Structural Time Series Modelling for 18 years of Kapenta Fishing in Lake Kariba

Author
Lara Dalmeyer

Supervisor
Dr Birgit Erni

Co-supervisor
Professor Linda Haines

Submitted to the Department of Statistical Sciences in fulfillment of the requirements for the degree of

Master of Business Science in Mathematical Statistics
at the
University of Cape Town

December 3, 2012

Acknowledgements

I am truly indebted and grateful to my supervisor Dr. Birgit Erni for her valuable guidance and continuous support throughout this thesis. Besides, I would like to thank my co-supervisor, Prof. Linda Haines, for her input and advise, both theoretical and presentational.

I am grateful to the National Research Fund (NRF) and Institute of Applied Statistics for funding my studies in both 2011 and 2012.

Additionally, I would like to thank Dr. Res Altwegg and Dr Charles Musil from SANBI who kindly provided the dataset for this thesis, without which the Kapenta application would not have been possible.

I would also like to thank Ms Ndebele-Murisa for her agreement and concession in using the Kapenta dataset for this thesis. This data was originally used in her paper, “The Implications of a changing climate on the Kapenta Fish Stocks of Lake Kariba, Zimbabwe” which was written in collaboration with Trevor Hill and Emmanuel Mashonjowa in 2011.

Abstract

Structural time series models, formulated as linear Gaussian state space models, and ARIMA models are considered for modelling *Limnothrissa miodon*, or Kapenta fish, catch data from Lake Kariba, Zimbabwe. Having considered the advantages and disadvantages of both methodologies, structural time series models are chosen for modelling. The theory of defining, estimating and checking linear Gaussian state space models, and specifically structural time series models, is discussed in this thesis. The data to be modelled is daily Kapenta catch per unit effort (CPUE) data for basin 5 of Lake Kariba, over the period I January 1986 to 31 December 2003. Structural time series models separately model trend, cyclical, seasonal and regression components, and unlike ARIMA models do not need to be rendered stationary prior to modelling. Problems were experienced modelling daily CPUE, and this was attributed to the high sampling frequency and large variance within the data. After summing daily data into weekly data, models explaining more of the variation within the CPUE data were produced. Local linear trend models, a random walk type model, best describes the trend component of weekly CPUE, indicating no significant upward or downward movement in CPUE over the period investigated. Additionally, significant elements in modelling weekly CPUE data include yearly and monthly cyclical components. The yearly cyclical component is attributed to the movement of Kapenta fish in and out of open waters over the winter and summer months respectively for breeding purposes. The monthly cyclical component is attributed to the effect of moonlight (moon cycle= 29.53 days) on the efficacy of attraction lights used for Kapenta fishing, which takes place in the evenings. The effects of temperature, precipitation, lake level and cloud cover were also investigated, but only temperature showed to be significant in explaining additional variation in weekly CPUE. Unknown model parameters can be estimated using maximum likelihood or Bayesian analysis, both methods of which were used in this thesis, and generate the same parameter estimates. Basins 1 to 4 of Lake Kariba are considered in the multivariate extension of the univariate model developed for basin 5. The multivariate model provides a unified approach to modelling all the basins of Lake Kariba, and it is seen that factors affecting CPUE for basin 5 affect CPUE for other basins in Lake Kariba as well. Implementation of multivariate models is, however, much more difficult.

Plagiarism Declaration

1. This dissertation is my own work. It has not been submitted before for any degree or examination to any other University.

2. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as his or her own work.

3. Each significant contribution to, and quotation in, this dissertation from the work of other people has been cited and referenced.

Signature: ……..

Date: …….

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 General Overview

Kapenta (*Limnothrissa miodon*), a small sardine-type fish, was introduced from Lake Tanganyika into Lake Kariba between 1967 and 1969, and it now supports a large and viable fishery for Zimbabwe and Zambia who share the lake. The objective of this thesis is to understand long-term trends, seasonal and cyclical components, and external factors such as cloud cover, water level, temperature and rainfall, which influence Kapenta catch over the period 1 January 1986 to 31 December 2003.

A recent paper by Ndebele-Murisa et al. (2011) investigated how the Kapenta stocks of Lake Kariba are affected by changing climatic and hydrological variables. Catch data were used as a proxy for fish stock. Graphical and regression analyses were used to ascertain any trends in climatic variables, hydrological variables and catch data. Linear regressions were also performed to view any existing relationships between these environmental factors and catch data. Results showed that changes in climatic variables were apparent, and that these changes were affecting the fish stocks of Lake Kariba. The vast data set, consisting of Kapenta catch and climatic variable data in the area, was mainly collected by the Lake Kariba Fisheries Research Station, Zimbabwe Meteorological Services, Kariba Weather Station and Zambezi River Authority, all of which are collected for research purposes. This dataset was made available for the purposes of this

thesis and inspired an idea to perform a more thorough time series analysis on the catch data. This dataset contains daily Kapenta catch (in kg's) per vessel. These values are summed across the vessels to produce one catch value per day. Problems were however experienced in modelling daily Kapenta catch, and daily values were summed to produce weekly values. Modelling weekly Kapenta catch produced stronger models, and resolved problems experienced using daily data.

Autoregressive integrated moving average (ARIMA) models and structural time-series models, which can be formulated as state space models, are both appropriate procedures for modelling time series data. The theory, advantages and disadvantages of these methods are investigated and the most appropriate methodology is selected for performing the time series analysis on Kapenta catch data.

The traditional method of time series modelling is based on the Box-Jenkins methodology of identifying an autoregressive integrated moving average (ARIMA) model by differencing to obtain a stationary series, then using tools such as the sample autocorrelation function, to select the order of the autoregressive and moving average parts. However, major objections regarding these models and their selection methodology have been noted (Harvey, 1997) to the ARIMA approach in recent years. ARIMA models and the concerns regarding these models will be discussed in greater detail in Chapter 2 (§2.2 and §2.3). Another methodology to modelling time series data involves structural time series models. These are formulated as linear Gaussian state space models and are set up in terms of components, such as trend, seasonal and cyclical components, that have direct interpretation. It is also possible to include explanatory variables into these models. These models can be made more flexible by allowing components to be dynamically included, through modelling these components stochastically. Structural time series models avoid having to decide on a degree of integration, as the data does not need to be stationary prior to the model building process (Harvey, 1997). A single model that describes trend, seasonal and variance aspects simultaneously is thus constructed. Structural time series models are discussed in greater detail in Chapter 2 (§2.4 and §2.5) of this thesis, where the

advantages and disadvantages of these models are compared to those of the ARIMA models and methodology.

1.2 Overview of Kapenta Fishing in Lake Kariba

Papers by Madamombe (2002) and Kolding, Musando & Songore (2003) describe Lake Kariba and the Kapenta fishing industry. This lake (length: 277 km; area: 5364 km^2; volume: 160 km^3; mean depth: 29m; max. depth: 120m) is located on the Zambezi River between latitudes 16°28' to 18°04'S and longitudes 26°42' to 29°03'E. It was the largest man-made reservoir in the world at the time of construction, and is today the second largest reservoir in Africa by volume. The construction of the Kariba dam wall occurred between 1955 and 1959. The catchment area covers 663 817km^2 extending over parts of Angola, Zambia, Namibia, Botswana and Zimbabwe. The lake is naturally divided into 5 basins and is almost equally shared by the two countries Zambia and Zimbabwe; as shown below in Figure 1.1.

