE DATA PREPROCESSING

As mentioned in the previous section, it is necessary to interpolate the time series data to obtain a time series appropriate for modelling. But before interpolation, the outliers should be removed from the time series. And the time series should also be checked to meet certain quality standards. The four parts of time series preprocessing, filtering, checking, interpolating and selecting, are discussed in this section. Most of the ideas are based on [19], who worked on the same time series and encountered similar problems.

3.2.1. FILTERING

In the time series, there are mainly point outliers, single values that do not follow the general pattern. For instance, for the arterial blood pressure it is known that at random times very high values are measured, because the patient is moved. Next to that, outliers occur because of incorrect recording. In figure 3.1 one can see example of such point outliers. There are two instances, where there are very high values just for one time point. This motivates two types of filtering

First, a simple threshold will be applied. There are values that are clearly wrong, like negative or very small heart rate. Similarly, there are values that are unrealistically high. For each time series, upper and lower bounds will be determined. The bounds can be

variable	minimum value	maximum value	maximum difference actual and interpolated
heart rate	30	200	30
blood pressure	10	300	30

Table 3.1: The minimum, maximum and the max difference between the interpolated and actual value for each of the variables.

check	value
Minimum number of observations	30 observations
Maximum time no observations	15 minutes
Minimum observations density	1 every 5 minutes

Table 3.2: The specific values used for the three checks in checking the suitablity of intraoperative data for modelling.

found in Table 3.1. Secondly, each point will be considered an outlier if it does not follow the trend at that point. Each point is estimated by linearly interpolating the surrounding points. The value is thrown away whenever the difference between the interpolated value and the actual value are too high (see Table 3.1). In figure 3.1 one can see that some blue values do not have an orange plus next to them, which means that these values are filtered out.

3.2.2. CHECKING

After filtering, one needs to make sure that enough good values are in the time series. Three criteria are determined which the time series should adhere to be able to reliably fit a model to the data. Generally speaking, when only a relatively small amount of observations is present, one is fitting a model to interpolated values instead of the observations.

The first criterion is that the time without observations is limited. When there are no observations for more than this period, it makes no sense to fit a model. Secondly, the time series should have a minimum observation density. That is, on average there should be at least that much observations per minute. Finally, there should be an overall minimum number of observations. This mainly excludes very short time series. The actual values used are in Table 3.2 The result of the checks can be found in figure D.1.

3.2.3. INTERPOLATION

The next step of time series preprocessing is interpolating the values. One can only fit a time series model to an equidistant time series. The time series is interpolated to have a value for each minute. As the VAR is a linear model, one should probably avoid linear interpolation as the model would probably overfit on the interpolation. The simplest non-linear interpolation is quadratic spline interpolation, which is what will be used. Spline interpolation was introduced by [55]. In quadratic spline interpolation a quadratic function is fitted between two observations, using that the values at the endpoints should

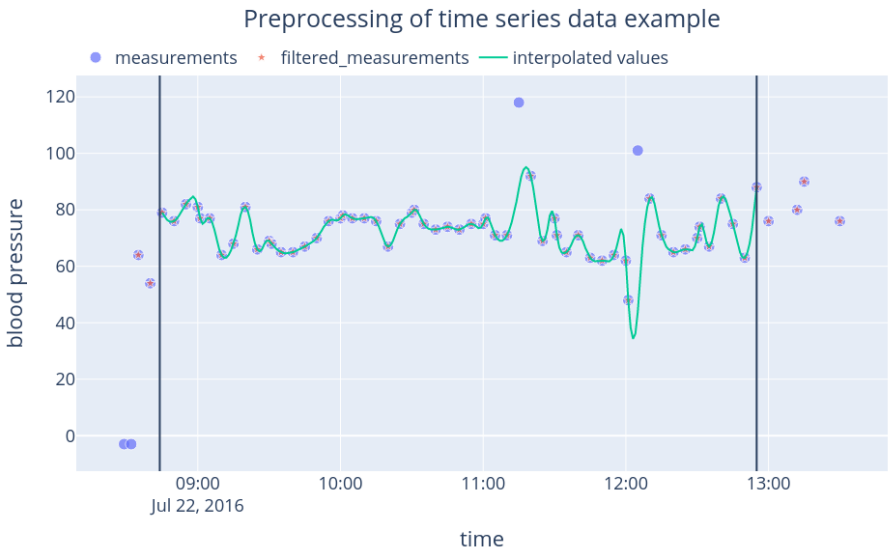


Figure 3.1: Example of processing time series, with filtering, interpolation and selection. The blue dots indicated the measurements. The blue dots with a red cross are the measurements that are kept after filtering. The two at the start are below zero and thus obvious outliers. There are also two separate points at about 11:05 and 12:03 that are considered outliers. The filtered values are interpolated, which is indicated by the green line. The black lines indicate the operation room arrival and departure. Only data within this interval is selected.

correspond to the observations and the derivatives should be continuous. This induces a series of linear equations, of which the solution gives the coefficients for a piece-wise quadratic function. The function can be used to determine the value at the times for which no (valid) observation is present. For the implementation the FITPACK is indirectly used [56, 57]. In figure 3.1 the resulting interpolation is shown.

3.2.4. SELECTION

The final step is selecting the data corresponding to the surgery. Usually the hemodynamic data is only registered with high frequency around the surgery. To treat all surgeries equal the following choice was made. The surgery is considered to start at the time noted as the operation room arrival in the or room and to end at the time noted as the operation room departure. This thus includes the waiting in the operation room, the induction of the patient and if the case the awakening of the patient. An example of the selection is indicated with the black lines in figure 3.1.

3.3. TROPONIN PREDICTION MODELS

Given the relative state times as extracted from a MSVAR model, one can predict troponin using the prediction model. In [25] logistic regression was used for predicting mortality, but other options are available. First, one has to decide whether to do regression or classification, because with troponin one can do both. There are known classes

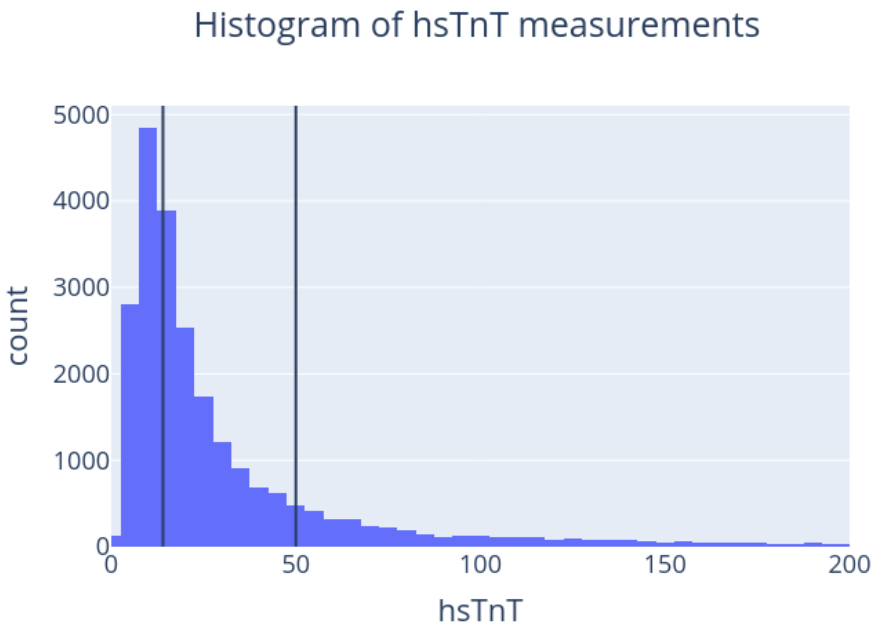


Figure 3.2: Distribution of hsTnT values. The part for the first black line is considered low hsTnT. The region between the two black lines is considered elevated hsTnT and the last part is considered high hsTnT.

of troponin values:

$$\begin{cases} \text{low hsTnT} & 0 \leq \text{hsTnT} \leq 14, \\ \text{elevated hsTnT} & 14 < \text{hsTnT} \leq 50, \\ \text{high hsTnT} & 50 < \text{hsTnT} \end{cases} \tag{3.1}$$

One could predict these classes, or just whether it is high troponin and binary classification. As classification seems more medically relevant, it is decided to do classification. Two different classification models are considered to distinguish the kind of effects present in the data. First, logistic regression as in [25] is described, followed by the alternative model, random forest.

3.3.1. LOGISTIC REGRESSION

Linear models are the simplest models in general, because calculations with linear functions are easy. The simplest linear classification model is logistic regression. In contrast to what the name suggests, this is a classification model. It is simple, linear in the features and widely used [58]. Given a set of samples with features, X, and labels, y, being 0 or 1, the probability of the label one belonging to the sample is modelled by expressing the log-odds as a linear function of the features [59]. The log-odds are another way to

express a probability (equation 3.2).

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) \tag{3.2}$$

$$p_k = \frac{1}{1-e^{-\beta x_k}} \tag{3.3}$$

The coefficients (β in equation 3.3) describing the linear dependence of the log-odds on the features are usually chosen as the maximum likelihood estimate or equally by minimizing the negative log-likelihood. The negative log-likelihood or the surprise for sample k is given in equation 3.4.

$$ll_k = \begin{cases} -\log p_k & \text{if } y_k = 1 \\ -\log(1-p_k) & \text{if } y_k = 0 \end{cases} \tag{3.4}$$

The closer that p_k is to y_k the less the surprise and thus the better the model. The total negative likelihood is obtained by summing over all samples. There is no analytical solution for finding the minimum, however numerical solutions are available.

Logistic regression is simple and easily interpretable as there is a single coefficient for the effect of each feature. The con of the model is that there is no interaction between the different features. In predicting postoperative troponin T, the labels would be whether the postoperative hsTnT is high (> 50 ng L^{-1}). The features would be the relative time in each of the states.

3.3.2. RANDOM FOREST

To see the effect of the model in classifying high postoperative troponin T, the results obtained with logistic regression are compared to another relatively simple classification model, namely, random forests. Random forests are an ensemble of decision trees [60]. First, decision trees will be explained. Second, the ensemble of multiple of these trees is explained.

A decision tree is a binary tree, where each node represents a decision on the features of the sample [61]. Each decision splits the set of samples in two subsets. The decision tree thus recursively subsets the sample into smaller subsets. Learning a decision tree from data usually happens with the top-down-induction-of-decision-trees (TIDIT) algorithm [62]. One starts with the full sample and decides upon the best decision based on the features at this point. The best decision is the one that splits the labels as well as possible, which can be defined in many ways. In this thesis, the most common, the Gini impurity, will be used. The Gini impurity measures how often a random sample from a set would be incorrectly labelled, if it was labelled randomly according to the other members of the set. If the decision perfectly splits the labels, the Gini impurity would be 0 as all labels are the same in each set. In the worst case, where each set consists of 50% of both labels, the probability of wrongly labelling a random point is 0.5. One chooses the decision that minimizes the Gini impurity. This is recursively continued for each subset resulting from either side of a decision. The procedure is usually terminated based on some stopping criterion to reduce overfitting. Usual stopping criteria are a maximum tree depth or a minimum number of samples required to split a set. The estimate for the

probability of an unseen sample is the relative number of positive labelled samples in the subset, where the new sample would end up after following the decision tree.

A random forests is an ensemble or collection of these decision trees. The basic idea of ensembling different classifiers is bagging [63]. In bagging different random subsamples are taken from the dataset. For each sample, the classifier (decision tree in this case) is estimated [64]. The idea is that the variance of the resulting classifier is lower. The classifier is less influenced by a single sample and generalizes better over the entire dataset. This is especially the case when the decision trees are uncorrelated. To reduce the correlation among different decision trees, extra randomness is induced in a random forest [60]. For each split, only a random subset of features is considered.The final prediction is the mean of all predictions by the separate decision trees.

4

EM ESTIMATION AND MODEL SELECTION

As discussed in the previous chapter, the idea is to fit a Markov-Switching Vector Autoregression to intraoperative data. One can fit an MSVAR with the EM algorithm (see section refsec: msvar). In this chapter the details of MSVAR estimation and analysis of the convergence will be presented. Next to that, the most appropriate model for modelling intraoperative heart rate and blood pressure will be chosen.

4.1. SPECIFIC IMPLEMENTATION DETAILS EM ALGORITHM

The general idea of fitting an MSVAR to time series data was already discussed in section 2.4.4. In this section, details of the implementation will be described. The EM algorithm is an iterative method that monotonically converges to a maximum of the likelihood function in theory. It should converge to a maximum, but that could also be a local maximum. The claim for monotone convergence does not hold anymore in this case, because the covariance for the VAR models is not estimated with the maximum likelihood estimator, but with a shrinkage estimator. Furthermore, probabilities can become very small, leading to numerical imprecision. Both should have little effect on the resulting model, but convergence is not necessarily monotone anymore.

Given the fitting problem, the following specific implementation details were decided. In an upcoming section, 4.3, the convergence is analysed, and these implementation details are evaluated. First, the maximum iteration time is set to 200 iterations. In theory, the model will improve with each iteration, but changes might be insignificant. In practise, the likelihood might not even increase with it every iteration. It is therefore decided to always stop iterating when no likelihood improvement is made in the last 10 iterations. The EM algorithm converges to a maximum based on the random initialisation. It is decided to try 5 different random initialisations and choose the model that corresponds to the highest found maximum. Randomly restarting the EM algorithm is known as multiple restart EM (MREM) algorithm [65].

Using these implementation details of the EM algorithm, a fitted model was obtained. Analysis of the model showed some unwanted behaviour, which is discussed in the next section.

4.2. ADAPTATIONS

An analysis of the resulting models shows a lot of state transitions. The idea of this research is to interpret the states a patients goes through during a surgery. It is therefore expected that the state a patient is in is quite stable. In this section, this problem is investigated for the model with 5 states specifically. There is no reason why the solution would not work for the models with another number of states.

In Figure 4.1 one can see the probability for each of the states as determined by an MSVAR model. The probabilities frequently switch between zero and almost one and the most probable state is different almost every minute. This is in contrast to what one would expect medically. During surgery, the patient is in a similar state most of the time, with the main changes occurring at the start and end. Switches in the middle of surgery can occur, but should not occur frequently. Figure 4.1 shows an entirely different picture and there is no medical reason to have such frequently occurring state switches. It is therefore desirable to adapt the model and the fit algorithm to have a model that is potentially better medically interpretable.

First, one should start by analysing the transition matrix, because that mainly determines the state transitions. The probability of staying in the same state is equal to the diagonal value of the transition matrix corresponding to the state. The transition matrix for the model is in 4.2. The large probabilities are not necessarily on the diagonal, but all over the place. This explains the often occurring transitions. However, it should be noted that this is a 'chicken and egg' story as the transitions are mostly determined by the transition matrix and the transition matrix is directly estimated from the transitions. To break this loop, shrinkage will be applied to the transition matrix in the E-step of the EM algorithm.

4.2.1. TRANSITION MATRIX SHRINKAGE

The maximum likelihood estimator for the transition matrix is to count all transitions and normalize for each of the 'from' states as more elaborately described in section 2.4.2. The desired transition matrix has an expected consecutive time in a state of about 90 minutes. The consecutive time in a state is a geometric distribution with the probability being the probability of leaving the state, 1 minus the diagonal element of the transition matrix. For an expected value of 90 (minutes), the value on the diagonal should be equal to $1 - \frac{1}{90}$. Suppose that an expected consecutive time of ect is desired. The rest of the

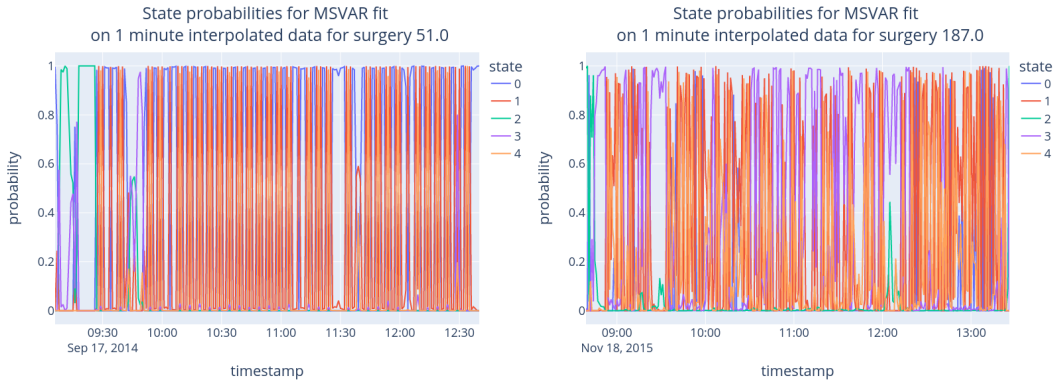


Figure 4.1: State probabilities from the resulting model for an MSVAR with ten states for two surgeries with or codes 51 (left) and 187 (right).

mass can be divided over the other elements of the matrix.

$$P_{\text{desired}} = \begin{bmatrix} 1-\frac{1}{\text{ect}} & \frac{1}{|S|\cdot\text{ect}} & \cdots & \cdots & \frac{1}{|S|\cdot\text{ect}} \\ \frac{1}{|S|\cdot\text{ect}} & 1-\frac{1}{\text{ect}} & \frac{1}{|S|\cdot\text{ect}} & \cdots & \frac{1}{|S|\cdot\text{ect}} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \frac{1}{|S|\cdot\text{ect}} & \cdots & \frac{1}{|S|\cdot\text{ect}} & 1-\frac{1}{\text{ect}} & \frac{1}{|S|\cdot\text{ect}} \\ \frac{1}{|S|\cdot\text{ect}} & \cdots & \cdots & \frac{1}{|S|\cdot\text{ect}} & 1-\frac{1}{\text{ect}} \end{bmatrix} \tag{4.1}$$

One shrinkage estimator [1] for the transition matrix is given by a weighted combination of the MLE and the desired transition matrix, where the weights sum up to one.

$$\hat{P} = \alpha P_{\text{desired}} + (1-\alpha)\hat{P}_{\text{MLE}} \tag{4.2}$$

Using the shrinkage estimator with $\alpha = 0.95$ and ect = 30, the following results are obtained. Note that using a shrinkage with $\alpha = 0.95$ is quite strong, but this makes the effect best visible. In Figure 4.3 the resulting state probabilities over time are visible. The number of transitions has decreased; however, in the figure for surgery 51, many switches are still present. Even more worrying is that there seems to be a pattern in the alternation of state 0 and 3. A closer look shows that this pattern corresponds to the frequency at which the data was recorded. Some points are almost exactly predictable. This leads to a state that has a very small covariance, having low likelihood predictions for most points, but very good predictions for some. The good predictions are so likely that the a switch occurs with probability almost equal to one almost independent of the previous state and transition matrix. The prediction for the timestamp after it is so much worse that the model jumps back to the old state. This results in a state change that corresponds to the interpolation and has no medical meaning, which is not desirable. Therefore, the interpolation is changed.