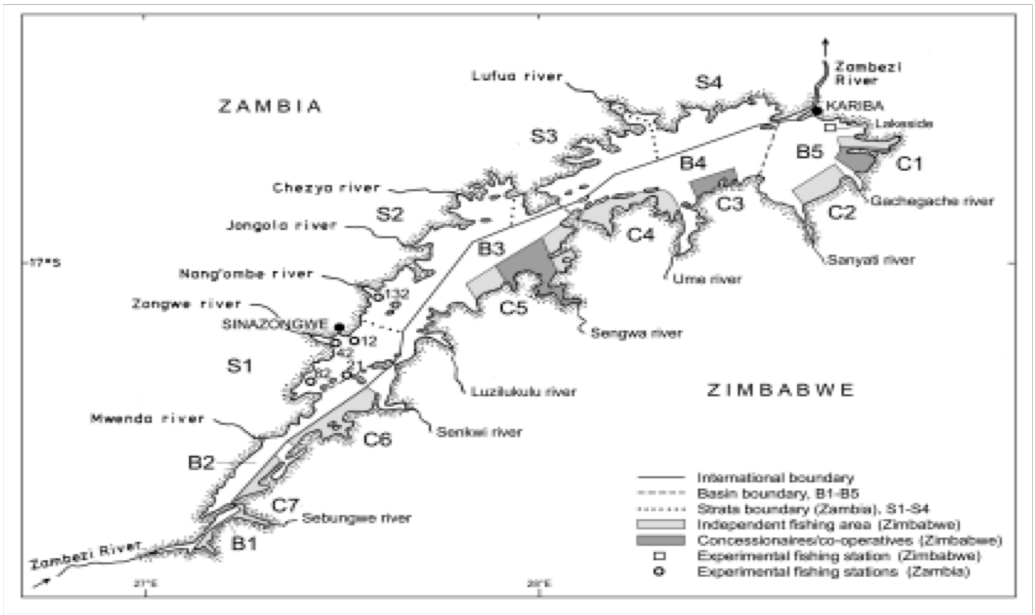


Figure 1.1. Map of Lake Kariba showing the five natural basins (B1 to B5), the international boundary between Zambia and Zimbabwe, as well as Kariba town situated in Zimbabwe. Kolding, Musando & Songore (2003)

According to Madamombe (2002), the primary objective for constructing the lake was for harnessing water for hydroelectricity supplied to mines in Zambia and for supporting the emerging agricultural and industrial sectors in Zambia. Between 1967 and 1969, Kapenta fish (*Limnothrissa miodon*) were introduced into Lake Kariba from Lake Tanganyika. This was due to the fact that a study by Jackson (1961) predicted that pelagic habitat of Lake Kariba would remain non-colonised since the species present in the Zambezi river had evolved in a riverine habitat and would only inhabit the shallow littoral zones. The introduction was a success and the Kapenta fishing industry has now turned into a million dollar industry, with between 20-30 000 tons landed annually. The Kapenta fishing industry is alone responsible for most of the infrastructural development that has occurred on the Zimbabwean shoreline, according to Bourdillon et al. (1985). The fisheries on the Zimbabwean and Zambian side of the lake undergo different management regimes, and the Zimbabwean side is, compared to the Zambian, more regulated and enforced, resulting in a fishing pressure and fishing pattern which has not changed much over time and where fish stocks are only moderately exploited (Kolding et al., 2003).

Lake Kariba experiences its summer months from September to March, with October known as the hottest month of the year. Temperatures are known to average 40°C in the day over the summer months. The rainy or wet season occurs between late November and April, and winter spans from April to September, still averaging a hot 25°C in the day (Madamombe, 2002). Kapenta fishing takes place in the evenings with lift nets from pontoons and light attractions. In summer months, Kapenta fish move inshore to protected bays to breed, and fish catches are low. In the winter months they move back into the open waters. Kapenta feed primarily on phytoplankton. According to Ndebele-Murisa et al. (2011), phytoplankton depends on sufficient rainfall, and similarly lake levels, to replenish the lake with nutrients necessary for it's biomass and production. Phytoplankton is situated in the upper layers of the lake and responds adversely to increased water temperatures, as higher temperatures drive more stable stratification and nutrients become locked up in the bottom layer of the lake. This adversely affects the phytoplankton in the upper layer of the lake, and this

decrease affects fish stocks. Additionally, higher temperatures increase evaporation rates, adversely affecting water and nutrient levels.

In addition to identifying long-term trends, seasonal and cyclical components, relationships between Kapenta catches and explanatory variables, temperature, water level and precipitation, are investigated. This thesis will also investigate the effect of moonlight on Kapenta catch, as clear monthly cycles are observed in the exploratory data analysis (§4.3). Papers such as Horky et al. (2006) have noted moonlight to have an effect on the behaviour of fish. Moonlight is specifically investigated in this thesis due to the fact that Kapenta fishing takes place in the evenings with light attractions; it is hypothesized that over full moon, when the moon emits most light, the efficacy of the light attractions may be diminished. By contrast, when it is new moon, the efficacy of the light attractions may be enhanced, allowing for greater catch. For similar reasons, the effect of cloud cover over the evenings on Kapenta catch is also investigated. With vast cloud cover in the evenings, light emitted from the moon is diminished and may enhance the efficacy of light attractions.

The paper by Ndebele-Murisa et al. (2011) noted that climatic and hydrological factors (rainfall, lake level, minimum temperature, maximum temperature and evaporation rates) could explain the variation in Kapenta catch, where Kapenta catch was concluded to be decreasing. The paper concluded that lake level was the most significant factor considered in explaining Kapenta catch, followed by maximum temperature, evaporation then rainfall. However, since water levels are largely influenced by other climatic variables, it was concluded that both climatic variables (particularly maximum temperature) and nutrients, which are influenced by water levels, are the main determinants driving Kapenta production. Additionally this study shows that overall, the declines in Kapenta catch observed from 1974 to 2008, are unprecedented since the last two long-term studies. Papers by Marshall (1982 and 1988), Mtaba (1987), Karenge and Harding (1995), Magadza (1996) and Chifamba (2000) all show that there is a relationship between climate, hydrological factors and Kapenta catches. Chifamba (2000) found maximum temperature to be the best predictor of catch; as temperature around Kariba increases, fish production is adversely affected. Papers such as

Marshall (1982) and Magadza (1996) show the importance of hydrological factors, such as rainfall and water level, on catch. Magadza (1980) shows that the spatial distribution of fish is associated with areas of river inflow, and Marshall (1982) concluded that nutrient influxes caused by river inflow and water levels are followed by peaks in fish production. By contrast, Karenge and Kolding (1995) showed that no relationship between catch and absolute water levels is seen to exist, even during periods of droughts.

The objective of the paper by Ndebele-Murisa et al. (2011) was to determine how changes in climatic variables affect Kapenta fish stocks in Lake Kariba. This thesis intends to undertake a broader understanding of what influences Kapenta catch, through building a time series model. More specifically, the objectives of this thesis are as follows:

- To gain a good understanding of time series models, particularly ARIMA and state space models (or more specifically structural time series models). An investigation into the advantages and disadvantages of each of these models (§2.3 and §2.5) led to the decision to use structural time series models for modelling Kapenta catch. The structure of these models, the Kalman filter and smoothing algorithms, frequentist and Bayesian methods for estimating unknown model parameters and multivariate extensions are investigated.
- To gain a good understanding and skill in the application of structural time series models to time series data, through the use of programming packages in R.
- To understand how trend, seasonal, cyclical and regression factors influence Kapenta catches, over the period 1 January 1986 to 31 December 2003. Regression factors considered will include lake level, temperature, precipitation, moonlight and cloud cover.

1.3 Structure of Thesis

In order to document the research study, this thesis is divided into 5 chapters. This introductory chapter introduces the various time series models considered. Additionally it provides a background into the Kapenta fishing industry, and

introduces biological factors and processes that must be considered when building a model for Kapenta catch. Chapter 2 provides a detailed look into the time series models considered for use in this thesis, namely the Box-Jenkins ARIMA models and structural time series models. The motivation for the choice of model chosen is then provided; this will be seen to be structural time series models, a kind of linear Gaussian state space model. Chapter 3 provides the detailed theory and methodology for structural time series models. The formulation of structural time series models are described, including how model parameters are estimated using maximum likelihood. The Kalman filtering and smoothing algorithms are discussed, and explains how they are initialised. Additionally Chapter 3 describes how unknown parameters can alternatively be estimated from a Bayesian approach, and how the model for Kapenta catch in basin 5 can be extended into a multivariate model that includes all basins. Chapter 3 concludes with a section describing the programming package in R, a package called `dlm` (Petris, 2009), which was used for building structural time series models in this thesis. The programming methods and functions of this package used are provided, and any other programming methods and considerations are explained. Chapter 4 presents a detailed look into the Kapenta catch data and provides an exploratory data analysis. Model hypotheses, the model building process using structural time series models and results are presented. Additionally, problems encountered with modelling daily data are discussed and a remedy to this problem is provided. Results from the Bayesian analysis and multivariate extension are also presented in this chapter. Lastly, Chapter 5 presents the conclusion to this thesis, based on the results discussed in Chapter 4 and the objectives outlined in this chapter.