[1] A shrinkage estimator is an estimator that, either explicitly or implicitly, incorporates the effects of shrinkage. In loose terms this means that a naive or raw estimate is improved by combining it with other information.

$$\begin{bmatrix} \mathbf{0.58} & 0.39 & 0.01 & 0.01 & 0.01 \\ 0.32 & \mathbf{0.11} & 0.01 & 0.17 & 0.39 \\ 0.02 & 0.02 & \mathbf{0.72} & 0.23 & 0.01 \\ 0.02 & 0.21 & 0.11 & \mathbf{0.61} & 0.05 \\ 0.02 & 0.81 & 0.01 & 0.09 & \mathbf{0.07} \end{bmatrix} \tag{4.3}$$

Figure 4.2: State transition matrix belonging to an MSVAR with 5 states, which is the same model as in Figure 4.1. The important diagonal values are made bold.

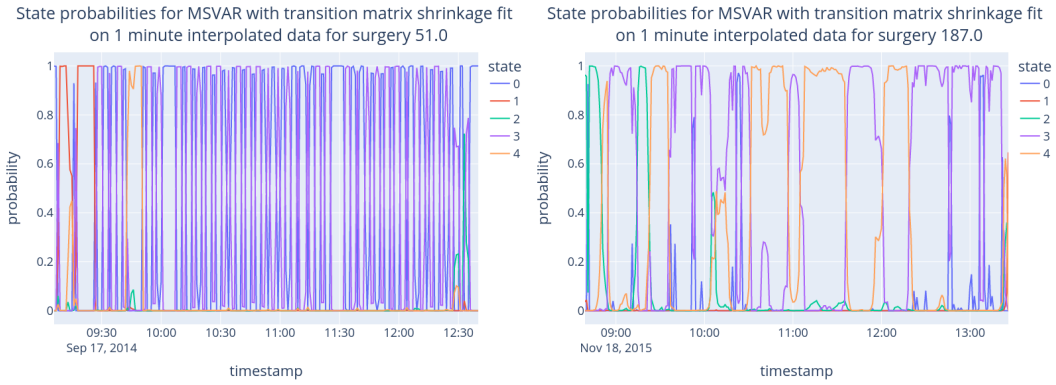


Figure 4.3: State probabilities from the resulting model for an MSVAR with ten states for two surgeries with or codes 51 (left) and 187 (right). In fitting this MSVAR shrinkage is applied to the transition matrix.

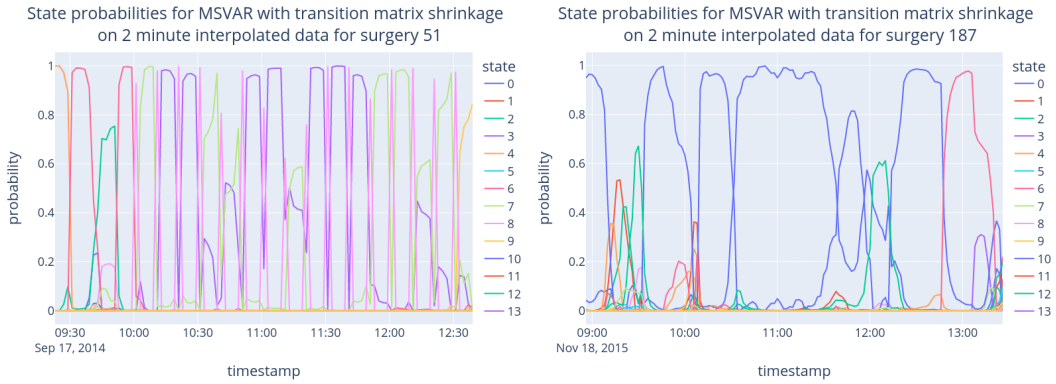


Figure 4.4: State probabilities from the resulting model for an MSVAR with ten states for two surgeries with or codes 51 (left) and 187 (right). In fitting this MSVAR shrinkage is applied to the transition matrix. Furthermore the model is applied to 2 minute cubic spline interpolated data.

4.2.2. DIFFERENT INTERPOLATION

As shown in the previous section, there is a relation between the interpolation of the data to a univariate time series and the state transitions. The model is overfitting on relations that are part of the data processing and have no medical meaning. Therefore, it is decided to change the interpolation. Instead of interpolating to a 1-minute interval, a 2-minute interval is used. That means that there is no series of interpolated points between two actual data points, making the time series not so predictable. Furthermore, the order of the spline interpolation is increased from quadratic to cubic.

The resulting state probabilities, as shown in Figure 4.4, are as desired. There are state switches, but these do not occur very frequently and there is no immediate relation to the time stamps.

4.3. CONVERGENCE

In this section, the convergence of the EM algorithm is analysed. The EM algorithm is an iterative method. However, only a finite number of iterations can be computed. Additionally, the algorithm was implemented with computation time in mind. Reducing computing time is good as long as the algorithm still (almost) converges. The algorithm is meant to optimize for the likelihood, that is therefore the best place to start the analysis [2].

4.3.1. CONVERGENCE OF LIKELIHOOD

In Figure 4.5 the convergence of the likelihood for each of the five runs can be seen. One can see that each run is stopped after 10 iterations without reaching the maximum number of iterations. All runs stop before the iteration limit of 200. Most convergence occurs in the first few iterations after which only small improvements are made. There are dif-

[2] The analysis is done on the convergence of the MSVAR model fitted on 10 lags and using 20 states. This is the most appropriate model as will be shown in section 4.4.

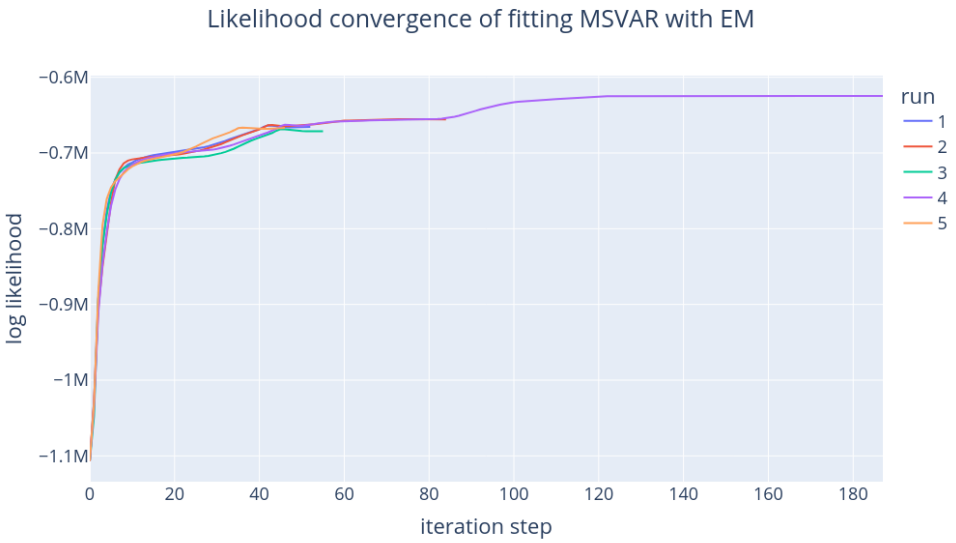


Figure 4.5: Trace of the likelihood over the iterations for 5 runs of fitting an MSVAR with the EM algorithm. Each run the algorithm is initialized randomly.

Table 4.1: Statistics on each of the 5 runs in fitting an MSVAR with the EM algorithm.

run	final log likelihood	number of iterations
1	-663636	53
2	-655258	85
3	-668509	56
4	-624659	188
5	-666496	47

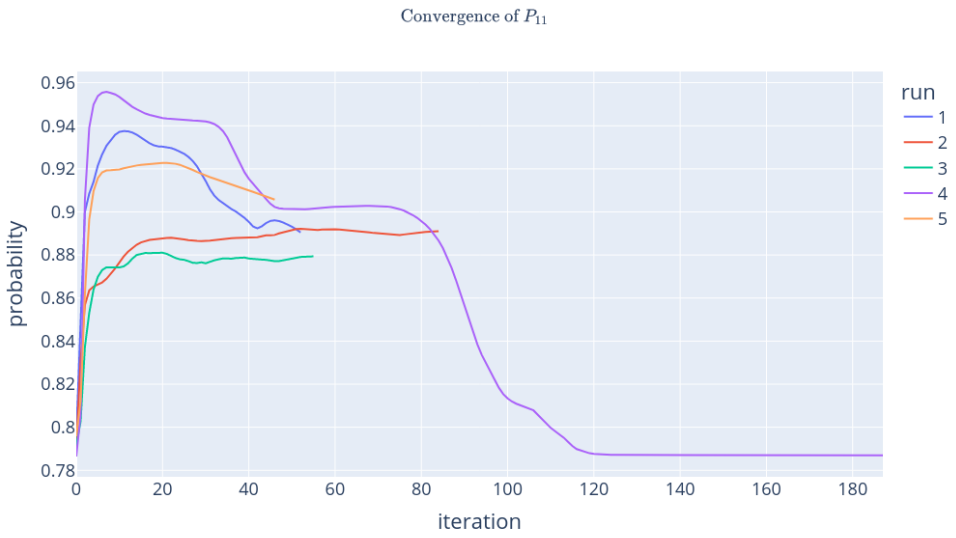


Figure 4.6: Trace of the probability of transitioning from state 0 into state 0 for different runs of the EM algorithm.

ferences between the maximum likelihoods attained, but these are relatively small. One can than wonder whether the models are similar as they are attaining similar likelihoods. This is hard to establish, as a permutation of the states and the corresponding parameters is again the same model, although, when comparing the parameters, the models look entirely different. As all estimators depend deterministically on the state probabilities, one can compare two MSVAR models by comparing their state probabilities. Correlations between the state probability vectors are calculated for the model of the best run compared to the models of the other runs [3]. The correlation matrices are in appendix A. There one can see that the models are almost always about a state permutation different, because usually one value per row and column is close to one. This means that the runs converge to similar maxima, up to a permutation of the states.

In appendix C the EM estimation of an extended model is shown. There the conclusion can be drawn that due to the increases number of parameters, the algorithm struggles to find the global maximum. A suited solution would be to increase the number of randomly initiated runs. For the model estimated only on heart frequency and blood pressure, this does not seem to be the case. The number of runs, 5, seems to lead to good convergence.

4.3.2. CONVERGENCE OF PARAMETERS

Next to convergence in likelihood, it is interesting how the parameters change while the likelihood is optimised. An MSVAR has many parameters. As a representable subset the

[3] Despite, the other results being on a model with 20 states, these results are obtained from a model with 5 states, because the results are only visualisable for a small number of states

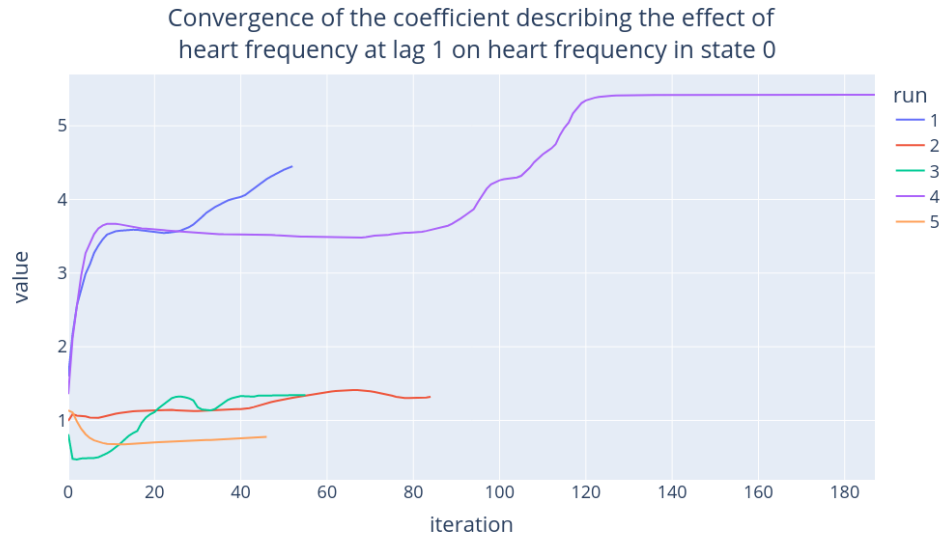


Figure 4.7: Trace of the coefficient describing the effect heart frequency at lag one on heart frequency in state 0 for different runs of the EM algorithm.

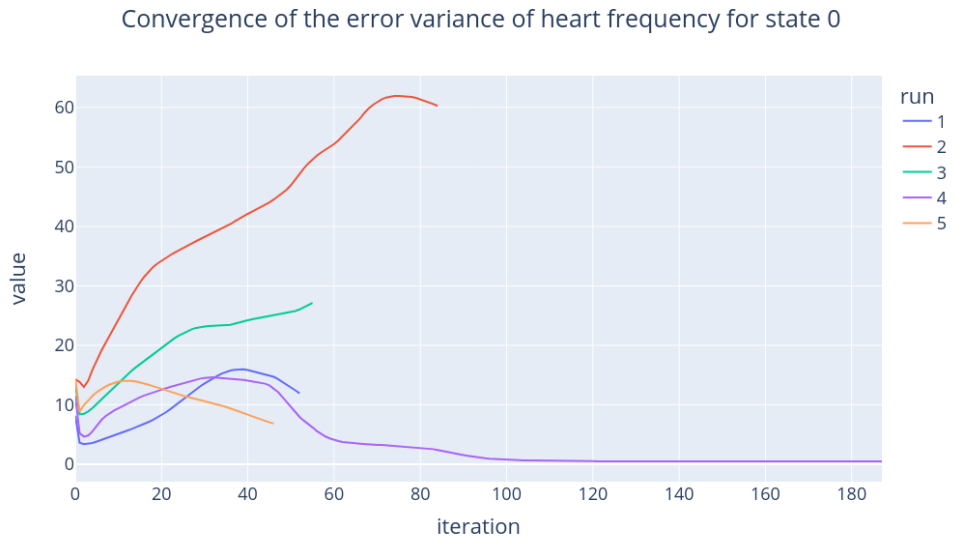


Figure 4.8: Trace of the variance of the error term for heart frequency in state 0 for different runs of the EM algorithm.

following parameters will be analysed:

- the probability of transitioning from state 0 into state 0,
- the var coefficient, that is the effect of the first lag of heart frequency onto heart frequency of state 0,
- the variance of the error term of the heart frequency of state 0.

The convergence of the parameters is visualised in figures 4.6, 4.7 and 4.8. A few observations can be made. First of all, in many of the runs, especially the shorter runs, the parameters did not seem converged. This indicates that there is a large set of model parameters with very similar likelihoods. The parameters changed a lot in the last 10 iteration steps, however the likelihood did not increase. Furthermore, the parameter coefficients are correlated a lot. There are obvious correlations in MSVAR models, such as between the different VAR coefficients in the VAR of a specific state. However, the graphs show correlations in seemingly unrelated parameters. Finally, the fourth run continued for a long period with only very small likelihood improvements. These parameters hardly changed in the last 100 iterations. It might have been good enough to half the upper bound on the number of iterations in hindsight, however now one can be sure that the algorithm converged. Better stopping criteria might have been more computationally efficient.

4.4. MODEL SELECTION

In this section, the most appropriate model will be chosen. It is decided that an appropriate model is a model that describes the data well, i.e. has a high log-likelihood, with a minimal number of parameters. This is equivalent to finding a model with minimal BIC or AIC (see section 2.6). Three comparisons will be made. First, the effect of data transformations, log transformation and differencing (section 2.1), will be discussed. Second, the (shrinkage) MSVAR is compared with other time series models. Third, the best hyperparameters are found.