Chapter 2

Statistical Methodology Overview

2.1 Introduction

This chapter provides an overview of time series models. Both Box-Jenkins ARIMA and structural time series models were considered for modelling the Kapenta catch data. Each of these methods were individually investigated, and the advantages and disadvantages of each considered before the most appropriate method was chosen. Section 2.2 describes the mathematical and statistical formulations of Box-Jenkins type ARIMA models, and Section 2.3 provides a discussion of the advantages and disadvantages of these models. Section 2.4 provides an introduction to the theory of structural time series models, but a more detailed description of the theory, processes and practical implementation of these models is discussed in Chapter 3. Section 2.5 discusses the advantages and disadvantages of structural time series models. Section 2.6 concludes with the choice of modelling methodology, and summarises the reasons for this choice.

This thesis is particularly concerned with modelling non-stationary time series data. Both ARIMA and structural time series models are appropriate for modelling this kind of data. First it is important to clarify what is meant by non-stationary time series data. Time series data are stationary if the mean, variance and

covariance of the data do not depend on time or seasonality. This definition refers to weak stationarity, and one may adjust a time series, by differencing for example, to render it as such. Occasionally the condition of strict stationarity is imposed. This is a stronger condition whereby the joint probability distribution of a set of r observations at times $t_1, t_2, \ldots, t_r$ is the same as the joint probability of the observations at times $t_1 + \tau$, $t_2 + \tau$, …, $t_r + \tau$. Strict stationarity implies weak stationarity provided the first two moments of the joint distribution exist (Harvey, 2003). In this thesis the term stationarity refers to weak stationarity. The Kapenta catch data, as will be seen in Chapter 4, show strong seasonal, cyclical and trend components, implying non-stationary data.

2.2 Autoregressive Integrated Moving Average Models

The most popular and widely used methods for modelling non-stationary data (and time series in general) are autoregressive integrated moving average (ARIMA) models. ARIMA models are formed by moving average and autoregressive processes, after differencing the data until stationarity conditions are satisfied. The theory of ARIMA models is briefly investigated in this section and Section 2.3 provides an investigation into the advantages and disadvantages of these models. The standard reference used for the theory and notation of ARIMA models is Chatfield (2004) unless otherwise stated. A more detailed discussion on any of the topics mentioned in this section can be seen there or alternatively in Box and Jenkins (1970) or Harvey (2003).

2.2.1 Moving Average Processes

Suppose that $\{Z_t\}$ is a purely random process with mean zero and variance σ_Z^2. Then a process $\{X_t\}$ is said to be a moving average process of order q (written MA(q)) if

$$X_t = \beta_0 Z_t + \beta_1 Z_{t-1} + \ldots + \beta_q Z_{t-q} \qquad (2.2.1)$$

where $\{\beta_i\}$ are constants. Additionally $E(X_t) = 0$ and $Var(X_t) = \sigma_Z^2 \sum_{i=0}^{q} \beta_i^2$, since the Z variables are independent. The Z variables are usually scaled so that $\beta_0 = 1$. A moving average process of any order can also be expressed by using the backward shift operator, denoted by B, which is defined by

$$B^j X_t = X_{t-j} \qquad \text{for all } j$$

Equation (2.2.1) may then be rewritten as

$$\begin{aligned} X_t &= (\beta_0 + \beta_1 B + ... + \beta_q B^q) Z_t \\ &= \theta(B) Z_t \end{aligned} \tag{2.2.2}$$

where $\theta(B)$ is a polynomial of order q in B. No restrictions on the $\{\beta_i\}$ are required for the process to be stationary, but restrictions are imposed to ensure that the process satisfies an invertibility condition, that is the roots of $\theta(B) = 0$ must lie outside the unit circle. This effectively means that the process can be written in the form of an autoregressive process, possibly of infinite order, whose coefficients form a convergent sum. It ensures that there is a unique moving average process for a given autocorrelation function.

2.2.2 Autoregressive Processes

Suppose that $\{Z_t\}$ is a purely random process with mean zero and variance σ_Z^2. Then a process $\{X_t\}$ is said to be an autoregressive process of order p (written AR(p)) if

$$X_t = \alpha_1 X_{t-1} + ... + \alpha_p X_{t-p} + Z_t \tag{2.2.3}$$

One can express an autoregressive process of finite order as a moving average process of infinite order. This may be done by successive substitution, or more easily, by utilizing the backward shift operator. Thus equation (2.2.3) may be written as

$$\phi(B) X_t = Z_t \tag{2.2.4}$$

where $\phi(B) = 1 - \alpha_1 B - ... - \alpha_p B^p$, so that $X_t = \phi(B)^{-1} Z_t$ is an infinite series. The stationarity condition for autoregressive processes requires the roots of $\phi(B)$=0 to

lies outside the unit circle. Autoregressive processes have been applied to many situations in which it is reasonable to assume that the present value of a time series depends linearly on the immediate past values together with a random error.

2.2.3 Mixed ARMA Models

A useful class of models for time series is formed by combining moving average and autoregressive processes. A mixed autoregressive/moving-average process containing p autoregressive terms and q moving average terms is said to be an ARMA process of order (p,q). It is given by

$$X_t = \alpha_1 X_{t-1} + ... + \alpha_p X_{t-p} + Z_t + \beta_1 Z_{t-1} + ... + \beta_q Z_{t-q} \qquad (2.2.5)$$

Using the backward shift operator B, equation (2.2.5) may be written in the form

$$\phi(B) X_t = \theta(B) Z_t \qquad (2.2.6)$$

where $\phi(B)$, $\theta(B)$ are polynomials of order p, q, defined in (2.2.4) and (2.2.2) respectively. The importance of ARMA processes lies in the fact that a stationary time series may adequately be modelled by an ARMA model, involving fewer parameters than a pure moving average or autoregressive process by itself.

2.2.4 ARIMA Models

Most time series, including the Kapenta data used in this thesis, are non-stationary. In order to fit a stationary model, it is necessary to remove non-stationary sources of variation. If the observed time series is non-stationary in the mean, then one can difference the series. When X_t is differenced/integrated of order d until the data is rendered stationary, denoted $\nabla^d X_t$, it can replace X_t in equation (2.2.6). Such a model is called an integrated model, because the stationary model that is fitted to the differenced data has to be summed or integrated to provide a model for the original non-stationary data. Writing

$$W_t = \nabla^d X_t = (1 - B)^d X_t$$

the general form of an ARIMA process is of the form

$$W_t = \alpha_1 W_{t-1} + ... + \alpha_p W_{t-p} + Z_t + ... + \beta_q Z_{t-q} \qquad (2.2.7)$$

By analogy with equation (2.2.6), one may write equation (2.2.7) in the form

$$\phi(B)W_t = \theta(B)Z_t$$

Thus one has a model for W_t describing the dth differences of X_t, which is said to be an ARIMA process of order (p,d,q).

Many time series contain a seasonal periodic component, which repeats every s observations. Box and Jenkins (1970) generalized the ARIMA model to deal with seasonality and defined a general multiplicative seasonal ARIMA model as

$$\phi_p(B)\varphi_p(B^s)W_t = \theta_q(B)\vartheta_Q(B^s)Z_t \qquad (2.2.8)$$

where B and B^s denote the backward shift operators of the series and seasonal respectively; $\phi_p,\varphi_p,\theta_q,\vartheta_Q$ are polynomials of order p, P, q, Q respectively; Z_t denotes a purely random process and

$$W_t = \nabla^d \nabla_s^D X_t$$

denotes the differenced series. If the integer D is not zero, then seasonal differencing is involved. The above model is called a SARIMA model of order (p,d,q) x $(P,D,Q)_s$.

2.3 Advantages and Disadvantages of ARIMA Models

Meyler et al. (1998) consider the main advantages and disadvantages of ARIMA models. Unlike other methods, ARIMA models do not assume knowledge of any underlying economic model or structural relationships. It is assumed that past values of the series plus previous error terms contain information for the purposes of forecasting. These models have proven themselves to be relatively robust in terms of short-run forecasting ability, and are thought by some to outperform more sophisticated structural models in this regard (Stockton and Glassman; 1987 and Litterman; 1986). One disadvantage of ARIMA models is that some of the identification techniques are subjective and the reliability of the chosen model can depend on the skill and expertise of the modeller. Another disadvantage is that ARIMA models are not embedded within any underlying theoretical model or structural relationships. The economic significance of the chosen model is

therefore not always clear. Lastly, ARIMA models are backward looking, and are poor at predicting turning points, unless this represents a return to a long-run equilibrium.