In Figure 4.9 the likelihood and the number of free model parameters is plotted. One would like to end up in the left upper corner, where the model makes the data likely with a minimal set of free parameters. In general, the log likelihood will increase with the number of free parameters as models are fitted using MLE methods. Similar graphs on transformed data can be found in Figure B.1. It appears that there is no real difference between any of the models fitted on the untransformed data or the transformed data in terms of likelihood. The model with the highest likelihood is fitted to untransformed data. Using a model on untransformed data makes it easier to interpret. Therefore, the data will not be transformed from here on.

Looking at Figure 4.9 one can see that the Panel VAR models have a lot of free parameters, without actually having a good likelihood. Therefore, these models are not suitable for the data. The MSVAR models describe the data best. Especially the MSVARs with a lot of states and lags.

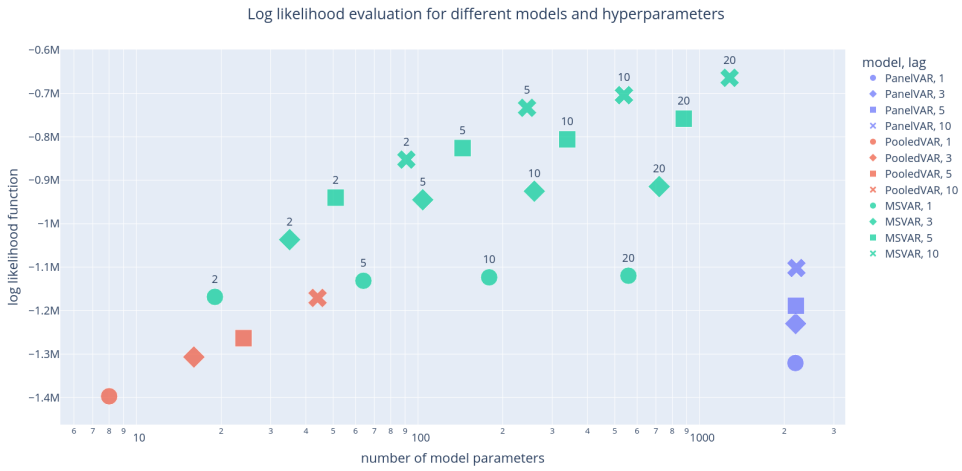


Figure 4.9: Scatter plot of all considered models with the number of free model parameters on the x-axis and the log likelihood on the y-axis. The models are Pooled VAR (red), Panel VAR (blue), (shrunken) MSVAR (green), which also differ in lag (symbol) and for the MSVAR in the number of states (text in plot).

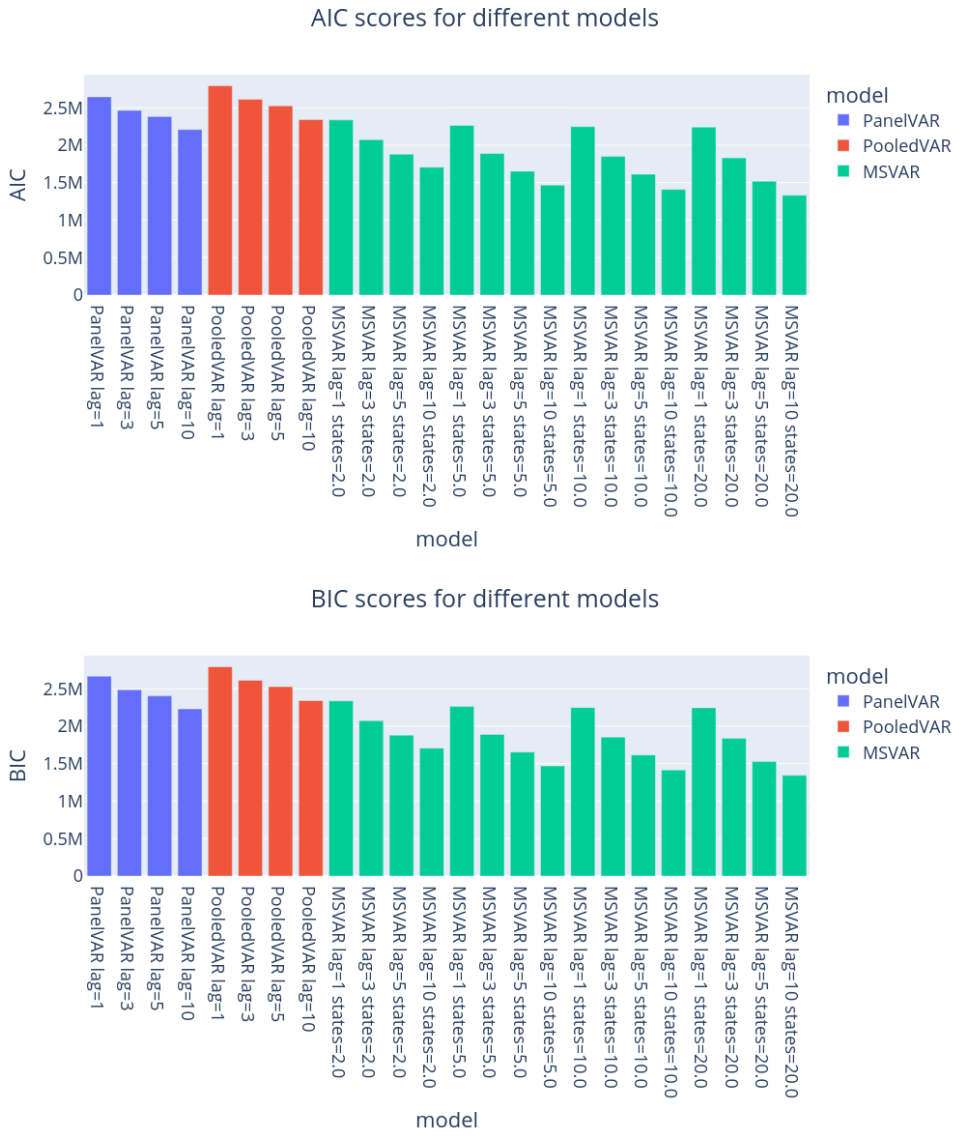


Figure 4.10: The AIC(top) and BIC(bottom) scores for three different models: PanelVAR, PooledVAR and the MSVAR. All models are estimated for several different hyperparameter values.

5

GIBBS SAMPLER ESTIMATION

In the previous chapter a MSVAR was fitted with the EM algorithm. The final hyperparameters were chosen based on penalized likelihood scores. The optimal lag was 10 and the optimal number of states was 20. In this chapter the exact same model will be fitted in Bayesian fashion by using a Gibbs sampler.

5.1. HYPERPARAMETER CHOICES

In section 2.5 it was discussed how one could fit a MSVAR in Bayesian fashion. In the specifications of the priors, nine hyperparameters were left open. In this section, decision for these hyperparameters will be made. However, first, the label switching problem will be discussed.

5.1.1. LABEL SWITCHING PROBLEM

In Bayesian estimation of a MSVAR with a Gibbs sampler, there is the so-called label switching problem [47, 66, 67]. If the prior distribution for the vector autoregression in all states is equal, any state permutation of each possible model is equally likely. It will therefore probably happen that the model samples in the MCMC-chain consist of multiple (state) permutations of the same model. A state permutation of an MSVAR model is where all parameters are switched around by relabelling the states. The model is the same, except for the labels given to the states. An option is to permutate the states every sample by by ordering based on some property every state has (identifiability constraints). The most straightforward way to circumvent this problem is to choose different priors for each state. States will specialize in some direction and the MCMC chain is more likely to converge. In conclusion, some of the prior hyperparameters will be chosen to differ over the states to circumvent the the label switching problem.

5.1.2. PRIORS FOR MARKOV CHAIN

In section 2.5.5 it was mentioned that there are three hyperparameters in the prior distribution for the Markov chain, the expected consecutive time in a state, the confidence

in the initial probability, and the confidence in the transition matrix prior. The expected consecutive time, will be, similarly as in the EM algorithm shrinkage case (see 4.2), set to 90 minutes. The confidence in the transition matrix prior should be set quite high to have the desired behaviour. To approximate the 0.8 shrinkage coefficient, one should do the following. The expected number of observations for each separate Dirichlet distribution is the total number of observations divided by the number of states. To have the shrinkage part of the estimator contribute 80%, the confidence in the prior should be four times as big as the observations. There is no reason to have the confidence for the initial distribution prior equally high. It is decided to have it about half of the number of observations, i.e., the number of surgeries, such that the data outweighs the prior.

The Dirichlet distribution is a conjugate prior for the rows of the transition matrix and the initial distribution as it is a conjugate prior for a multinomial distribution. A Dirichlet distribution, $\mathscr{D}(e_1,\ldots,e_n)$, has the multinomial distribution with the normalized weights as probabilities as expectation. Where the normalization divides the weights by their sum. The sum of the prior weights is a measure for the confidence in the prior.

5

5.1.3. VAR PRIORS

For the VAR priors, 6 hyperparameters should be chosen: ϕ_0, ϕ_1, ϕ_2, ϕ_3, ν and Σ_0. Some of these hyperparameters should differ over the states to avoid the label switching problem. The following decisions are made.

ϕ_0 The strictness of the prior is decided to differ over the states. The smaller ϕ_0 the more the resulting VAR will be random walk-like. The five chosen values are: {1, 0.7, 0.5, 0.3, 0.1}.

ϕ_1 The relative strictness of endogenous coefficients is decided to be set to 0.5. A priori, one would expect to have a lag of the same variable to have a bigger effect than another variable.

ϕ_2 The relative strictness of exogenous coefficients is decided to be 0.5, for a similar reason as for ϕ_1.

ϕ_3 The exponent of the lag function determines whether the coefficients at large lags can still have a large influence. This thus also influences how much the VAR is like a random walk. For large values of ϕ_3, the coefficients at large lags will be close to zero and the resulting VAR will be like a random walk. It is decided to differ at the same time as ϕ_0 over 5 different values being $\{3,2,1,0.5,0.1\}$.

ν The confidence of the covariance prior is decided to be the total number of observations divided by two times the number of states. In expectation, the data has twice as much influence as the prior.

Σ_0 The basic VAR covariance matrix is decided to differ over the states. As a basis it is chosen to take the covariance of the PooledVAR with the same number of lags. This covariance matrix is equal to

$$\Sigma_{\text{PooledVAR}} = \begin{bmatrix} 14.13 & 2.03 \\ 2.03 & 18.80 \end{bmatrix} \tag{5.1}$$

In conclusion, $\Sigma_0 = \lambda_i \Sigma_{\text{PooledVAR}}$ with $\lambda_i \in \{10, 1, 10^{-1}, 10^{-2}\}$. There are thus states that are quite predictable (low λ) and states that are not predictable (high λ).

ν The confidence in the covariance prior is chosen to equal the total number of observations divided by six, to have a relatively loose prior.

The different priors for the twenty states are given by the Cartesian product of the 5 types of random walk-like states and four types of predictability.

5.2. RESULTING MODEL

Using the Gibbs simpler (see 2.5.2) a Markov Chain Monte Carlo sample was created for the MSVAR model with 20 states, 10 lags, and prior hyperparameters as in the above section. It was chosen to get 5000 samples.

5.2.1. CONVERGENCE

First, the trace of the MCMC is analysed. Next to that, a burn-in size should be chosen. Finally, it is interesting to see whether the chain converges in some sense.

In figures 5.1, 5.2, 5.3 MCMC trace plots are shown for many parameters in the model. These plots do not contain all parameters, but it is checked that these plots are representative for the general behaviour. Many things stand out. First, of all one can see that the most change is happening at the start. From a random start, the chain is converging towards a more probable configuration. The last major changes are happening just before the thousandth step. Hence, the burn-in period is set to 1000 iterations. One can see that the changes are correlated, which is to be expected. After the thousandth step the algorithm seems converged towards a state in which it is varying around an optimal value as to be expected in a MCMC-method.

5.2.2. POSTERIOR DISTRIBUTIONS

Given the burn-in period, the posterior distribution can be determined. In Figure 4.2 the histogram for the diagonal values of the transition matrix are shown. These are the most important probabilities of the transition matrix as these determine the probability of staying in the same state. The value of the prior mean is added. Similarly as with the EM algorithm, the maximum likelihood estimates of the probabilities are much lower. There is no state for which this probability is higher than the prior mean. Due to shrinkage or using a prior with a mean close to one, in this case, the values are closer to one than with the maximum likelihood estimate. This induces more stability in the Markov chain, which is desired. A model with a not so stable Markov chain is also more prone to overfitting. State 3 stands out. The transition probability is very low given the prior mean. This means that the MLE is very low and that patients do not stay in this state for a long time. The uncertainty in the posterior estimate seems similar for each state.

In Figure 5.5 a similar graph is shown for the VAR coefficient describing the effect of heart rate lag 1 on heart rate for each of the states. For this coefficient, the prior mean was 1, like in a random walk. Both the location and the scale of the posterior distributions differ a lot between states. The strictness of the prior increased with the state number.

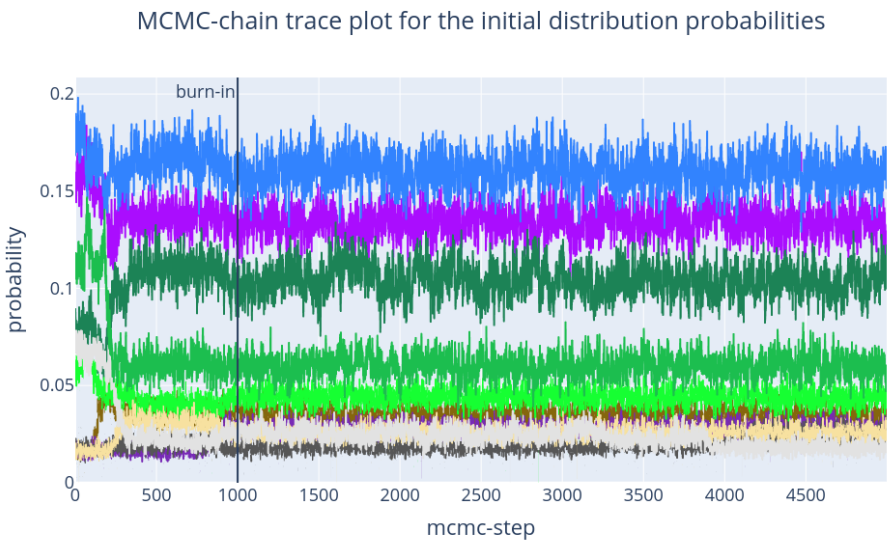


Figure 5.1: MCMC trace plot for the probabilities of the initial distribution for the first 10 states. The burn-in is decided to bet set at 1000. The colours are given by the usual state colouring as can be found in Figure E.1.

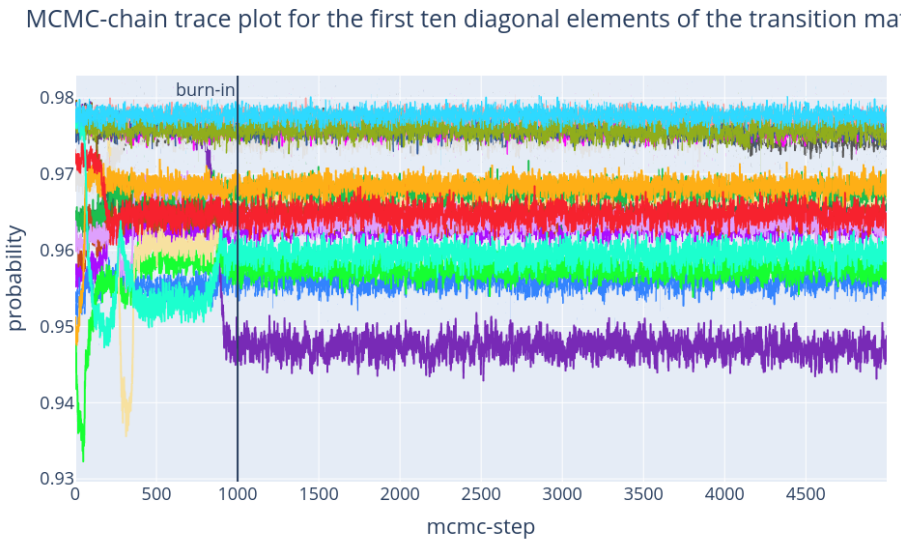


Figure 5.2: MCMC trace plot for the first 10 diagonal elements of the transition matrix. The burn-in is decided to bet set at 1000. The colours are given by the usual state colouring as can be found in Figure E.1.