2.4 State Space Models: Structural Time Series Models

The next methodology considered for modelling Kapenta catch data is that of structural time series models, a specific kind of linear Gaussian state space model. This section will briefly introduce structural time series models, but a more detailed investigation is provided in Chapter 3 of this thesis.

State space models assume that the development over time of a system is determined by an unobserved series of vectors $\alpha_1,...,\alpha_n$ (the state), with which are associated a series of observations $y_1,...,y_n$. The relationship between the α_t's and y_t's is specified by the state space model. The state space model, based on the state vector, α_t, and the observational vector, y_t, is such that

$$\alpha_{t+1} = f(\alpha_1,...,\alpha_t,w_t)$$
$$y_t = g(\alpha_t,v_t)$$

The functions f and g define the state evolution and observation processes respectively, and w_t and v_t represent the state evolution and observation noise respectively. State space models may be linear or non-linear, but the models dealt with in this thesis are strictly linear. More specifically, this thesis deals with linear Gaussian state space models or dynamic linear models, and thus by definition, w_t and v_t are Gaussian distributed (Mergner, 2009). Linear Gaussian state space models are, according to Harvey (1989), appropriate for many datasets including those from economics, sociology, operational research, geography, meteorology and engineering. However, most applications in the literature are economic or financially based. Early applications of state space models and the Kalman filter in economics and finance include Fama and Gibbons (1982) who modelled the unobserved ex-ante real interest rate as a state variable that follows an AR(1) process. Clark (1987) used an unobserved-components model to decompose gross national product (GNP) data into two independent components of trend and cycle.

Additional works related to economics include that of Stock and Watson (1991) and Hamilton (1994).

A structural time series model is a linear Gaussian state space model constructed from trend, seasonal, cyclical and irregular components; additionally they may be extended to include regression terms. Each of these components has their own model, and each of them can be directly interpreted. The state of the system represents these various unobserved components. There are no stationarity restrictions when building structural time series models and the models achieve great flexibility by allowing any of the component coefficients to change over time. The estimate of the unobservable state can be updated by means of a filtering procedure as new observations become available, while smoothing algorithms give the best estimate of the state at any point in time based on all the observations.

2.5 Advantages and Disadvantages of Structural Time Series Models

Harvey (1997) and Jalles (2009) outlined the reasons for their belief in the superiority of structural time series models over ARIMA models. Structural models have several advantages when compared with ARIMA models. ARIMA models can be formulated in state space form and many structural models admit ARIMA representation. For time series with a simple underlying structure both formulations are equivalent to each other; however when the structure is more complex then the differences between the approaches become more evident. In a structural time series model each component, such as the trend, cycle, or seasonal changes, is explicitly formulated and therefore it is possible to get specific information about them. This allows structural models easier interpretive value as observations are modelled directly. This is perhaps the main advantage of structural models over ARIMA models, in which the trend and seasonal components are eliminated by applying convenient differences to the original series before carrying out the analysis. ARIMA methodology constitutes itself as a kind of black box in which the adopted model depends entirely on the data,

without a prior analysis of the structure underlying the generating system. Structural models are more transparent as they allow checking if the predicted behavior by the model for each component corresponds to what is expected from the data. The requirement of stationarity in the Box and Jenkins' (1970) approach implies differencing the series, but one is not always able to decide on the right integration order. In fact, basic tools to identify ARIMA models, i.e., autocorrelation functions, are merely guiding tools and very often do not allow opting for a unique model. Using a sample autocorrelation function does not allow for complex models in smaller samples to be identified. ARIMA models were typically developed to identify simple models in large samples.

Structural Time Series models are also more flexible. They eliminate the problem of restrictive deterministic trends and increase flexibility by letting the slope and level parameters change over time. The recursive nature of the model and the computational techniques used for its analysis allow the direct incorporation of known breaks in the system structure over time. Box and Jenkins ARIMA models, however, are based on the assumption that differenced series are stationary which immediately makes these models more restrictive. With the structural approach forecasting is relatively straightforward and missing observations are easier to treat. Brockwell and Davis (1991) consider that state space representation and recursive equations, which characterize the Kalman filter, are ideal to analyse series with missing observations. Observations corresponding to multivariate series can be manipulated by direct extension of the univariate structural formulation. Moreover, the Markovian nature of state space models allows the necessary computations to be implemented in a recursive way; this, in fact, allows manipulation of high dimensional models without an overwhelming increase of the computational task. Structural time series models also allow for the inclusion of regression terms, which may be time varying. These models thus combine the flexibility of a time series model, with the interpretation of a regression.

2.6 Chapter Conclusion

This thesis aims to understand all trend, seasonal, cyclical and covariate components that affect Kapenta catch in Lake Kariba. A modelling methodology into which explanatory variables, seasonal and cyclical components can be included is thus necessary. Although ARIMA models take seasonal or cyclical behaviour into consideration by removing them prior to modelling, they do not explicitly account for these components in the model. Explanatory variables are modelled for stationary data, usually differenced data, and the effect of these variables on the original data is difficult to interpret. Structural time series models directly model seasonal and cyclical components, and easily allow for the inclusion of regression terms without rendering the data stationary beforehand, and can be dynamically included. This allows for interpretable models. ARIMA models have been noted in Section 2.3 to outperform more sophisticated structural models in terms of short-run forecasting ability. However, since this thesis intends to take a more long-term, backward-looking view of components affecting Kapenta catch, this is not applicable. Additionally, as discussed in Section 2.5, structural time series models provide numerous advantages over ARIMA models and these advantages appear more relevant to the Kapenta catch data this thesis intends to model. Structural time series models are more flexible and handle missing values well. These are appropriate for the highly variable Kapenta catch data that contains missing values. For the reasons discussed above, this thesis makes use of structural time series models.

Chapter 3

Linear Gaussian State Space and Structural Time Series Models

3.1 Introduction

The previous chapter provided an overview of various time series models suitable for non-stationary data, including a brief mention of structural time series models used in this study. Structural time series models are a particular kind of linear Gaussian state space model. This chapter starts with a discussion on the general theory of linear Gaussian state space models, after which the particular features of structural time series models are discussed. The following features of linear Gaussian state space models are addressed: the formulation of these models in Section 3.2, the Kalman filtering and smoothing processes and how they are initialised in Section 3.3, the estimation of unknown parameters by maximum likelihood in Section 3.5, as well as model diagnostics and goodness-of-fit assessments in Section 3.6. The particular features of structural time series models, where trend, seasonal, cyclical and regression components are modelled separately are then described in Section 3.7. The formulation of each of these components is provided, and the state space equivalent form of these models described. Additionally, the multivariate form of structural time series models is

considered. This chapter also discusses the estimation of unknown parameters from a Bayesian perspective in Section 3.8 and concludes with a discussion on the implementation of structural time series models in R, using the `dlm` package. Detailed treatments of state space models can be found in Harvey (1989) and Harvey and Shephard (1993) among others. If not indicated otherwise Durbin and Koopman (2001) and Mergner (2009) serve as the standard reference for this chapter.

3.2 Linear Gaussian State Space Models

A general linear Gaussian state space model can be written as

$$\begin{aligned} y_t &= Z\alpha_t + \varepsilon_t, & \varepsilon_t &\sim N(0,H) \\ \alpha_{t+1} &= T\alpha_t + \eta_t, & \eta_t &\sim N(0,Q) \end{aligned} \tag{3.2.1}$$

where y_t is the $N \times 1$ multivariate time series of observations and α_t the $m \times 1$ unobserved state vector, at each date t, for t=1,...,n. A state space model is in principle any model that includes an observation process and a state process.