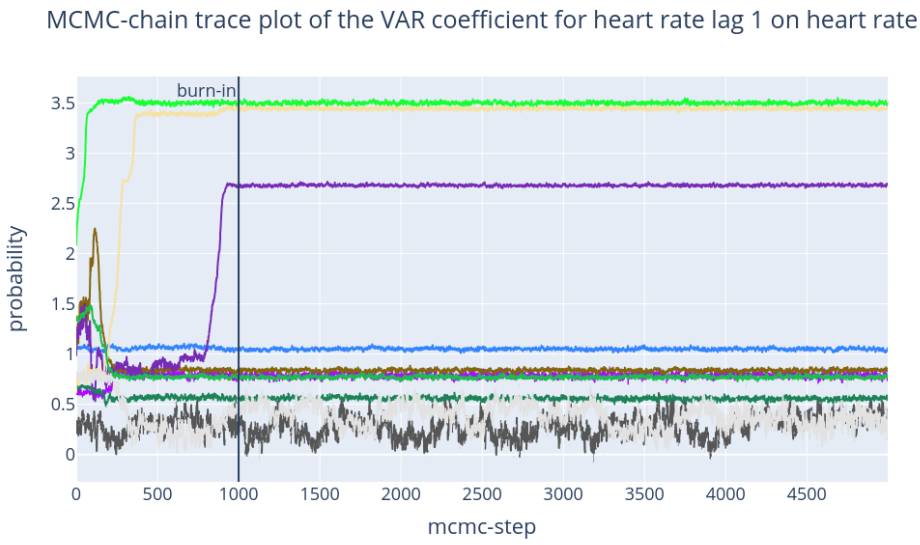


Figure 5.3: MCMC trace plot of the coefficient for the effect of heart at lag 1 on heart rate for all the states. The burn-in is decided to bet set at 1000. The colours are given by the usual state colouring as can be found in Figure F.1.

However, there does not really seem to be a relation between the state number and the posterior mean. However, it is the case, that most posteriors are located close to one.

Finally, in Figure 5.6 the sample histogram is plotted for the prediction variance of the heart rate. There are a lot of variances close to zero, but there are also large variances. The posterior uncertainty seems to be related to the location, which is not surprising. The states differ a lot in how predictable the heart rate is apparently. The prior mean for the var covariance is highest for states divisible by four. This seemed to have a big effect. The group with approximately a variance of 150 consists of states 4, 8, 12, and 16. The only state with a high prior mean as well, state 0, is located at about 65. The prior seems to have a large influence. Of course, the prior has influence, however there seems to be going on more here. Unpredictable data points seem to have been gathered in certain states, increasing the effect of the higher prior variance.

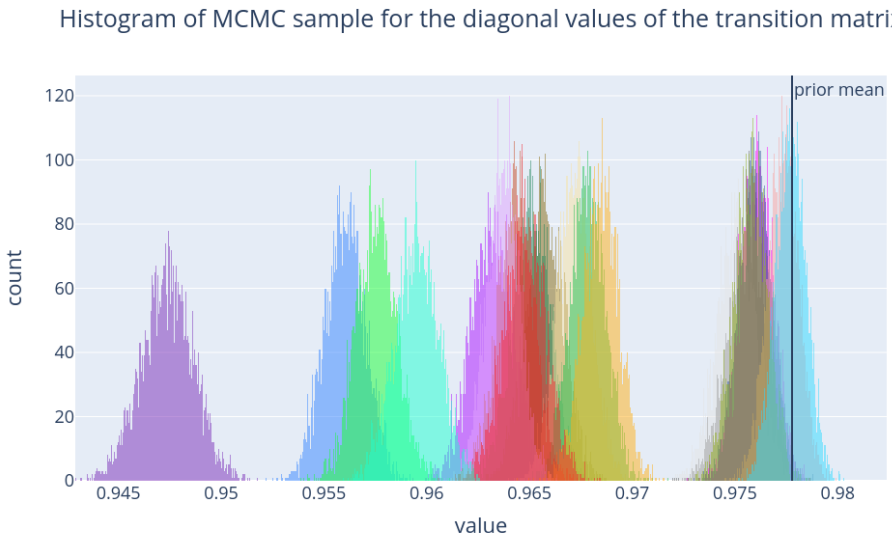


Figure 5.4: MCMC sample histogram of the diagonal values of the transition matrix of the MSVAR. The sample consists of 4000 samples obtained after a burn-in period of 1000. The colours are given by the usual state colouring as can be found in Figure F.1.

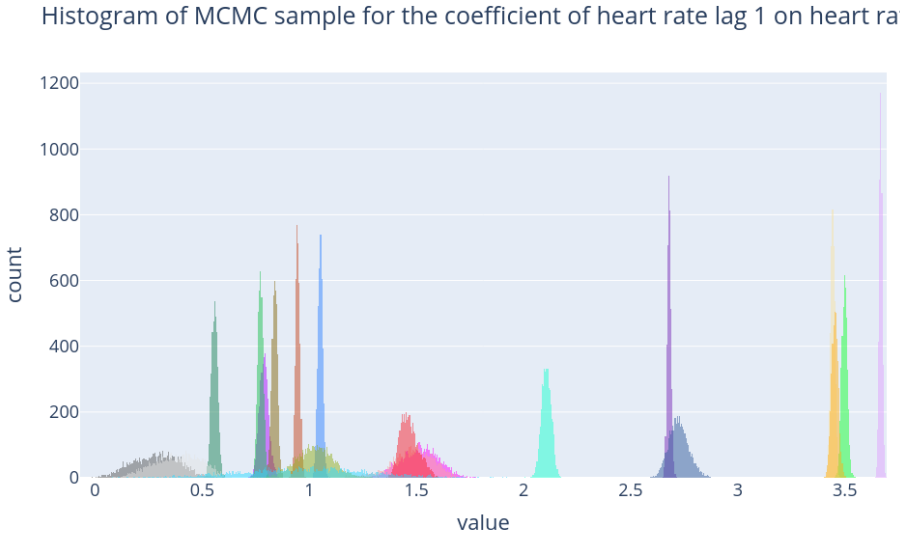


Figure 5.5: MCMC sample histogram of the MSVAR coefficient describing the effect van heart rate lag 1 at heart rate for each state. The sample consists of 4000 samples obtained after a burn-in period of 1000. The colours are given by the usual state colouring as can be found in Figure F.1.

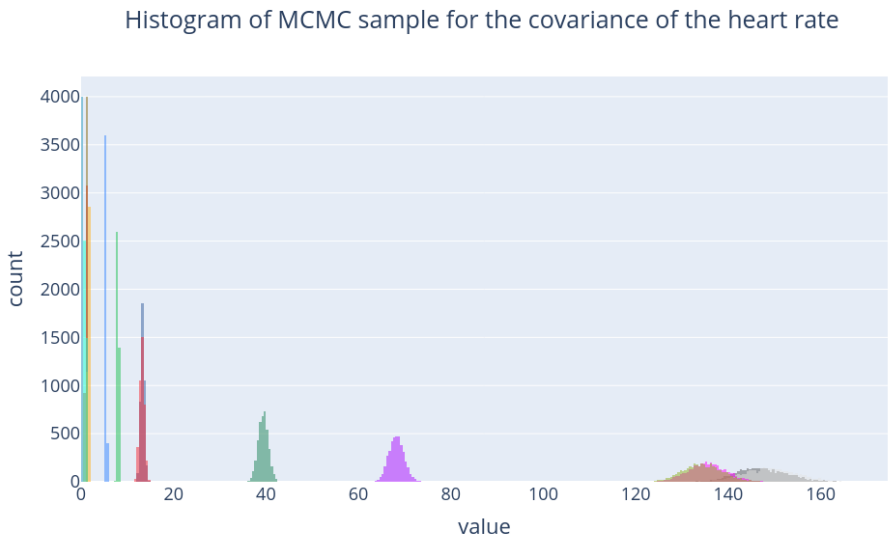


Figure 5.6: MCMC sample histogram of the covariance of the heart rate as modelled with a MSVAR. The sample consists of 4000 samples obtained after a burn-in period of 1000. The colours are given by the usual state colouring as can be found in Figure F.1.

5.3. HIERARCHICAL MODEL

In section 5.1 choices were made on the hyperparameters. In many cases, these choices were quite arbitrary. To research the influence of uncertainty in these hyperparameters, a hierarchical model will be constructed. Most of the hyperparameters will be suited with a prior distribution. This prior distribution will be independent of the data and other parameters. First, these distributions will be chosen. Next, the resulting model will be presented, including a comparison with the non-hierarchical model.

5.3.1. HYPERPARAMETER PRIORS

To create a hierarchical model, the hyperparameters should be equipped with probability distributions. There are already multiple values in use for the hyperparameters that differ over the states. Therefore, it is decided to not equip these with a distribution. The others will be discussed one by one. In general, the prior mean is chosen equal to the fixed value in the non-hierarchical model, as that seems the best value a priori. The other degrees of freedom in the distribution are chosen to represent the a priori uncertainty in the hyperparameter.

ϕ_1 The relative strictness of endogenous variables of other variables should be between zero and one. The fixed value was 0.5. An appropriate distribution is therefore a Beta(2,2) distribution. It has the same mean as the fixed value and a high variance.

ϕ_2 The relative strictness for exogenous variables is actually not interesting, because there are no exogenous variables in the model.

ν The confidence in the covariance prior will be reparametrized. Because the inverse-Wishart prior is conjugate and can be viewed as a weighted combination of the prior and the MLE, ν can be reparametrized as the relative part the prior contributes to the new estimate in expectation. For this new ν, $\tilde{\nu}$, a Beta(2,10) is appropriate. The variance is medium and the mean corresponds with the fixed value version.

confidence initial distribution For the confidence in the initial distribution, a similar approach is used. The value is reparametrized to the fraction the prior contributes to the posterior (mean). Again, using a semi-strict prior with a mean corresponding to the fixed value, a Beta(2,12) distribution is appropriate.

confidence transition matrix For the confidence in the transition matrix the same approach is used again. The value is reparametrized to the fraction the prior contributes. In the EM algorithm, it was found that the shrinkage of the transition matrix is important to obtain good results, this means more is known about appropriate values, and a stricter prior will be used, namely, the Beta(40,10) distribution.

expected consecutive time To have a uniform approach, the parameter is rewritten as the probability of staying in the same state, i.e. the diagonal elements of the transition matrix. This should again be a very strict prior to obtain a model with a stable Markov chain. The prior distribution of this probability is decided to be the Beta(88,2) distribution.

Given these prior distributions, a step should be added to the Gibbs sampler (Algorithm 4) to obtain a valid posterior sample. In each iteration, the hyper parameters should be independently drawn. These new hyperparameter values should be used in the other priors in the current iteration step. In the next section, the effect of this added layer will be discussed.

5.3.2. POSTERIOR DISTRIBUTIONS

In the previous section the model was expanded with an extra layer of priors for the hyper parameters. The most interesting part is whether this extra prior uncertainty also results in posterior uncertainty. A glance at the trace plots shows that there is more variability. Most of the big changes are gone after the thousandth sample, however some are still left. For comparison reasons still a burn-in period of 1000 iterations will be used.

Figure 5.7 is similar to Figure 5.4 except for the hierarchical model used. In this case, it is very clear that the extra layer of uncertainty resulted in a lot of posterior uncertainty. This can be explained by the strong prior on the transition matrix in combination with a looser prior on the desired transition matrix, this leads to a posterior with a high variance. Similarly, Figure 5.8 is the update of Figure 5.5. In this case, the plots look very similar, with a similar distribution of the peaks and the scales vary in a similar way. It must be noted that specific peaks changed a lot, indicating that the states are permutated in some kind. Finally, Figure 5.9 is the hierarchical model version of 5.6. The same things can be set as about the VAR coefficient. In general, it looks the same with the same distributions of location and scale. However, looking at a specific state shows big

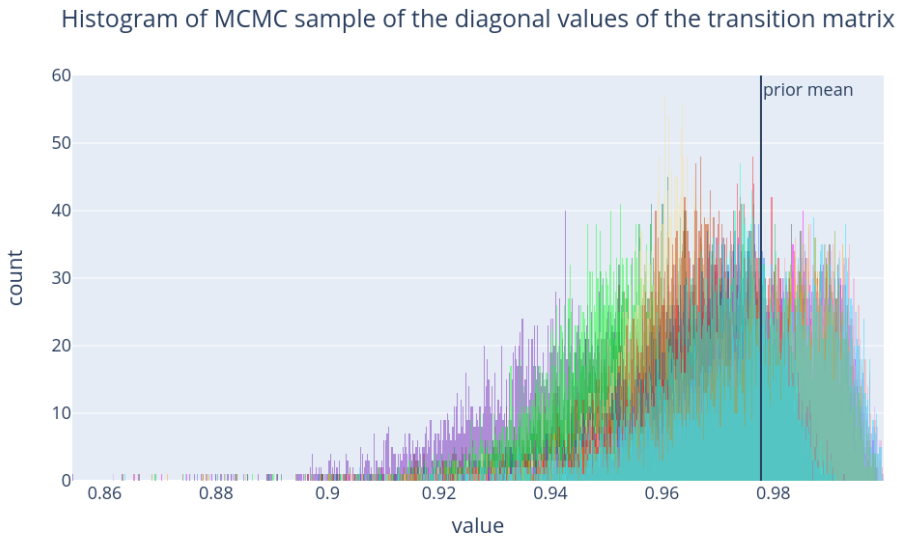


Figure 5.7: MCMC sample histogram of the diagonal values of the transition matrix of the MSVAR. The models includes a hierarchical model for the priors. The sample consists of 4000 samples obtained after a burn-in period of 1000. The colours are given by the usual state colouring as can be found in Figure F.1.

changes. This contributes to the hypothesis that the states are permutated in some way. In conclusion, the extra layer of uncertainty seems to mainly have affected the Markov chain for the states. This is not unreasonable, as this is where the strongest priors are applied. The VARs per state have a relatively loose prior and changing these hyperparameters has little effect.

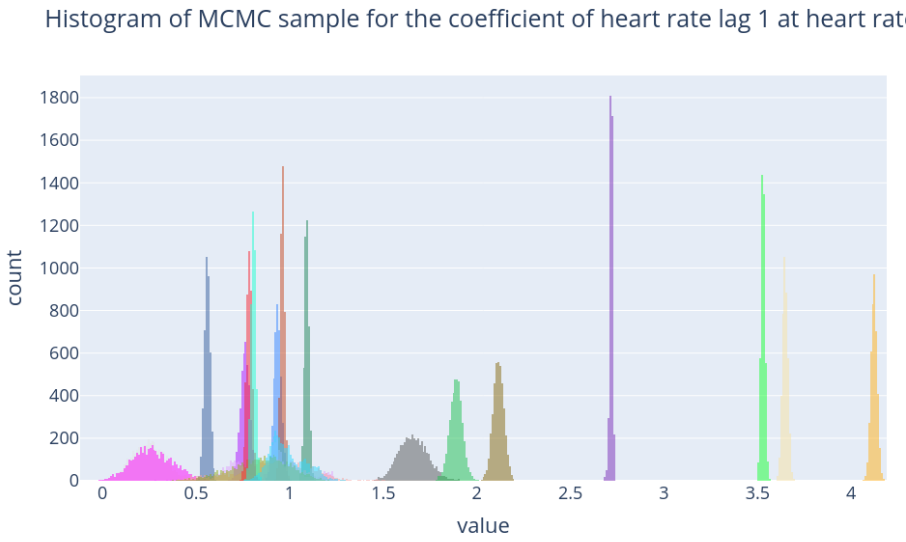


Figure 5.8: MCMC sample histogram of the MSVAR coefficient describing the effect van heart rate lag 1 at heart rate for each state. The models includes a hierarchical model for the priors. The sample consists of 4000 samples obtained after a burn-in period of 1000. The colours are given by the usual state colouring as can be found in Figure E1.

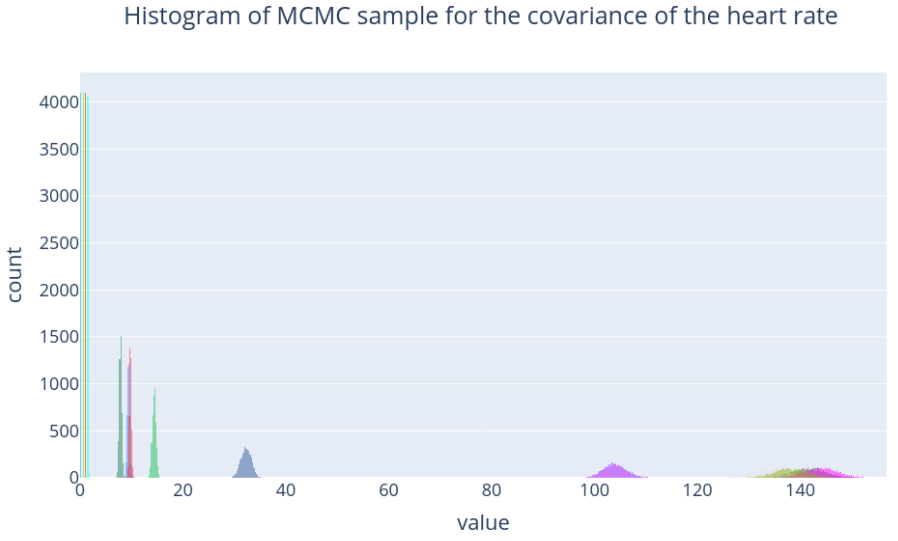


Figure 5.9: MCMC sample histogram of the covariance of the heart rate as modelled with a MSVAR. The models includes a hierarchical model for the priors. The sample consists of 4000 samples obtained after a burn-in period of 1000. The colours are given by the usual state colouring as can be found in Figure E1.

6

INTERPRETING STATES

The MSVAR model was chosen with the idea in mind to see whether the states are medically interpretable. Especially of interest is whether it is possible to predict postoperative troponin T from the relative time in each of the states. In this chapter, it will be investigated how the states can be interpreted and how they can be related to other variables related to the surgery. The idea is to see whether there is any practical meaning to the (so-far) abstract states. The results are based on the bayesian hierarchical model as that is the most extensive model, describing the uncertainties best.

6.1. GENERAL DISTRIBUTION OF STATES

In this section, light will be shed on the general distribution of the states. First, whether the states are equally occurring. Next, whether the states occur in different parts of the surgery.

In Figure 6.1 the total number of periods is counted where each state is the mode (most probable) state. There are a lot of differences among the states. State 7 is by far the most occurring. There are also many states patients are hardly in, being: 4, 8, 11, 12, 14, 15, 16, 17, and 20. This is remarkable as the 20-state model clearly outperformed the 10-state model on both the AIC and BIC criteria (Figure 4.9). There does not seem to be any pattern of states on a first glance. Next, each surgery is divided into 30 time periods. For each period, it is recorded how much the patient is in each state and this is combined for each surgery. The total per period is normalized to sum to one. The relative time in each state per surgery period is visualized in Figure 6.2. Clear patterns are visible. Some states (0, 5, 13) are more prominent at the start, whereas others (7, 10, 17, 18) are more prominent during the middle phase of the surgery. There is a clear final phase of the surgery, where (6, 14) are occurring relatively more. It is remarkable how well these phases are visible given that the surgeries have very different durations. Furthermore, it can be noticed that the start is much longer than the end. This seems to be because the first 20 minutes of each surgery are not taken into account. To have lagged data for the autoregressions, no inference is possible for the first 20 minutes of each surgery. Therefore, the sharpest change of the surgery start has already been. The last section seems to

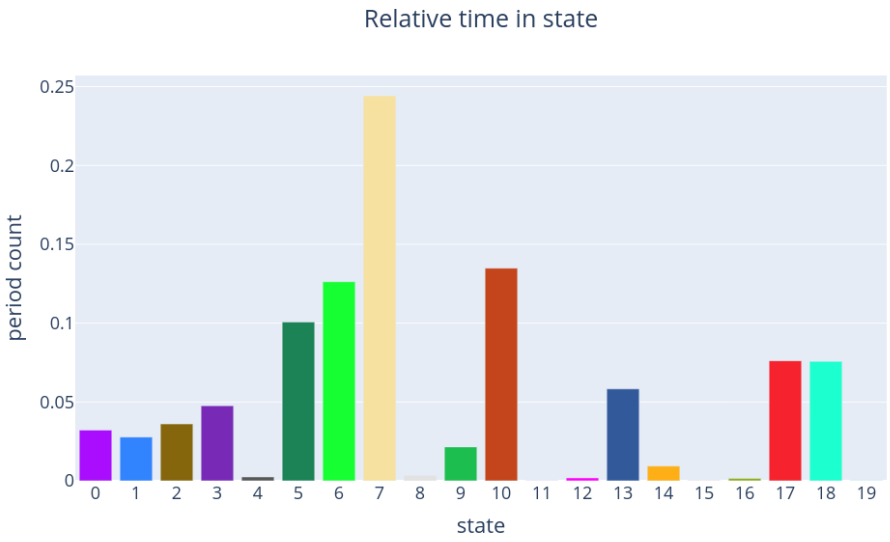


Figure 6.1: The amount of time points each of the state is most likely to be the state. The colours are given by the usual state colouring as can be found in Figure E.1.

coincide with the awakening of the patient leading to big changes in the hemodynamics. In conclusion, the amount states are occurring differs a lot. Furthermore, the state occurrence is clearly related to the different phases of the surgery.

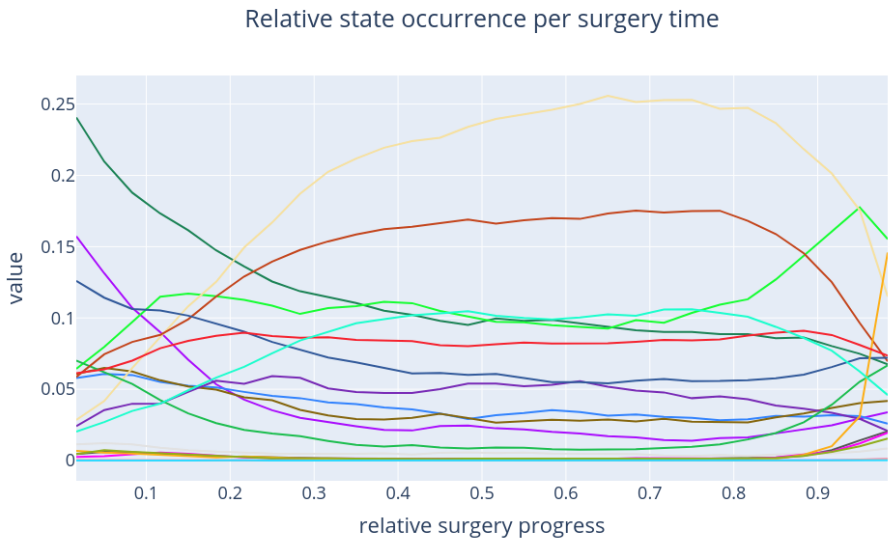


Figure 6.2: For each of 30 quantiles the (relative) occurrence of the different states is determined. At the start, in the middle and at the end different states are dominant. The colours are given by the usual state colouring as can be found in Figure F.1.

6.2. POSTOPERATIVE PREDICTIONS

The main goal of the research is to find intraoperative variables with prognostic value for postoperative troponin T. As explained in chapter 3 the idea in this research is to use the relative time in states per surgery as predictors for high troponin T. This idea is based on the approach used in [25, 26, 27] to use MSVAR relative time in states as predictors for mortality. In this section, the prognostic value of the relative time in states for postoperative variables is researched.

The following binary variables were taken into account:

- High postoperative troponin T, that means that at least one of the high-sensitivity troponin T in the three days following surgery was above 50 ng L^{-1}. (N=1516)
- 30-day mortality, which is whether the patient deceased in the 30 days following surgery. (N=1558)
- 1-year mortality is whether the patient deceased within one year following surgery. (N=1376)
- Whether there is a positive change in troponin T measurement over the course of the surgery. (N=280)

Three different feature sets were used, purely the relative time in states, just preoperative features (see the list in appendix E) and the combination of both. Results are obtained using 5-fold cross validation. That is, the sample is splitted in 5 folds. 5 times a model

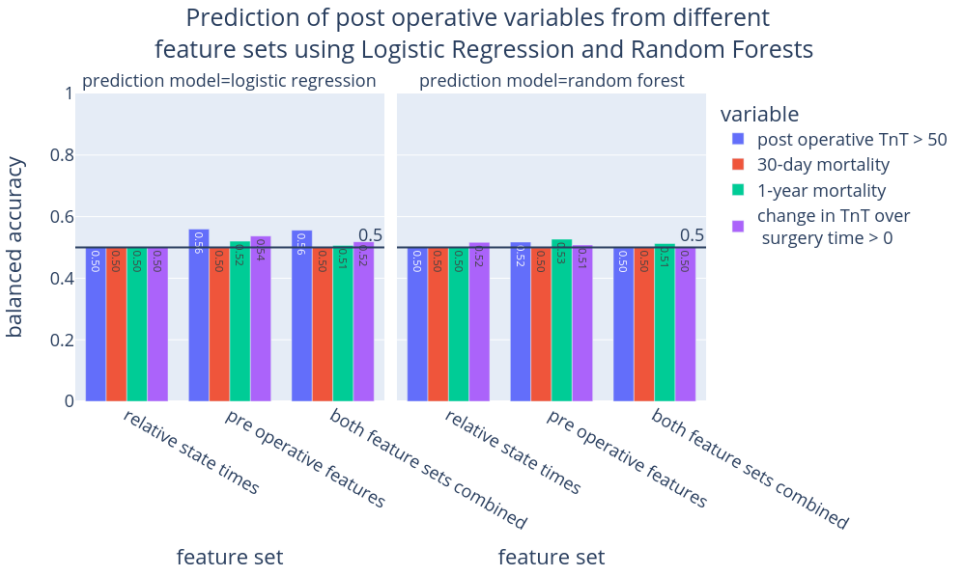


Figure 6.3: Balanced accuracy scores for several post operative predictions made by using a logistic regressions (left) and random forest (right) on three different feature sets. Each time the score is based on 5-fold cross validation. The 0.5 line corresponds to the score random guessing would receive.

is estimated on 4 of the 5 folds and the results are obtained on the remaining fold. This gives reliable results on unseen data. The score measure used is the balanced accuracy, as the data is quite unbalanced, especially for the mortality variables. Two different models are used as discussed in 3.3. The results of the experiment can be found in Figure 6.3.

Many conclusions can be drawn from Figure 6.3. First of all, there are no real differences between using either logistic regression or random forests. Secondly, the relative time in states alone have no prognostic value for any of the postoperative variables. The line at a balanced accuracy score of 0.5 indicates the line of random guessing, and the performance of the models on the relative state times achieves that score. Thirdly, the preoperative features have a little bit of prognostic value, especially for postoperative hsTnT. Finally, the prognostic value of the combination is approximately the same as for the preoperative values. There is thus no prognostic and no added prognostic value in the relative state times for any of the postoperative variables. As this includes troponin T, one can conclude that the MSVAR approach does not yield variables with prognostic value for postoperative troponin T.

6.3. INTRAOPERATIVE PROPERTIES PER STATE

The next way the states will be interpreted is by calculating the mean values for each state. The mean values considered are on vasopressor usage and the actual hemodynamic values.

In Figure 6.4 the mean heart rate and mean blood pressure are plotted for each of the

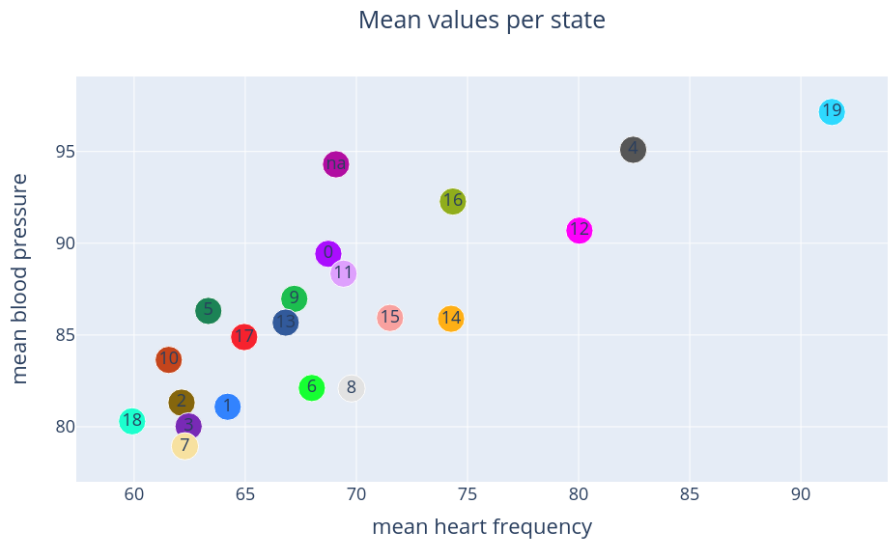


Figure 6.4: Scatter plot of the mean blood pressure and hart rate for each of the states. The colours are given by the usual state colouring as can be found in Figure F.1.

states. There is a clear trend, where the mean heart rate and mean blood pressure is high, as is the case in general. There are no states that differ from this trend. The states seem specialized and are divided over the spectrum. It is unclear whether this is significant. The Kruskal-Wallis test has a test statistic of 8464.637466002121 and a p-value numerically equal to zero. However, the test assumes independent samples, which is definitely not the case. Often the hemodynamics are quite stable and similar values follow each other. Together with the fact that the transition matrix has large diagonal values, there is a lot of dependence, which makes the test much more erroneous significant. There is no clear alternative to test the significance of this. However, because we are dealing with clearly visible differences over more than 200.000 data points, it looks significant.

For the vasopressors, only three vasopressors are considered as these are supplied most often. The results are in Figure 6.5. The values for each of the three vasopressors are clearly related. The amount of dobutamine differs a lot more than fenylefrine, which seems to be provided almost always. Again, the states seem specialized in states where more and less vasopression is supplied.

Finally, a relation between vasopressor usage and blood pressure is expected. Vasopressors are used to increase the blood pressure when it is too low. Therefore, one would expect vasopressors to be used in states that also have a low average blood pressure (see Figure 6.6). The relations found are too be expected and exist in general. It is interesting to see though that the states are clearly specialized over this spectrum.

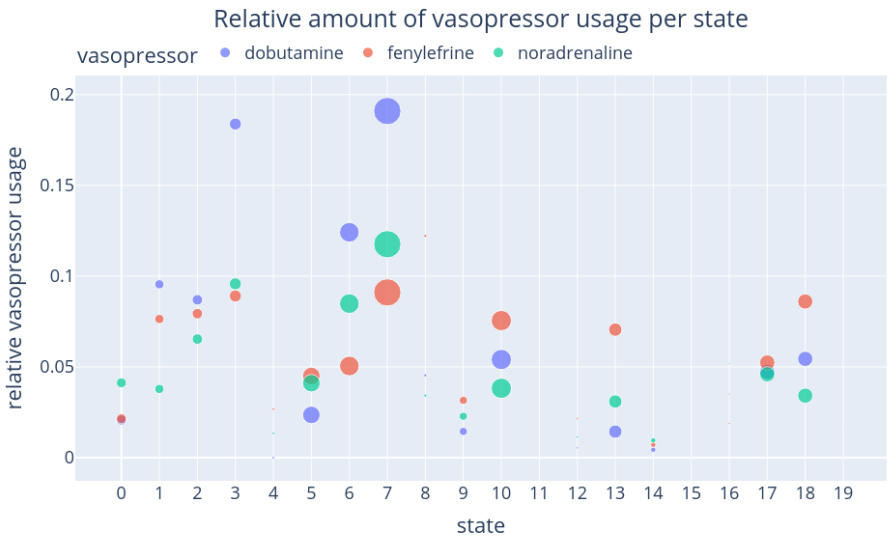


Figure 6.5: Overview of the relative amount of vasopressor usage per state for the three most common vasopressors. The colours are given by the usual state colouring as can be found in Figure F.1.

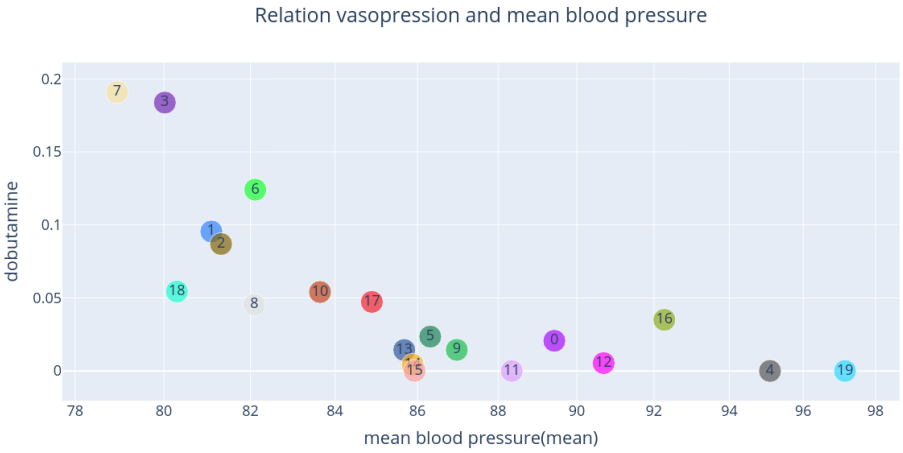


Figure 6.6: Relation between relative dobutamine usage over the states and the mean blood pressure mean per state. The graphs for the other vasopressors are similar as these are all related. Each point represents a state, where the number is the state number and the colours are the same for the states as in the other graphs. Vasopressor (dobutamine) usage is mainly present in states with low mean blood pressure. The colours are given by the usual state colouring as can be found in Figure F.1.

6.4. VAR ANALYSIS

The final analysis of the states is by analysing the VAR corresponding to the state itself. This will be done in two ways. First, the covariances of the VARs are analysed. Next, the VARs are simulated and properties are extracted from the simulations

The diagonal elements of the covariance matrix (variances) of a VAR show the predictability of the model. High covariances mean that there is little relationship and a lot of uncertainty in the data. The prior covariances were different for each state and exactly that pattern is visible in Figure 6.7. The posterior mean covariance is clearly related to the prior. There was a lot of uncertainty in what was a good covariance prior. Because there was a lot of uncertainty in the prior covariance it was varied over the states at four levels of magnitude. Because it was varied over the states, it was not taken as an hyperparameter in the hierarchical model. If the pattern in the covariance is compared with the occurrence of each state in Figure 6.1 there is a pattern that the states with the highest order of magnitude in the prior, and hence also in posterior covariance are the least occurring. The prior covariance is thus one of the reasons why states are not occurring. However, it also shows that difference in VAR priors are effective in preventing label switching as the posterior are actually quite different.