The development of the system over time is determined by α_t according to the second equation in (3.2.1). Equation (3.2.1) shows the deterministic parameter matrices T and Z, of dimension $m \times m$ and $N \times m$ respectively. Unobserved structural components such as trend, seasonal and cycle may be modelled by an appropriate definition of Z and α_t. The $N \times 1$ and $r \times 1$ error terms ε_t and η_t are assumed to be serially uncorrelated and normally distributed, with zero mean and positive definite covariance matrices Q and H, of dimensions $r \times r$ and $N \times N$ respectively. Additionally these disturbances are further assumed to be uncorrelated with each other at all lags and independent of the initial state vector α_1. The matrices Z, T, H and Q are called system matrices. They are assumed known and deterministic unless otherwise stated. The initial $m \times 1$ state vector α_1 is assumed to follow a $N(a_1,P_1)$ distribution and does so independently of the error terms ε_t and η_t, where a_1 and P_1, of dimensions $m \times 1$ and $m \times m$, are assumed known for now. The more common case on how to proceed in the absence of

knowledge of a_1 and P_1 is discussed in Section 3.3.5 under the initialisation of the state vector.

3.3 Filtering and Smoothing

Once a model is in state space form, the Kalman filter computes the optimal forecasts of the mean and covariance matrix of the normally distributed state vector α_{t+1} based on information through to time *t.* Filtering aims to update the system as each observation y_t becomes available. Smoothing enables one to base estimates of quantities of interest on the entire sample. Smoothing is performed while proceeding backwards through observations using what is known as the state smoothing recursions, while filtering is done by moving forward through the observations by applying the Kalman filter.

3.3.1 Filtering

The objective of filtering is to update our knowledge of the state vector as each new observation becomes available, i.e. to obtain the conditional distribution of α_{t+1} given Y_t for $t=1,...,n$, where Y_t denotes $\{y_1,...,y_t\}$. The Kalman filter uses the state estimate from the previous time step to produce an estimate of the state at the current time step. This predicted state estimate does not include observation information from the current time step. The current prediction is then combined with current observation information to refine the state estimate. These two steps alternate, but if an observation is unavailable for some reason, the update may be skipped and multiple prediction steps performed.

Kalman Filter Derivation:

For linear Gaussian state space models, all distributions and conditional distributions are normally distributed. It is assumed that α_t given Y_{t-1} is $N(a_t,P_t)$, and for now a_1 and P_1 are assumed known. The required conditional distribution of α_{t+1} can be characterized by its mean $a_{t+1} = E(\alpha_{t+1} | Y_t)$ and covariance $P_{t+1} = \text{var}(\alpha_{t+1} | Y_t)$. The mean of the conditional distribution of α_{t+1} represents an

optimal estimator of the state vector at time t+1; it minimizes the mean square error matrix for all α_{t+1}.

Due to the assumption that α_t given Y_{t-1} is $N(a_t,P_t)$, it can be shown that a_{t+1} and P_{t+1} are calculated through recursive algorithms from a_t and P_t resulting in the following set of equations that constitute the Kalman filter:

$$\begin{aligned} v_t &= y_t - Za_t, & F_t &= ZP_tZ' + H, \\ K_t &= TP_tZ'F_t^{-1}, & L_t &= T - K_tZ, \\ a_{t+1} &= Ta_t + K_tv_t, & P_{t+1} &= TP_tL_t' + Q, \end{aligned} \qquad t = 1,...,n \qquad (3.3.1)$$

The $N \times 1$ vector v_t is defined as the one step ahead forecast error of y_t given Y_t. It is assumed that the $N \times N$ matrix F_t, defined as the variance of v_t, is non-singular. The equations in (3.3.1) can be reformulated to generate a recursion that incorporates the computation of the state vector estimator $E(\alpha_t | Y_t)$ and its associated error variance matrix, denoted by $a_{t|t}$ and $P_{t|t}$ respectively. This reformulation looks as such:

$$\begin{aligned} v_t &= y_t - Za_t, & F_t &= ZP_tZ' + H, \\ & & M_t &= P_tZ', \\ a_{t|t} &= a_t + M_tF_t^{-1}v_t, & P_{t|t} &= P_t - M_tF_t^{-1}M_t', \\ a_{t+1} &= Ta_{t|t}, & P_{t+1} &= TP_{t|t}T' + Q, \end{aligned} \qquad t=1,\ldots,n \qquad (3.3.2)$$

One Step Ahead Forecast Errors:

One-step ahead forecast errors (v_t), defined in (3.3.1), measure the difference between observations and the corresponding one-step-ahead predictions generated by the Kalman filter, where $\text{var}(v_t) = F_t$. These forecast errors are independent of each other and are used in analyzing model fit and diagnostic measures. These measures are presented in Section 3.6. Additionally, one-step ahead forecast errors are used in calculating the log-likelihood function of linear Gaussian state space models. This is observed in Section 3.5, when unknown model parameters are estimated using maximum likelihood.

3.3.2 State Smoothing

The filtered estimate of α_t only takes into account the 'past' information relative to α_t. By incorporating the 'future' observations relative to α_t, a more refined state estimate can be obtained. In other words, using *y* to denote the stacked vector $(y_1^{'},...,y_n^{'})^{'}$, smoothing considers the estimation of α_t given the entire time series *y*. State smoothing was introduced by de Jong (1988) and Kohn and Ansley (1989). All system distributions are normal and α_t is estimated by it's conditional mean $\hat{\alpha}_t = E(\alpha_t \mid y)$. Additionally, the error variance matrix $V_t = Var(\alpha_t - \hat{\alpha}_t)$ $= Var(\alpha_t \mid y)$ is also calculated for $t = 1,...,n$. Here $\hat{\alpha}_t$ is known as the smoothed state and V_t as the smoothed state variance.

Given the assumption that $\alpha_1 \sim N(a_1,P_1)$, where a_1 and P_1 are assumed known, the smoother recursively estimates $\hat{\alpha}_t$ and V_t. The smoothed state estimator is also an optimal estimator minimising the mean square error matrix.

It can be shown that the smoothed state vector $\hat{\alpha}_t$ can be calculated by the following backward recursion:

$$r_{t-1} = Z^{'}F_t^{-1}v_t + L_t^{'}r_t, \qquad \hat{\alpha}_t = a_t + P_t r_{t-1}, \; t = n,...,1 \qquad (3.3.3)$$

This is initialized with $r_n = 0$ and is an efficient algorithm for calculating $\hat{\alpha}_1,...,\hat{\alpha}_n$. The $m \times 1$ vector r_{t-1} is a weighted sum of future innovations.

It can also be shown that the smoothed state variance matrix V_t can be calculated recursively by the following:

$$N_{t-1} = Z^{'}F_t^{-1}Z + L_t^{'}N_t L_t, \quad V_t = P_t - P_t N_{t-1} P_t, \quad t=n,...,1 \qquad (3.3.4)$$

This is initialized with $N_n = 0$. It can also be shown that the $m \times m$ matrix $N_t = Var(r_t)$.

The above results (3.3.3) and (3.3.4) constitute the state smoothing recursion.

3.3.3 Covariances for Smoothed Estimates

States may be compared at different points in time using smoothed estimates, and the significance of this difference can be assessed using covariances for smoothed

estimates, $C_{s,t|n}$. For the state space model in (3.2.1), the smoothing algorithm in (3.3.3) and (3.3.4) recursively calculates the conditional expectations $\hat{\alpha}_t$ and associated conditional covariance matrices V_t, conditioning on y. Additionally, seen in (3.3.2), the Kalman filter recursively calculates the conditional expectations $a_{t|t} = E(\alpha_t \mid y_1,...,y_t)$ and associated conditional covariance matrices $P_{t|t} = \text{cov}(\alpha_t - a_{t|t})$. A recursive formula for the covariance matrices

$$C_{s,t|n} = (\alpha_s - \hat{\alpha}_s, \alpha_t - \hat{\alpha}_t) \qquad (s, t < \infty)$$

is seen as

$$C_{s,t|n} = A_s C_{s+1,t|n} \qquad (s < \min(t.n))$$

where $A_s = P_{s|s} T' F_s^{-1}$ $(s < n)$ and, for $s \geq 1$, $P_{s|s}$ and F_s are the mean squared error matrices of α_s and α_{s+1} given $y_1,...,y_s$. Cross-covariance matrices are thus expressed in terms of the matrices V_t, calculated using the Kalman filter and smoother. The derivation of this result is seen in De Jong and Mackinnon (1988).