From the simulations, as shown in Figure 6.8, several conclusions can be drawn. First of all, the VARs differ widely in their stability. States 11, 24, 25, and 19 are clearly not stable. In addition, states 12 and 16 also have a clear trend. All these states are not occurring so much. The frequently occurring states are all stable. Next to the stability, one can also look at the smoothness. One can see the differences best when one compares states 4 and 7. Both are stable with a similar general pattern, however, the VAR simulation for state 7 is much smoother. Each column in Figure 6.8 has the same covariance prior and unsurprisingly the smoothness is related to the covariance and thus covariance prior as well. This is because a large covariance matrix is the main source of jaggedness. In Figure 6.9 it can be seen that vasopressor usage and posterior covariance are related. Again, this is a relation that is to be expected as vasopression suppresses the natural dynamics of the hemodynamics. However, it is interesting to see that the states specialized over this relation.

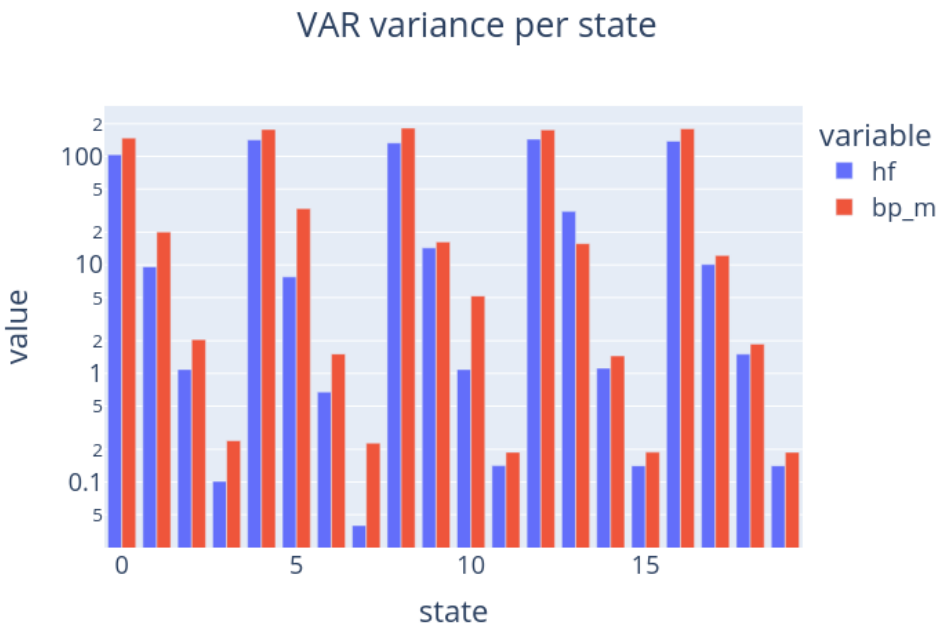


Figure 6.7: The variance (diagonal elements of covariance matrix) for each variable and for each state. The point estimate of the covariance element is the posterior mean.

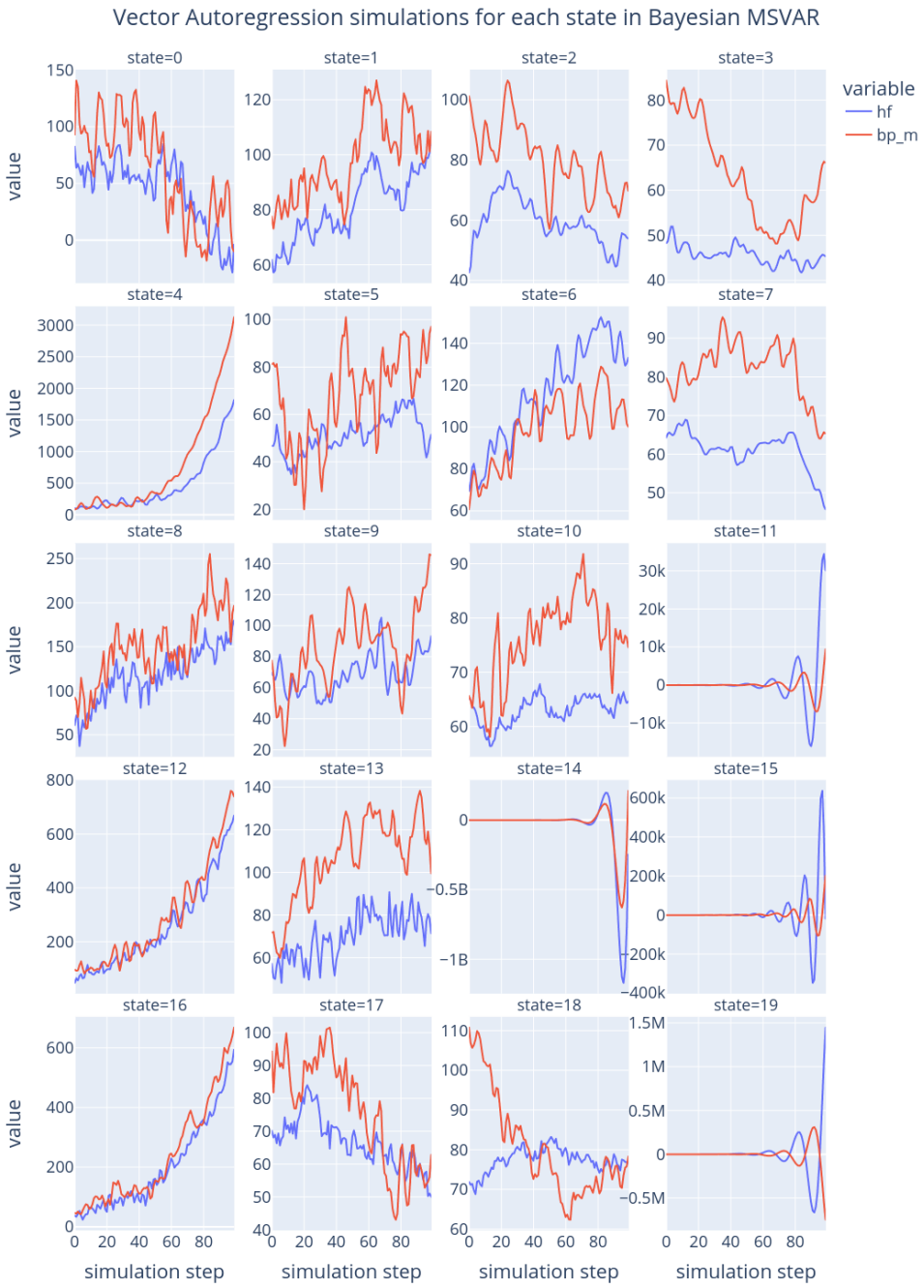


Figure 6.8: VAR simulations for each of the states in the MSVAR. For each of the coefficients the posterior mean is taken and the resulting VARs are simulated for 100 steps. Note that y-axis can differ a lot.

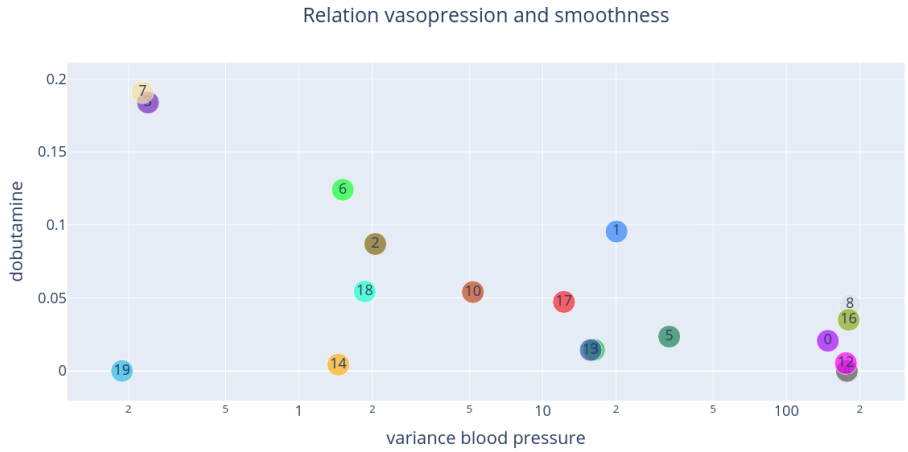


Figure 6.9: The relation for a specific vasopressor, namely dobutamine and the smoothness, or predictability of the VAR. The predictability or smoothness is inverse to the variance of the VAR, which is on the x-axis here. The more to the left, the more predictable or smooth. The colours are given by the usual state colouring as can be found in Figure F.1.

6.5. CONCLUSION

In this section, we try to combine the insights from the previous few sections into general conclusions on the states. First, the most occurring state, state 7 is discussed. This state mainly occurs in the middle part of the surgery. The mean heart rate and especially the mean blood pressure are low. This explains why a lot of vasopressor is used. This is thus the state most patients are in during the big middle part of the surgery. A simulation of the corresponding Vector Autoregression shows that it is very smooth. This corresponds to a low covariance, or a very predictable part. This is not surprising as a lot of vasopression is used. Other states that occur a lot in the big middle part of the surgery are states 17, 18. Here, less vasopression is used. This results in more jagged simulations, because the time series is less predictable.

The states mostly occurring at the start of the surgery are quite similar. They are all quite unpredictable. In addition, there is a relatively high blood pressure and heart rate. These values are especially high in state 0. Lastly, there is little vasopression used in all states.

In general, simulating these VARs shows that some are stable and some are not stable. Interestingly, from all VARs corresponding to states that occur frequently, only the VAR corresponding to state 9 is increasing a bit. All really unstable VARs correspond to states that hardly occur. These states are probably overfitting on a few data points and therefore show unrealistic behaviour.

Several general conclusions can be drawn. First, we have seen that the prior influences the occurrence of states. In addition, there are many associations between state properties. The amount of vasopressor usage is related to the smoothness or covariance of the VAR corresponding to the state. The vasopressor usage is also related to the

mean heart frequency, mean blood pressure and VAR covariance (see figures 6.6 and 6.9). These relations are all known and to be expected. What is interesting is that these things differ over the states. The states thus specialized in a part of the spectrum of the relation. Next to that, the states are occurring at different parts of the surgery. Three parts can be identified, a start, a large middle part, and a short final part. The states dominant in each of these parts have the mean hemodynamic values which is as to be expected from that part. The states thus largely seem to coincide with general patterns in surgery. However, no prognostic value could be found for postoperative variables, including troponin T.

7

DISCUSSION

In the previous three chapters, many results were presented. In this chapter, the results are interpreted and combined to answer relevant questions. First, it will be discussed whether the chosen model is a good way of modelling intraoperative hemodynamic data. Next, the difference between the results corresponding to the estimation with the Gibbs sampler and the EM algorithm is investigated. Finally, a closer look is taken on whether postoperative troponin t is predictable from the states resulting from a fitted MSVAR.

7.1. MODEL CORRECTNESS

In this section, it is tried to answer the question whether the chosen model is an appropriate model for modelling intraoperative hemodynamics. The first notable thing is the interpolation of the data as described in section 3.2.3. For the estimation of (MSV)AR models, it is required that the data is equidistant. It was chosen to interpolated the data to 1-minute intervals. However, in section 4.2 it was shown that the interpolation had a large effect on the resulting model. Among other things, it was decided to interpolated the data to a 2-minute interval. Most immediate problems (state switches corresponding to interpolated and non-interpolated points) were solved. However it remains debatable that all data points are treated equally, whereas some are real measurements and some are interpolated with the measurements and thus include a lot more uncertainty. It might be a good idea to weight the interpolated points as there is already the weighted estimation of vector autoregressions in MSVAR. An even better approach might be Bayesian imputation. One can treat the missing data points as latent variables and sample them each step of the Gibbs sampler. One could take the distribution of all measured values as a prior. The posterior would be based on the prediction of the corresponding VAR and on the effect of predictions by VARs on future points. One could not obtain the posterior directly, however, it is good enough for the Metropolis-Hastings algorithm (see 2.5. One could use a Metropolis-Hastings step in the Gibbs sampler to obtain samples for the necessary, but unobserved data points. This is more in line with

Bayesian principles and adequately addresses the uncertainty in the currently interpolated values.

Given the interpolated values, the MSVAR seems suitable to model the hemodynamic data. In figures 4.9 (heart rate and blood pressure) and C.2 (also with ST segments) it was shown that the MSVAR is more suitable than the other considered models, namely, the PooledVAR and the PanelVAR. The best AIC and BIC scores corresponded to the MSVAR model. Next to that, patients switched states and states were shared among patients. Several intraoperative variables differed a lot over the different states, indicating that states specialize in certain situations. Hence, the states add value to the time series modelling. In [25] it was already claimed that the rich dynamical structure of hemodynamic time series is best captured by an MSVAR model. However, in [20] it was shown that is probably cointegration in the data with Johansen test [68]. Cointegration was not considered in this thesis. The extension of the VAR model dealing with cointegration is VECM [32]. Equivalently, there also is the MSVECM, which can also be estimated in Bayesian fashion [69]. The VECM and MSVECM were not considered in this thesis and especially the MSVECM might be even more appropriate than the MSVAR. From the considered models, the MSVAR was by far the most suitable. Another pro for the MSVAR is that it is a model with a lot of coefficients, but it remains interpretable. The idea of patients being in several states speaks to medical professionals.

In conclusion, the MSVAR is the most suitable model of the considered models. The MSVECM might be an even more appropriate extension. The preprocessing of the data, especially the interpolation, is debatable. Especially Bayesian imputation seems like a good alternative to the current interpolation, because it fully uses that Bayesian principle that everything that is not observed is a random variable.

7

7.2. DIFFERENCE BETWEEN EM ALGORITHM AND GIBBS SAMPLER

The second question that should be answered is the difference in results between estimating using the Gibbs sampler and the EM algorithm. First, the theoretical differences are discussed. Next, the similarities in the implementations in this thesis are discussed. Finally, the results between the algorithms are compared. In theory, the biggest differences are in the foundations of the algorithm. The EM algorithm is an algorithm for maximum likelihood estimation of models with latent variables. The Gibbs sampler is an MCMC method for sampling from a posterior distribution in Bayesian estimation. The result of the EM algorithm is a point estimate for all model parameters without obvious uncertainty estimates. The result of the Gibbs sampler is a chain of samples from the posterior distribution. Point estimates can be obtained by taking the posterior mean or median. The sample from the posterior, if large enough, also gives a good estimate for the uncertainty. Next to that, one can incorporate prior beliefs into the model.

In this thesis, a Bayesian version of the MSVAR model was developed by selecting appropriate priors. Additionally, given these priors, a Gibbs sampler was developed to estimate the model in Bayesian fashion. This model was extended to an hierarchical model by equipping the hyperparameters with prior distributions. The Gibbs sampler worked and converged. The hierarchical model showed that most priors were appro-

priate. The prior covariance was the only prior which is retrospectively quite debatable. The large variations in prior covariance over the states resulted in states that were not occurring at all. Having the prior covariances differing less over the states would have made the model more appropriate.

Although the theoretical foundations for the Gibbs sampler and EM algorithm are quite different, the implementations are surprisingly similar. In the implementation of the EM algorithm, it was eventually decided to use shrinkage for the transition matrix (see 4.2). There are many cases in which a maximum likelihood estimator with shrinkage is equal to the posterior mean of some conjugate Bayesian model. An example is simple linear regression. The posterior mode (MAP) is equal to the estimate from maximum likelihood estimator with l2-regularization [70]. In this case, the equations for the transition matrix are surprisingly similar. Next to that, both are iterative methods, where subsets of parameters are updated sequentially. Both algorithms can get stuck in local minima. The EM algorithm can be initiated randomly several times to reduce this problem. The Gibbs sampler includes a lot of randomness and a burn-in period to get out of local minima. As such, the implementations have a lot of similarities. Both result in an estimate of an MSVAR, one with point estimates and the other with a sample from the posterior distribution. If the posterior mean is taken as a point estimate than the EM algorithm with shrinkage is very comparable to the Gibbs sampler.

However, the main question is whether the estimation algorithm influenced the obtained results. The uncertainty estimate was hardly used, only to see that the Gibbs sampler did converge. The priors were mainly used to get a transition matrix with some desired properties. Shrinkage in the EM algorithm did the same. With the hierarchical model, it was shown that many of the priors hardly influenced the resulting model. The only real influential prior, besides the one on the transition matrix, was the prior on the covariance, which was differing among the states. The prior caused many states to hardly occur at all, because the prior induces covariances not appropriate for any of the data. Thus, in practice the main differences are that the Gibbs simpler can give easy uncertainty estimates, however unused states are easily created with priors that are differing.

In conclusion, although based on very different principles, the EM algorithm and Gibbs sampler are surprisingly similar. This is especially the case when shrinkage is applied in the EM algorithm and only point estimates are used from the Gibbs sampler. In practice, the difference was the available uncertainty estimates for the Gibbs sampler. Additionally, different priors for different states also lead to states hardly occurring.

7.3. PREDICTING TROPONIN T FROM TIME IN STATES

The final question discussed in this chapter is more related to the main goal of this thesis. The question is to what extend it is possible to predict postoperative troponin T from the time in states in an MSVAR. The prediction of troponin T and other postoperative variables was mainly discussed in section 6.2. The main results are in Figure 6.3. It is quite clear that the states hold no prognostic value nor additional prognostic value for troponin T or other postoperative variables. The analysis with multiple prediction models makes sure that there is no model dependent association. This is on the contrary to, for instance [21], where statistical functionals were used to summarize intraoperative

data and some prognostic value was found.