3.3.4 Missing Observations

Missing values are easily dealt with using state space models and for a missing set of observations, the original filtering and smoothing recursions can be used. When a set of observations y_t for $t = \tau,...,\tau^* - 1$ is missing, the vector v_t and matrix K_t of the Kalman filter in (3.3.1) are set to zero for these values, reducing the Kalman filter updates for a_{t+1} and P_{t+1} to

$$a_{t+1} = Ta_t, \qquad P_{t+1} = TP_tT' + Q, \qquad t = \tau,...,\tau^* - 1$$

and the backward smoothing recursions for $r_{t-1}$and N_{t-1} to

$$r_{t-1} = T' r_t, \qquad N_{t-1} = T' N_t T, \qquad t = \tau^* - 1,...,\tau$$

Other relevant equations remain the same.

3.3.5 Initialisation of the Filter and Smoother

In most applications at least some of the elements of a_1 and P_1 are unknown. Initialisation is the process of starting up the Kalman filter when this is the case.

Consider the general case where some of the elements α_1 have a known joint distribution while other elements are completely unknown.

A general model for the initial state vector α_1 is given by

$$\alpha_1 = a + D\delta + \eta_0, \qquad \eta_0 \sim N(0,Q),$$

where the $m \times 1$ vector a is treated as a zero vector whenever none of the elements of α_1 are known constants. The $m \times q$ matrix D is a fixed and known selection matrix consisting of columns of the identity matrix, and the covariance matrix Q is assumed positive definite and known. The $1 \times q$ vector δ, treated as a random variable with infinite variance and following an $N(0,\kappa I_q)$ as $\kappa \to \infty$, is said to be diffuse. Initialisation on the Kalman filter when elements of α_1 are diffuse is called 'diffuse initialisation of the filter.' One begins by considering the Kalman filter with initial conditions $a_1 = E(\alpha_1) = a$ and $P_1 = Var(\alpha_1)$ where

$$P_1 = \kappa P_\infty + Q$$

and $P_\infty = DD'$, a diagonal matrix (since D contains column of the identity matrix) with q diagonal elements equal to one and the other elements equal to zero. With some elements of α_1 being diffuse, modifications to the Kalman filter are required. Modifications are needed in cases where P_∞ is a nonzero matrix as no real value can represent $\kappa \to \infty$. The R package `dlm` (Petris, 2010), used in this thesis to model linear Gaussian state space models (see §3.9 for details), replaces κ by a large but finite numerical value and this enables the use of the standard Kalman filter, as suggested by Harvey and Phillips (1979). Unless genuine information on a_1 and P_1 is available, a diffuse prior with $a = 0$, $Q=0$ and $P_\infty = I$ will be used such that $\alpha_1 \sim N(0,\kappa I)$. The value κ is set equal to 10^6 and then multiplied by the maximum diagonal value of the residual covariance to adjust for scale:

$$\kappa = 10^6 \times \max\left\{1, \begin{bmatrix} Q & 0 \\ 0 & H \end{bmatrix}\right\}$$

Alternative treatments for exact initialization were considered. One such method was developed for $\kappa \to \infty$ directly, and can be viewed in Ansley and Kohn

(1985), de Jong (1988) and Koopman (1997). Another such method by Rosenburg (1973) considers α_1 to be an unknown constant that can be estimated from the first observation y_1 by maximum likelihood. .

Initialization for the multivariate case can be complicated as the inverse matrix F_t^{-1} does not have a simple general expansion in powers of κ^{-1} for the first few terms of the series. This is due to the fact that in very specific situations the part of F_t associated with P_∞ can be singular with varying rank. For univariate series this problem does not exist since F_t is a scalar. Since this problem is easily dealt with in the univariate case, Durbin and Koopman (2001) describes a method in which the components of a multivariate series are brought into the analysis one at a time, essentially converting the multivariate series into a univariate one. For a detailed discussion of this methodology whereby a multivariate series is treated as a univariate one, refer to Durbin and Koopman (2001, §6.4).

3.4 Estimation of Regression Coefficients

General state space models can be extended to allow for the incorporation of explanatory variables into the model. To accomplish this extension, the first equation in model (3.2.1) is replaced by

$$y_t = Z\alpha_t + X_t\beta_t + \varepsilon_t \tag{3.4.1}$$

where $X_t = (x_{1,t},...,x_{k,t})$ is a matrix containing rows of the *k* explanatory variables and β_t is a $k \times 1$ vector of unknown regression coefficients. The vector of unknown regression coefficients is denoted to be time-varying but can be modelled as constant through replacing β_t by β for t=1,...,n in equations (3.4.1), (3.4.2) and (3.4.3). The inclusion of regression effects can be dealt with by including the coefficient vector in the state vector. More specifically, in state space form model (3.4.1) is formulated as

$$\begin{aligned} y_t &= \begin{bmatrix} Z & X_t \end{bmatrix} \begin{bmatrix} \alpha_t \\ \beta_t \end{bmatrix} + \varepsilon_t, & \varepsilon_t &\sim N(0,H) \\ \begin{pmatrix} \alpha_{t+1} \\ \beta_{t+1} \end{pmatrix} &= \begin{bmatrix} T & 0 \\ 0 & I_k \end{bmatrix} \begin{pmatrix} \alpha_t \\ \beta_t \end{pmatrix} + \eta_t, & \eta_t &\sim N(0,Q^{\Diamond}) \end{aligned} \tag{3.4.2}$$

for $t = 1, \ldots, n$. The covariance matrix of the state errors is defined as

$$Q^{\Diamond} = \begin{bmatrix} Q & 0 \\ 0 & Q^{\beta} \end{bmatrix}$$

The structure of the $k \times k$ block matrix Q^{β} determines the nature of the regression coefficients. For fixed coefficients Q^{β} is set to zero, and time varying coefficients is set to nonzero. Note that when regression components are added to a state space model, the Z matrix in (3.2.1) essentially becomes time varying, as it is described by the matrix $\begin{bmatrix} Z & X_t \end{bmatrix}$, which can alternatively be denoted by Z_t.

The initial state vector, β_1 (or merely β when regression coefficients are constant) can be taken as diffuse or fixed. In the diffuse case the model for the initial state vector is

$$\begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix} \sim N\left(\begin{pmatrix} a \\ 0 \end{pmatrix}, \kappa \begin{bmatrix} P_{\infty} & 0 \\ 0 & I_k \end{bmatrix} + \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} \right) \tag{3.4.3}$$

with κ defined in Section 3.3.4. The Kalman filter with initialisation methods discussed in Section 3.3.4 can be extended to this enlarged state space model and regression coefficients can be estimated by the maximum likelihood procedures to be discussed in Section 3.6. Since this is merely a special case of the model introduced in (3.2.1), it can be routinely handled by the standard Kalman filter and smoother.

3.5 Maximum Likelihood Estimation

So far system matrices have been assumed known. Generally, however, they depend at least partially on a vector of unknown parameters, denoted ψ. These unknown parameters are generally the observation and state evolution variances. In this section the vector of unknown parameters are estimated by maximum likelihood. This section introduces the loglikelihood function for the linear Gaussian state space model, where initial conditions are known and where initialization occurs with a diffuse prior. Additionally, a brief overview is provided on the maximization of the likelihood function.

3.5.1 The Loglikelihood Function

To estimate a model by maximum likelihood, the model has to be parametric and fully specified through the joint probability function. For *n* sets of observations $y_1,...,y_n$, which are assumed to be identically and independently distributed, the joint density is given by the product of the individual densities, denoted by $p(\bullet)$:

$$L(y,\psi) = p(y_1,...,y_n) = \prod_{t=1}^{n} p(y_t)$$

where $L(y,\psi)$ equals the joint probability density function. When the joint density is evaluated at a given data set, $L(y,\psi)$ is referred to as the likelihood function. It is generally easier to work with the logarithm of the likelihood function defined by

$$\log L(y,\psi) = \sum_{t=1}^{n} \log p(y_t)$$

The above two functions are denoted by $L(y)$ and $\log L(y)$ respectively, and the maximum likelihood estimates of the parameters ($\hat{\psi}$) are found by maximizing the loglikelihood with respect to ψ.

3.5.2 Likelihood when Initial Conditions are Known

In this section it is assumed that the initial state vector density $N(a_1,P_1)$ has a_1 and P_1 known. Due to the fact that the observations are not generally independent, especially in time series, the probability density functions in Section 3.5.1 are replaced by their conditional distribution. The likelihood is then defined as follows

$$L(y) = p(y_1,...,y_n) = p(y_1)\prod_{t=2}^{n} p(y_t \mid Y_{t-1})$$

where $Y_t = \{y_1,...,y_t\}$. In practice one generally uses the loglikelihood defined as

$$\log L(y) = \sum_{t=1}^{n} \log p(y_t \mid Y_{t-1}) \qquad (3.5.1)$$

where $p(y_1 \mid Y_0) = p(y_1)$. For the model defined in (3.2.1), $E(y_t \mid Y_{t-1}) = Za_t$. Putting $v_t = y_t - Za_t$, $F_t = Var(y_t \mid Y_{t-1})$ and substituting $N(Z,a_t,F_t)$ for $p(y_t \mid Y_{t-1})$ in (3.5.1) one obtains

$$\log L(y) = -\frac{nN}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{n}\log|F_t| - \frac{1}{2}\sum_{t=1}^{n}v_t' F_t^{-1} v_t \qquad (3.5.2)$$

The quantities v_t and F_t are calculated by the Kalman filter in (3.3.1), and therefore $\log L(y)$ is easily computed from the Kalman filter output. Note that for the purposes of this thesis, when regression components are added to the state space model, the *Z* matrix becomes time varying and must be denoted Z_t.

3.5.3 Diffuse Loglikelihood

Now consider the case where some elements of α_1 are diffuse. When the Kalman filter is intialised with *d* diffuse elements in the state vector, the first *d* innovations and their corresponding variances are excluded from the loglikelihood in (3.5.2). The joint density of $y_{d+1},...,y_n$ conditional on $y_1,...,y_d$ is

$$\log L(y) = -\frac{(n-d)N}{2}\log 2\pi - \frac{1}{2}\sum_{t=d+1}^{n}\log|F_t| - \frac{1}{2}\sum_{t=d+1}^{n}v_t' F_t^{-1} v_t \qquad (3.5.3)$$

Note that equations (3.5.2) and (3.5.3) apply for univariate time series in the same way.

3.5.4 Parameter Estimation by Numerical Maximization

Given sample observations the loglikelihood is maximised by means of numerical maximisation; the process finds the value of $\hat{\psi}$ that maximises the loglikelihood. Generally, different starting values are chosen and the algorithm chooses a direction in which to search based on derivatives of the loglikelihood function. If ψ is sufficiently close to the maximum of the loglikelihood, the algorithm stops, otherwise the search continues. Different algorithms differ with regards to search direction, time step and stopping rules. Many numerical maximisation algorithms are based on Newton's method, where the gradient or score vector determines the direction of the search and the Hessian matrix determines the step size. The numerical maximisation procedure used for this thesis is known as the "L-BFGS-

B" method of Byrd et al. (1995). This is a limited memory algorithm for solving large nonlinear optimisation problems subject to simple bounds on the variables.

3.6 Model Diagnostics and Goodness-of-Fit

All models are first assessed to check if initial model assumptions made are upheld (diagnostics), and are secondly assessed on how well they fit the Kapenta data. This section discusses the methods used in this thesis to assess model diagnostics and goodness-of-fit. In addition to Durbin and Koopman (2001), a reference for this section is Harvey (1989), where a more detailed discussion of these measures can be found. Harvey (1989) addresses the model diagnostics and goodness-of-fit measures from a univariate perspective only, and thus the discussion seen in this thesis is limited to the univariate case.

3.6.1 Diagnostics

In a well-specified model, the residuals must be independent and normally distributed. Here the various graphical procedures and tests for assessing these conditions are discussed for state space models. However, some notation is first addressed:

The one-step ahead forecast errors are defined in Section 3.3.1 and are defined as follows for univariate time series:

$$v_t = y_t - za_t$$

where the vector z is the univariate equivalent of the deterministic parameter matrix Z and a_t is the conditional mean of the state vector. The matrix F_t, defined as the $\text{var}(v_t)$, is replaced by the following scalar for univariate time series

$$f_t = z' P_r z + h$$

where h and z are the univariate equivalents of the multivariate matrices H and Z respectively and P_t is the covariance of the state vector. Note that when regression components are added to the state space model, as in the multivariate case, the

vector z is time varying and should rather be denoted z_t. The standardized innovations for univariate time series are defined as

$$\tilde{v}_t = \frac{v_t}{f_t^{1/2}}, \qquad t=d+1,\ldots,n$$

where d is the number of diffuse elements in the state vector.

The following model assumptions are assessed as follows:

Serial Correlation: The residual sample autocorrelation at lag τ is defined by

$$r_v(\tau) = \frac{\sum_{t=s+1+\tau}^{n}(\tilde{v}_t - \bar{\tilde{v}})(\tilde{v}_{t-\tau} - \bar{\tilde{v}})}{\sum_{t=s+1}^{n}(\tilde{v}_t - \bar{\tilde{v}})^2}$$

where $\bar{\tilde{v}}$ is the mean of the standardised innovations. The resulting correlogram gives an indication of any serial correlation; if less than approximately 5% of the lags show significant correlation, correlation is not deemed problematic. Alternatively, a joint test of significance for the first P residual autocorrelations is given by the Box-Ljung test statistic

$$Q^* = (n-d)(n-d+2)\sum_{\tau=1}^{P}(n-d-\tau)^{-1}r_v^2(\tau)$$

where Q^* is asymptotically χ^2 with P-s+1 degrees of freedom; where s represents the number of non-zero parameters in the model, n the number of observations in the data and $r_v^2(\tau)$ the square of the residual sample autocorrelation at lag τ. Values considered for P were 40 or 50 lags. However, correlograms were used in the present study to measure the presence of serial correlations instead of the Box-Ljung test statistic, as the single value result produced from the Box-Ljung tests were all highly significant, and made it difficult to distinguish between models.

Homoscedasticity: Homoscedasticity can be checked graphically through a plot of the standardized innovations ($\tilde{v}_t$) over time, where the innovations must appear constant over time. Alternatively, a diagnostic test for homoscedasticity can be constructed from the residuals. Suppose that h is the nearest integer to $(n-d)/3$. The test statistic is seen as

$$H(h) = \sum_{t=n-h+1}^{n} \tilde{v}_t^2 \Big/ \sum_{t=s+1}^{s+1+h} \tilde{v}_t^2$$

This statistic can be tested against an $F(h,h)$ distribution. This thesis will check the assumption of homoscedasticity graphically, as similarly to the Box-Ljung statistic, poor and non-indicative results were produced, showing consistently significant heteroscedasticity.

Normality: Normal quartile plots, referred to as QQ –plots, of the standardized innovations can be used to detect any deviations from normality. This is a plot of percentiles of a normal distribution against the corresponding percentiles of the observed data. If the observations follow approximately a normal distribution, the resulting plot should be roughly a straight line with a positive slope.

3.6.2 Goodness of Fit

In order to assess the fit of any single model and to make comparisons between different models, selected measures of fit are assessed as follows:

Prediction Error Variance

The vector of unknown parameters ψ can be reparameterised such that $\psi = \left[\psi_*^{'} \quad \sigma_*^2\right]^{'}$, where σ_*^2 is a positive scalar to which the variances of the disturbance terms are proportional and where the vector ψ_* contains one parameter less than ψ. The prediction error variance (P.E.V) is defined as

$$\sigma^2 = \sigma_*^2 \bar{f}$$

where $\bar{f}$ is the steady state value of f_t, namely

$$\bar{f} = \lim_{t \to \infty} f_t$$

The prediction error variance can be approximated as follows:

$$\tilde{\sigma}^2 = \frac{\bar{f}}{n-d} \sum_{t=d+1}^{n} \tilde{v}_t^2 = \frac{1}{n-d} \sum_{t=d+1}^{n} \frac{v_t^2}{f_t} \bar{f} \approx \frac{1}{n-d} \sum_{t=d+1}^{n} v_t^2$$

Coefficient of determination: This is a measure of fit similar to that of the traditional R^2 used in regression analysis, but is specifically for the use of state space models. The residual sum of squares for a univariate time series model is defined as

$$SSE = \bar{f} \sum_t \tilde{v}_t^2 = (n-d)\tilde{\sigma}^2$$

where $\tilde{\sigma}^2$ is the prediction error variance defined above, d is the number of diffuse elements in the state vector and n the number of observations in the data. The coefficient of determination, R_D^2, is obtained in a similar manner to the traditional R^2, but adapted by replacing the observations by their first differences

$$R_D^2 = 1 - SSE / \sum_{t=2}^{n} (\Delta y_t - \overline{\Delta y})^2$$

where $\overline{\Delta y}$ is the mean of the first differences. This value can be calculated as negative for models with very poor model fits.