The states do thus not hold any prognostic value, however the states are shown to be interpretable in other ways. Chapter 6 shows several of these interpretations. The main relation is that three clear surgery parts can be established: a start, a long middle part, and a short finish. Furthermore, states seem to be related to both vasopressor usage and mean hemodynamic values. The vasopressor usage seems to relate to more predictable parts of the hemodynamic data. The states thus more or less seem to coincide with the factors external to the patient during surgery instead of the state of the patient itself.

The states are thus interpreted entirely different than in the series of papers [25, 26, 27]. There, a very similar approach was applied to patients in the ICU. These patients are monitored similarly, however, the result in predicting postoperative variables are quite different. Before execution of the research, a preliminary analysis of the differences and similarities was done (see section 3.1.2). Now, possible explanations for the differences in the results are given. First of all, the data used in these papers is over a longer period, at least 24 hours, whereas the surgery data is usually about 2 hours. This means that periods like the start and end are not significant. However, more importantly, during surgery many external shocks are applied to the patient. The hemodynamics of a patient is a feedback control system, which continuously adapts to the state of the body. During surgery, many shocks are applied to this system like induction, the moment cuts are made, or the wounds are sealed. The hemodynamics change accordingly and thus the states are related to these external circumstances, instead of the actual patient state, which is not so important anymore. These differences are apparently so significant that all prognostic value for postoperative variables disappears.

In future research, it might be beneficial to incorporate many of the external circumstances into the model. This will result in that a state change is caused by a patient's reaction to a change in circumstances rather than a change in external circumstances. For instance, the amount of supplied vasopressor could be incorporated. Other external circumstances are much harder to incorporate, like the amount of stress put on the body by the medical intervention. States that actually relate to the patient's state probably hold prognostic value for the postoperative values. The patient is the constant factor from the surgery to the postoperative variables. Prognostic value should thus be found in describing the patient's state. Furthermore, it might also be good to look at the time in states at several different time points. Some time points might be informative and other not.

Finally, the idea is to see the effect of the surgery on troponin levels. However, many patients have chronic cardiac problems before surgery already. These patients have chronically elevated troponin T levels. In [22] it was already shown that preoperative and postoperative hsTnT are highly correlated. However, the change in troponin T is hard to model, as only a small number of patient's have preoperative troponin T measurements. More consistent hsTnT measurements, especially preoperatively, are necessary to have a better picture in what way the surgery influences the troponin T level.

In conclusion, the relative time in states of the fitted MSVAR models does not have any prognostic value for postoperative variables including troponin T. The states are interpretable in other ways, like time in surgery, vasopressor usage, and mean hemodynamic values. It seems that the states are mostly related to the circumstances external to

the patient instead of the patient state. State changes occur because of a change in external circumstances rather than a change in the patients reaction to the circumstances. This probably explains why the states do not have any prognostic value. Finally, more consistent hsTnT measurements, especially preoperatively might allow for a better understanding of the effect of surgery on the troponin T level.

8

Conclusion

Many surgeries result in cardiac complications irrespective of the underlying condition. A good marker for cardiac problems is the troponin T level in the blood, which is usually measured postoperatively. Little is known on the pathology of troponin T release in the blood stream during surgery. The goal of this thesis is to find patterns in intraoperative surgery data related to postoperative troponin T measurements. The chosen method was to fit a Markov-Switching Vector Autoregression in two separate ways, the EM algorithm and Gibbs sampler. The relative time a patient is in a state would be used as features for predicting the postoperative troponin T level.

In this thesis, it was shown that the MSVAR model is appropriate for modelling intraoperative hemodynamic data. The model has a high (log-)likelihood relative to the number of parameters, is interpretable and the states of MSVAR model specialized in different surgery circumstances. Interpolation of the intraoperative data seems dubious and it would be better to do Bayesian imputation for instance. Next to that, cointegration was not considered in this thesis although it is probably present. It would be good to research the possibility of using an MSVECM for modelling intraoperative hemodynamic data.

The Bayesian MSVAR model for intraoperative hemodynamic data was developed by providing the parameters with prior distributions. The corresponding Gibbs sampler was developed by using the conjugacy of the priors. The model was extended to an hierarchical model by adding prior distributions to the hyperparameters for which the Gibbs sampler could easily be extended.

Both the EM algorithm and the Gibbs sampler were used for estimating the MSVAR. These algorithms come from different theoretic principles. However, in practice, especially with using shrinkage in the EM algorithm, the algorithms are very similar. Using prior information is similar as to applying shrinkage. The Gibbs sampler gives a sample of the posterior and therefore an estimate of coefficients distribution instead of a point estimate.

Finally, the goal was to predict postoperative troponin T. It turned out that the relative time in states does not hold any prognostic value for prediction troponin T. It seems to

be due to the amount of external shocks applied to the patient during surgery. The states seem more related to these circumstances than to the patient state. It is recommended to incorporate more of these external circumstances into the model.

In conclusion, this thesis presents a way to model intraoperative hemodynamic data. The MSVAR is an appropriate model and can be estimated both based on frequentist and Bayesian principles, although in practice the differences are small. The states are related to many variables related to the surgery, but do not hold any prognostic value for troponin T.

Appendices

A

STATE CORRELATION MATRICES

This appendix includes the matrices with correlation coefficients between the state probabilities resulting from the most likely model in different runs of the EM algorithm. It should help answer the question of whether the different runs converge towards similar or totally different maxima.

A

	0	1	2	3	4
0	-0.02	-0.75	**0.98**	-0.07	-0.02
1	**0.84**	-0.31	0.04	0.12	-0.01
2	-0.33	**1.00**	-0.74	-0.25	-0.44
3	0.09	-0.44	-0.13	0.04	**0.92**
4	0.26	-0.24	-0.06	**0.93**	-0.05

(a)

	0	1	2	3	4
0	-0.33	**0.99**	-0.74	-0.24	-0.44
1	0.70	-0.31	0.04	0.42	-0.06
2	0.00	-0.74	**0.95**	-0.10	0.00
3	0.23	-0.24	-0.04	**0.84**	-0.02
4	0.12	-0.45	-0.10	0.00	**0.91**

(b)

	0	1	2	3	4
0	-0.01	-0.44	-0.08	-0.06	**0.96**
1	**0.82**	-0.31	-0.00	0.34	-0.03
2	-0.33	**1.00**	-0.72	-0.25	-0.47
3	0.37	-0.28	-0.07	**0.82**	0.03
4	0.01	-0.73	**0.98**	-0.06	-0.08

(c)

	0	1	2	3	4
0	0.44	-0.26	-0.08	0.78	-0.01
1	0.05	-0.54	0.19	0.04	0.65
2	-0.08	-0.54	0.65	-0.13	0.13
3	0.55	-0.43	0.30	0.20	-0.06
4	-0.31	**0.96**	-0.70	-0.22	-0.44

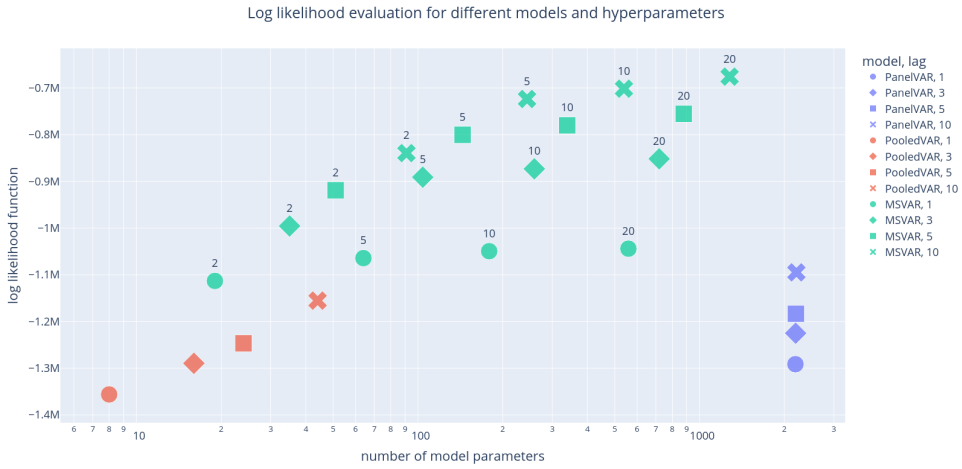
(d)

Figure A.1: For each of the 4 runs of fitting with the EM algorithm not containing the best model, the state probabilities are correlated with the state probabilities of the model resulting from the best run. The numbers labelling the rows and columns are the state numbers. The correlations of 0.8 or eight are high-lighted. One can see that except for the last one the others are almost only a state permutation different from the best run.

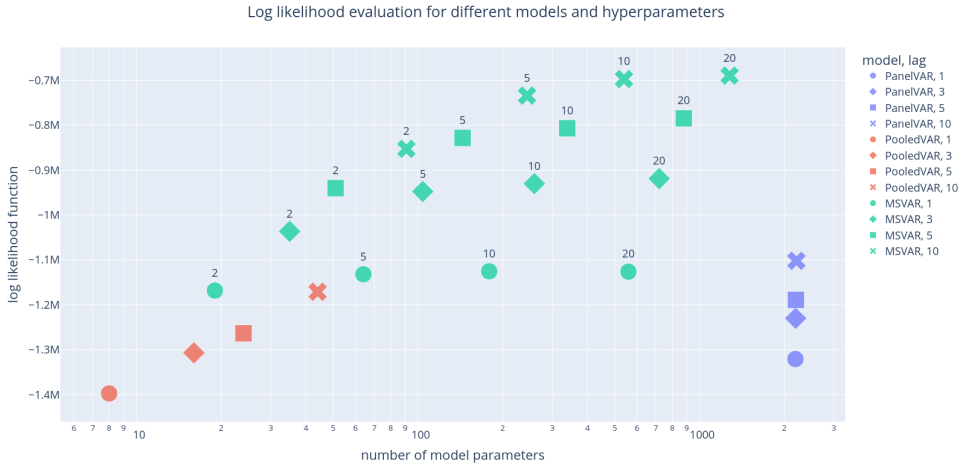
B

MODEL SELECTION PLOTS

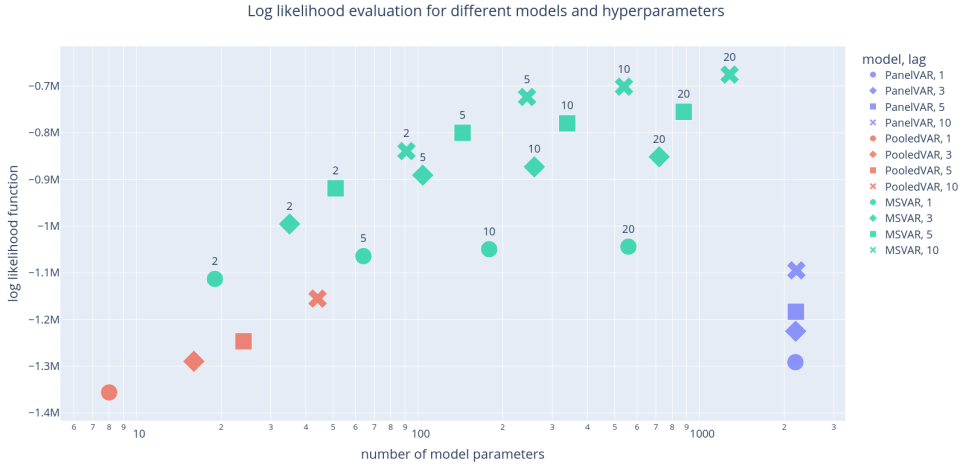
In section 4.4 the model comparison scatter plot for untransformed data was shown. In this appendix, the model comparison plots for the transformed intraoperative data are shown. The transformations are: log-transformations, differencing and the difference of the logs. All, including the one corresponding to the untransformed data (see Figure 4.9, are very similar. No, reason for any data transformation can be established.



(a) Differenced data



(b) Log-transformed data



(c) Log-transformed and differenced data

Figure B.1: Scatter plot of all considered models with the number of free model parameters on the x-axis and the log likelihood on the y-axis. The models are Pooled VAR (red), Panel VAR (blue), (shrunken) MSVAR (green), which also differ in lag (symbol) and for the MSVAR in the number of states (added text).

C

RESULTS EM ESTIMATION FOR HEMO-ST MODEL

In chapter 4 the result of estimation of a MSVAR model with the EM algorithm was shown. These results were obtained by fitting the model to the hemodynamics data (heart frequency and blood pressure). This model can easily extended by adding the st data as three extra endegenous variables. The results for fitting this model are in this chapter. However, is these results are similar as for fitting a MSVAR to hemodyanmics data the details are skipped. Next, data transformation did not seem benificiary for the hemodynamic model and will therefore not be used in this expanded model.

The convergence of the likelihood is shown in Figure C.1. The convergence seems similar as in Figure 4.5, except that the iterations are stopped a lot earlier. Local optima are found earlier. This seems counterintuitive as the model consists of a lot more parameters. Intuitively, one would say that is harder to optimize for more parameters. It could be because with many more parameters there are more local optima. Another explanation could be just randomness as the algorithm is initialized randomly each time.

Next to the convergence graph, the model scatter graph is also presented for the hemodynamic-st data combination. Several interesting things can be observed. First of all, there is the reassurance that the log likelihood is lower for every single model in 4.9. This is as expected as the new dataset is a superset of the old dataset and the model parameters are a superset of the old model parameters. Secondly, the distribution of the models over the spectrum is similar in general. It is therefore unsurprising that according to both the AIC and BIC scores, the model with 10 lags and 20 states is best. Thirdly, some models are oddly placed with respect to their neighbours, for instance, the model with 10 states and 10 lags. It has a lower log likelihood than the same model with 5 lags. This indicates that the EM algorithm gets stuck in local maxima earlier and struggles to find the global maximum due to the extra number of parameters.

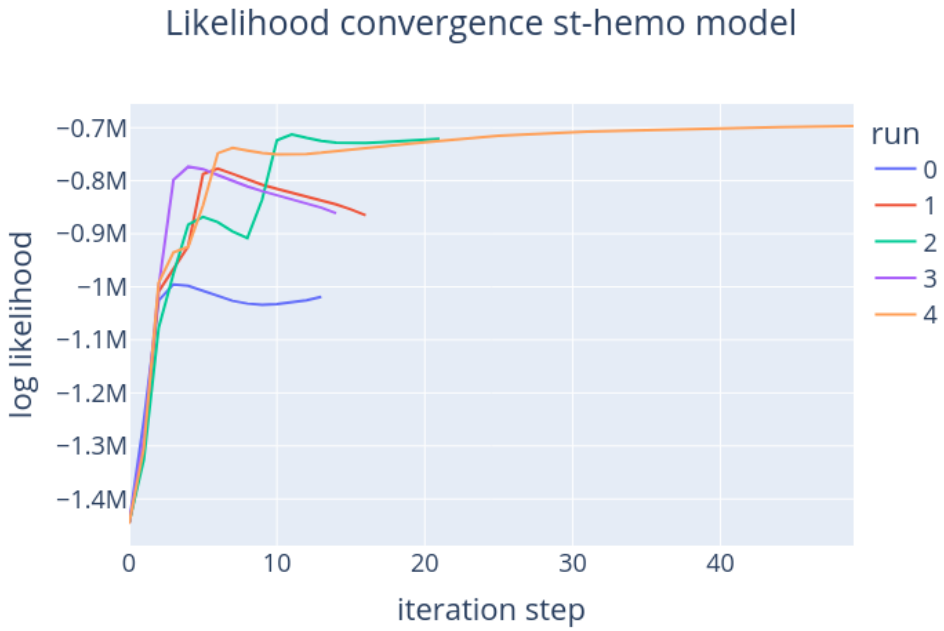


Figure C.1: Trace of the likelihood over the iterations for 5 runs of fitting a MSVAR.

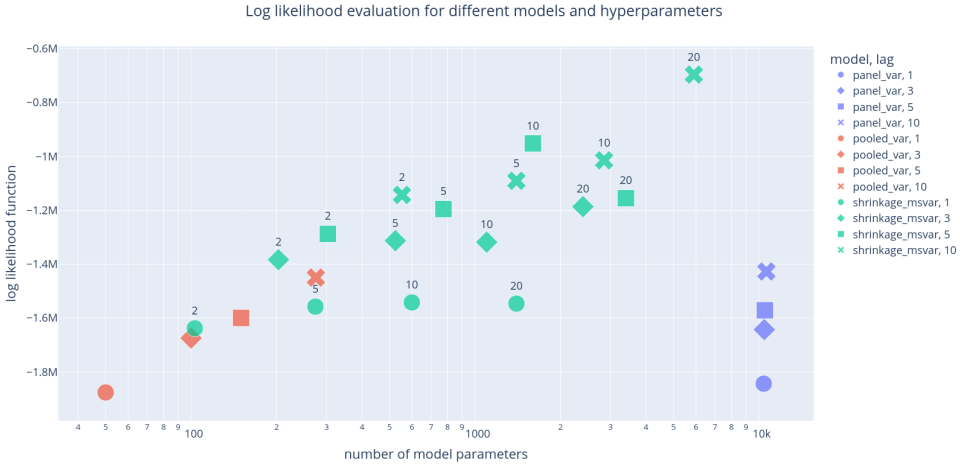


Figure C.2: Scatter plot of all considered models with the number of free model parameters on the x-axis and the log likelihood on the y-axis. The models are Pooled VAR (red), Panel VAR (blue), (shrunken) MSVAR (green), which also differ in lag (symbol) and for the MSVAR in the number of states (added text).

D

SURGERY SELECTION

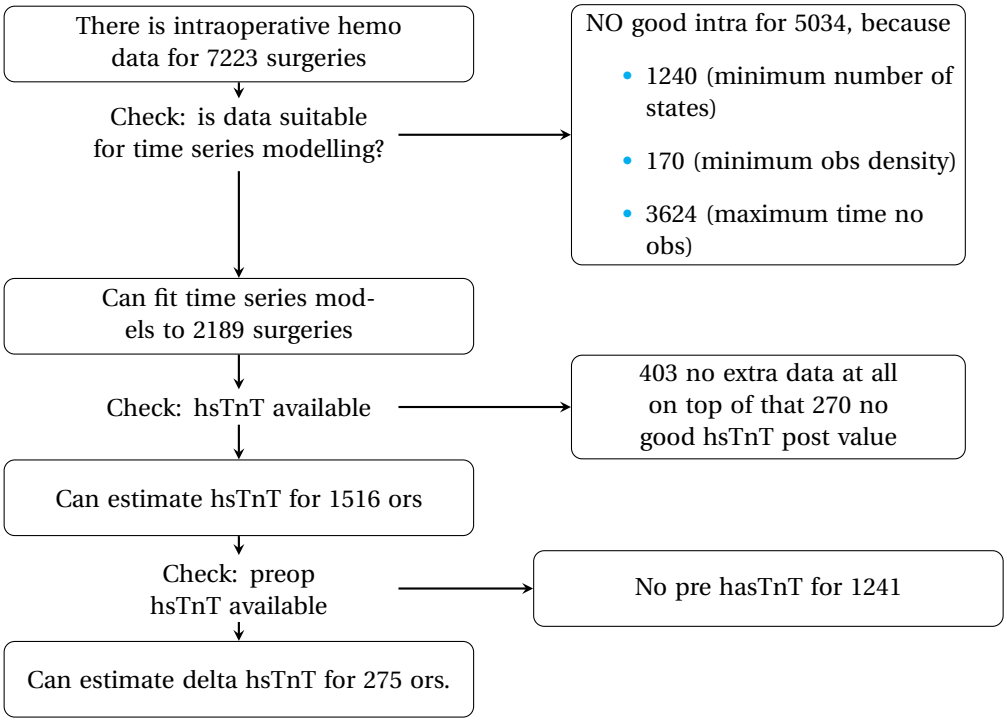


Figure D.1: The different checks and there results. There are 7223 different surgeries with hemodynamic data. 2189 surgeries have hemodynamic data that is suitable for MSVAR modelling. From these 2189 1516 have postoperative hsTnT measurements and 275 have preoperative hsTnT measurements as well.

E

PREOPERATIVE FEATURES

In predicting postoperative variables, often next to intraoperative features preoperative features are used as well.

- patient's age
- patient's gender
- surgery type
- medicine usage:
  - diuretics
  - atii
  - oac
  - ccb
  - ace
  - statins
  - aspirin
  - bb
- known conditions:
  - HT
  - MI
  - RENAL
  - RENAL type
  - PAD
  - DM type
  - CHF
  - CVA

F

STATE COLOURS

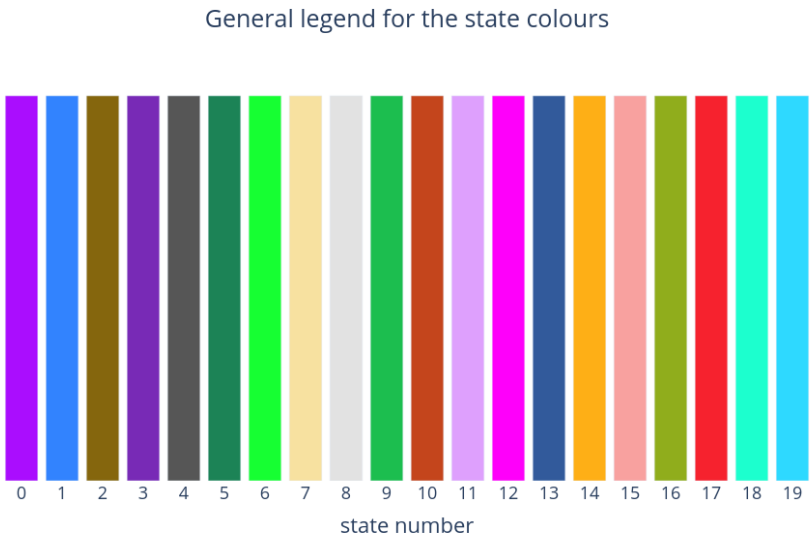


Figure F.1: The legend for many graphs, where colours are used to distinguish between states. These color labellings are used throughout the report.

BIBLIOGRAPHY

[1] Dmitri Nepogodiev et al. “Global burden of postoperative death”. In: *The Lancet* 393.10170 (2019), p. 401. ISSN: 0140-6736. DOI: https://doi.org/10.1016/S0140-6736(18)33139-8.

[2] Rupert M Pearse et al. “Mortality after surgery in Europe: a 7 day cohort study”. In: *The Lancet* 380.9847 (2012), pp. 1059–1065. ISSN: 0140-6736. DOI: https://doi.org/10.1016/S0140-6736(12)61148-9.

[3] Judith AR van Waes et al. “Myocardial injury after noncardiac surgery and its association with short-term mortality”. In: *Circulation* 127.23 (2013), pp. 2264–2271.

[4] PJ Devereaux et al. “Characteristics and short-term prognosis of perioperative myocardial infarction in patients undergoing noncardiac surgery: a cohort study”. In: *Annals of internal medicine* 154.8 (2011), pp. 523–528.

[5] PJ Devereaux et al. “Association of postoperative high-sensitivity troponin levels with myocardial injury and 30-day mortality among patients undergoing noncardiac surgery”. In: *Jama* 317.16 (2017), pp. 1642–1651.

[6] on behalf of The Vascular events In noncardiac Surgery patIents cOhort evaluatioN (VISION) Investigators Vascular events In noncardiac Surgery patIents cOhort evaluatioN (VISION) Writing Group. “Myocardial injury after noncardiac surgery: a large, international, prospective cohort study establishing diagnostic criteria, characteristics, predictors, and 30-day outcomes”. In: *Anesthesiology* 120.3 (2014), pp. 564–578.

[7] Emery N Brown, Ralph Lydic, and Nicholas D Schiff. “General anesthesia, sleep, and coma”. In: *New England Journal of Medicine* 363.27 (2010), pp. 2638–2650.

[8] John H Eichhorn et al. “Standards for patient monitoring during anesthesia at Harvard Medical School”. In: *Jama* 256.8 (1986), pp. 1017–1020.

[9] Jian-Ping Jin. “Evolution, regulation, and function of N-terminal variable region of troponin T: Modulation of muscle contractility and beyond”. In: *International review of cell and molecular biology* 321 (2016), pp. 1–28.

[10] K. H.J.M. Mol et al. “Postoperative troponin release is associated with major adverse cardiovascular events in the first year after noncardiac surgery”. In: *International Journal of Cardiology* 280 (2019), pp. 8–13. ISSN: 18741754. DOI: 10.1016/j.ijcard.2019.01.035.

[11] W Scott Beattie et al. “Use of clinically based troponin underestimates the cardiac injury in non-cardiac surgery: a single-centre cohort study in 51,701 consecutive patients”. In: *Canadian Journal of Anesthesia/Journal canadien d’anesthésie* 59.11 (2012), pp. 1013–1022.

[12] Peter Nagele et al. “Postoperative myocardial injury after major head and neck cancer surgery”. In: *Head & neck* 33.8 (2011), pp. 1085–1091.

[13] V. G.B. Liem et al. “Prognostic value of postoperative high-sensitivity troponin T in patients with different stages of kidney disease undergoing noncardiac surgery”. In: *British Journal of Anaesthesia* 120.1 (2018), pp. 84–93. ISSN: 14716771. DOI: 10.1016/j.bja.2017.09.003.

[14] PG Noordzij et al. “High-sensitive cardiac troponin T measurements in prediction of non-cardiac complications after major abdominal surgery”. In: *BJA: British Journal of Anaesthesia* 114.6 (2015), pp. 909–918.

[15] F. van Lier et al. “Association between postoperative mean arterial blood pressure and myocardial injury after noncardiac surgery”. In: *British Journal of Anaesthesia* 120.1 (2018), pp. 77–83. ISSN: 14716771. DOI: 10.1016/j.bja.2017.11.002.

[16] Victor G.B. Liem et al. “Postoperative Hypotension after Noncardiac Surgery and the Association with Myocardial Injury”. In: *Anesthesiology* 133.3 (2020), pp. 510–522. ISSN: 15281175. DOI: 10.1097/ALN.0000000000003368.

[17] Tobias Reichlin et al. “Early diagnosis of myocardial infarction with sensitive cardiac troponin assays”. In: *New England Journal of Medicine* 361.9 (2009), pp. 858–867.

[18] Thomas Van Der Jagt. *Erasmus MC internship*. Tech. rep. 2020.

[19] Daniël Hoonhout. *Vector autoregressive modelling of stochastic medical processes*. Tech. rep. 2021.

[20] Abdellah Rissalah. *Utilising Vector Error Correcting Modelling for Intra-operative Data*. Tech. rep. 2021.

[21] Laura Veerhoek. *Patterns in intraoperative data that are related to and could explain elevated troponin levels in patients undergoing major non-cardiac surgery*. Tech. rep. 2021.

[22] Antoine Pomari. *High-sensitivity troponin T elevations and the problem of feature selection : a study using Bayesian Networks and Lasso regression*. Tech. rep. 2021.

[23] Shounak Datta et al. “Added value of intraoperative data for predicting postoperative complications: the MySurgeryRisk PostOp extension”. In: *Journal of Surgical Research* 254 (2020), pp. 350–363.

[24] Bing Xue et al. “Use of machine learning to develop and evaluate models using preoperative and intraoperative data to identify risks of postoperative complications”. In: *JAMA network open* 4.3 (2021), e212240–e212240.

[25] Li Wei H. Lehman et al. “Tracking progression of patient state of health in critical care using inferred shared dynamics in physiological time series”. In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2013), pp. 7072–7075. ISSN: 1557170X. DOI: 10.1109/EMBC.2013.6611187.

[26] Li Wei H. Lehman et al. “A physiological time series dynamics-based approach to patient monitoring and outcome prediction”. In: *IEEE Journal of Biomedical and Health Informatics* 19.3 (2015), pp. 1068–1076. ISSN: 21682194. DOI: 10.1109/JBHI.2014.2330827.

[27] Li Wei H. Lehman, Shamim Nemati, and Roger G. Mark. “Hemodynamic monitoring using switching autoregressive dynamics of multivariate vital sign time series”. In: *Computing in Cardiology* 42 (2015), pp. 1065–1068. ISSN: 2325887X. DOI: 10.1109/CIC.2015.7411098.

[28] Robert H. Shumway and David S. Stoffer. *Time Series Analysis and Its Applications*. Vol. 53. Springer Texts in Statistics 3. Cham: Springer International Publishing, 2017, pp. 331–331. ISBN: 978-3-319-52451-1. DOI: 10.1007/978-3-319-52452-8.

[29] Wayne A. Fuller. *Introduction to statistical time series*. John Wiley &; Sons, 1976.

[30] G. E. P. Box and D.R. Cox. “An Analysis of Transformations”. In: *Journal, Source Statistical, Royal Series, Society* 26.2 (1964), pp. 211–252.

[31] Peter Whittle. *Hypothesis testing in time series analysis*. Vol. 4. Almqvist & Wiksells boktr., 1951.

[32] Helmut Lütkepohl. *New introduction to multiple time series analysis*. 2005, pp. 1–764. ISBN: 3540401725. DOI: 10.1007/978-3-540-27752-1.

[33] Badi H. Baltagi, James M. Griffin, and Xiong Weiwen. “To Pool or Not to Pool: Homogeneous versus Heterogeneous Estimators Applied to Cigarette Demand”. In: *The MIT Press* 82.1 (2000), pp. 117–126.

[34] James D. Hamilton. *A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle*. 1989. DOI: 10.2307/1912559.

[35] Sean Meyn and Richard L. Tweedie. *Markov chains and stochastic stability*. 2009, pp. 1–504. ISBN: 9780511626630. DOI: 10.1017/CBO9780511626630.

[36] Leonard E Baum et al. “A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains”. In: *The annals of mathematical statistics* 41.1 (1970), pp. 164–171.

[37] Arthur P Dempster, Nan M Laird, and Donald B Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (1977), pp. 1–22.

[38] CF Jeff Wu. “On the convergence properties of the EM algorithm”. In: *The Annals of statistics* (1983), pp. 95–103.

[39] James D. Hamilton. *Time series analysis*. Princeton University Press, 1994. DOI: 102307.

[40] Chang-Jin Kim. “Dynamic linear models with Markov-switching”. In: *Journal of Econometrics* 60.1-2 (1994), pp. 1–22.

[41] Nestor Parolya. personal communication. Mar. 25, 2022.

[42] Taras Bodnar, Arjun K Gupta, and Nestor Parolya. “On the strong convergence of the optimal linear shrinkage estimator for large dimensional covariance matrix”. In: *Journal of Multivariate Analysis* 132 (2014), pp. 215–228.

[43] Chain Monte Carlo. “Markov chain monte carlo and gibbs sampling”. In: *Lecture notes for EEB* 581 (2004), p. 540.

[44] Nicholas Metropolis et al. “Equation of state calculations by fast computing machines”. In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

[45] W Keith Hastings. “Monte Carlo sampling methods using Markov chains and their applications”. In: (1970).

[46] Stuart Geman and Donald Geman. “Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images”. In: *IEEE Transactions on pattern analysis and machine intelligence* 6 (1984), pp. 721–741.

[47] Sylvia Frühwirth-Schnatter and Sylvia Frèuhwirth-Schnatter. *Finite mixture and Markov switching models*. Vol. 425. Springer, 2006.

[48] Fabio Canova. “10. Bayesian VARs”. In: *Methods for Applied Macroeconomic Research*. Princeton University Press, 2011, pp. 373–417.

[49] Robert B Litterman et al. *Techniques of forecasting using vector autoregressions*. Tech. rep. 1979.

[50] Hirotugu Akaike. “A new look at the statistical model identification”. In: *IEEE transactions on automatic control* 19.6 (1974), pp. 716–723.

[51] Gideon Schwarz. “Estimating the dimension of a model”. In: *The annals of statistics* (1978), pp. 461–464.

[52] Mourad Kedadouche, Zhaoheng Liu, and Marc Thomas. “Feature extraction and selection using autoregressive coefficient and linear discriminant analysis : Application to bearings operating in variable conditions”. In: November (2016).

[53] Hervé Abdi and Lynne J Williams. “Principal component analysis”. In: *Wiley interdisciplinary reviews: computational statistics* 2.4 (2010), pp. 433–459.

[54] Mohammed Saeed et al. “Multiparameter Intelligent Monitoring in Intensive Care II (MIMIC-II): a public-access intensive care unit database”. In: *Critical care medicine* 39.5 (2011), p. 952.

[55] Isaac J Schoenberg. “Contributions to the problem of approximation of equidistant data by analytic functions”. In: *IJ Schoenberg Selected Papers*. Springer, 1988, pp. 3–57.

[56] Paul Dierckx. “An algorithm for smoothing, differentiation and integration of experimental data using spline functions”. In: *Journal of Computational and Applied Mathematics* 1.3 (1975), pp. 165–184.

[57] Paul Dierckx. “An improved algorithm for curve fitting with spline functions”. In: *TW Reports* (1981).

[58] Jan Salomon Cramer. “The origins of logistic regression”. In: (2002).

[59] David W Hosmer and Stanley Lemeshow. “Applied Logistic Regression. John Wiley & Sons”. In: *New York* (2000).

[60] Leo Breiman. “Random forests”. In: *Machine learning* 45.1 (2001), pp. 5–32.

[61] Leo Breiman et al. “Classification and regression trees. Belmont, CA: Wadsworth”. In: *International Group* 432 (1984), pp. 151–166.

[62] J. Ross Quinlan. “Induction of decision trees”. In: *Machine learning* 1.1 (1986), pp. 81–106.

[63] Leo Breiman. “Bagging predictors”. In: *Machine learning* 24.2 (1996), pp. 123–140.

[64] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[65] Wojciech Kwedlo. “A new random approach for initialization of the multiple restart EM algorithm for Gaussian model-based clustering”. In: *Pattern Analysis and Applications* 18.4 (2015), pp. 757–770.

[66] Matthew Stephens. “Dealing with label switching in mixture models”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 62.4 (2000), pp. 795–809.

[67] Gilles Celeux. “Bayesian inference for mixture: The label switching problem”. In: *Compstat*. Springer. 1998, pp. 227–232.

[68] Søren Johansen. *Likelihood-based inference in cointegrated vector autoregressive models*. OUP Oxford, 1995.

[69] Seuk Wai Phoong and Siok Kun Sek. “A Markov switching vector error correction model on oil price and gold price effect on stock market returns”. In: *Information Management and Business Review* 5.7 (2013), pp. 331–336.

[70] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.