3.6.3 Model Comparison

AIC: For rival models containing different numbers of parameters, comparison can be made on the basis of the Akaike information criterion. Taking the loglikelihood as calculated in Section 3.5 as $\log L(y|\psi)$, or $\log L_d(y|\psi)$ in the diffuse case, the AIC is

$$AIC = -2\log L(y|\psi) + 2w \qquad (3.6.1)$$

where $w=q+d$; d is the number of diffuse elements in the state vector and q is the total number of elements in the state vector. Section 3.9.7 demonstrates an example where the q and d parameters are specified.

3.7 Structural Time Series Models

3.7.1 Univariate models

This paper considers a specific kind of linear Gaussian state space model, namely structural time series models. These models decompose the observations into

trend, seasonal, cyclical, and regression components, plus an error term that can each be directly interpreted. Each of these components can be modelled through a random walk process in order to capture time dependence. The basic structural time series model is defined as follows:

$$y_t = \mu_t + \gamma_t + \varepsilon_t, \qquad t = 1,...,n \tag{3.7.1}$$

where μ_t represents the trend, γ_t the seasonal and ε_t the irregular components. The observations are denoted by y_t. Each of these components can represent scalars or vectors.

The basic structural time series model can be augmented to include a cyclical component c_t, as well as explanatory variables x_{jt} each with regression coefficient β_j:

$$y_t = \mu_t + \gamma_t + c_t + \sum_{j=1}^{k} \beta_j x_{jt} + \varepsilon_t \qquad t = 1,...,n \tag{3.7.2}$$

3.7.1.1 The Trend Component:

Harvey (2000) defines a trend as the part of a series which, when extrapolated, gives the clearest indication of the future long-term movements in the series (not including any of the seasonal, cyclical or covariate effects in the series). One can model the trend, μ_t, in both (3.7.1) and (3.7.2) in various ways:

(1) *The Local Level Model:* The trend is a random walk, and is modelled as follows:

$$\begin{aligned} y_t &= \mu_t + \varepsilon_t, \qquad \varepsilon_t \sim N(0,\sigma_\varepsilon^2), \qquad t=1,...,n \\ \mu_t &= \mu_{t-1} + \xi_t, \qquad \xi_t \sim N(0,\sigma_\xi^2) \end{aligned} \tag{3.7.3}$$

where the irregular and level disturbances, ε_t and ξ_t respectively, are mutually independent and normally distributed.

(2) *The Local Linear Trend Model:* This models the trend component with a stochastic slope l_t, which itself follows a random walk. Thus

$$
\begin{aligned}
y_t &= \mu_t + \varepsilon_t, & \varepsilon_t &\sim N(0,\sigma_\varepsilon^2) \\
\mu_t &= \mu_{t-1} + l_{t-1} + \xi_t, & \xi_t &\sim N(0,\sigma_\xi^2) \\
l_t &= l_{t-1} + \zeta_t, & \zeta_t &\sim N(0,\sigma_\zeta^2)
\end{aligned} \tag{3.7.4}
$$

where the irregular, level and slope disturbances ε_t, ξ_t and ζ_t, respectively, are mutually independent.

(3) *Deterministic Model:* This model contains both variances σ_ξ^2 and σ_ζ^2 in (3.7.4) as zero, and the trend of this model is deterministic, modelled as such

$$\mu_t = \mu_0 + lt, \qquad t = 1,..,T \tag{3.7.5}$$

(4) *Random Walk With Drift Model:* This model contains the variance σ_ξ^2 in (3.7.4) as zero only; thus the slope is fixed and the trend reduces to a random walk with drift

$$\mu_t = \mu_{t-1} + l + \xi_t \tag{3.7.6}$$

(5) *Smooth Trend Model:* This models allows the variance σ_ξ^2 to be positive, but sets σ_ζ^2 to zero, and results in an integrated random walk trend, which when estimated tends to be relatively smooth. It is desirable to have a smooth trend, although a smooth trend should not be imposed regardless of fit.

3.7.1.2 The Seasonal and Cyclical Components

There are two approaches to modelling the seasonal component γ_t, namely a seasonal factor model and a Fourier-form seasonal model.

(1) *Seasonal Factor Model*: suppose there are *s* 'months' per 'year.' If the seasonal pattern is constant, the seasonal values for months 1 to *s* can be modelled by constants $\gamma_1^*,...,\gamma_s^*$ where $\sum_{j=1}^{s}\gamma_j^* = 0$. For the *j*th month in year *i* one has $\gamma_t = \gamma_j^*$

where $t = s(i-1)+j$ for $i = 1,2,...$ and $j = 1,..,s$. It follows that $\sum_{j=0}^{s-1}\gamma_{t+1-j} = 0$ so

$\gamma_{t+1} = -\sum_{j=1}^{s-1}\gamma_{t+1-j}$ with $t = s\text{-}1, s, \ldots, 1$.

A time varying seasonal component is achieved by adding an error term ω_t to the relation

$$\gamma_{t+1} = -\sum_{j=1}^{s-1}\gamma_{t+1-j} + \omega_t, \quad \omega_t \sim N(0,\sigma_\omega^2) \quad \text{for } t=1,\ldots,n$$

(2) *Fourier-Form Model:* This method is more complicated than that of seasonal factors (Petris et al., 2007), but allows for a more parsimonious representation of real world seasonal phenomena and is the methodology used in this thesis to model seasonal components. Additionally, the Fourier representation of periodic components can be used to model cycles, c_t, whose period is less obviously related to the frequency at which the observations are taken, and the frequency does not have to be an integer. This methodology is thus used to model cyclical components in this thesis as well. The difference between cyclical and seasonal components is merely that cyclical components have a period shorter period than that of seasonal components, and they are modelled in the same way in this thesis. This section uses the notation of a cyclical component to demonstrate the Fourier representation of cycles or seasonals. In its simplest form, c_t is a pure sine wave, and is modelled as follows

$$c_t = \tilde{c}\cos\lambda_c t + \tilde{c}^*\sin\lambda_c t \tag{3.7.7}$$

where λ_c is the frequency of the cycle, $2\pi/\lambda_c$ is the period and quantities $\tilde{c}$ and $\tilde{c}^*$ are constants . This cyclical component can be allowed to vary stochastically over time

$$\begin{aligned} c_{t+1} &= c_t\cos\lambda_c + c_t^*\sin\lambda_c + \tilde{\omega}_t \\ c_{t+1}^* &= -c_t\sin\lambda_c + c_t^*\cos\lambda_c + \tilde{\omega}_t^* \end{aligned} \tag{3.7.8}$$

with

$$c_t^* = -\tilde{c}\sin\lambda_c t + \tilde{c}^*\cos\lambda_c t \tag{3.7.9}$$

where $\tilde{\omega}_t$ and $\tilde{\omega}_t^*$ are independent $N(0,\sigma_{\tilde{\omega}}^2)$ variables. The frequency λ_c can be treated as an unknown parameter to be estimated, but for the purposes of this thesis it is known.

3.7.1.3 Explanatory Variables

Explanatory variables or regression terms can also be incorporated into structural time series models. If there are *k* regressors $x_{1t},...,x_{kt}$ with regression coefficients $\beta_1,...,\beta_k$ that are constant over time, they can be included into the model as seen by (3.2.3). Additionally, these regression coefficients can be structured to vary over time by modelling them as random walks of the form

$$\beta_{j,t} = \beta_{j,t-1} + \chi_{j,t}, \qquad \chi_{j,t} \sim N(0,\sigma_\chi^2) \qquad j=1,\ldots,k \tag{3.7.10}$$

3.7.1.4 Structural Time Series Models and State Space Models

All structural time series models have a state space representation. This representation relates the disturbance vector (ε_t) to the observation vector (y_t) via a Markov process (α_t), through the relation seen in (3.2.1). The *local level model* (3.7.3) is essentially in state space form as it stands. Since ε_t and η_t are uncorrelated in all time periods, the fact that the transition equation is shifted forward in time in (3.2.1) is not important, so $Z = T = 1$, $\eta_t = \varepsilon_t$, $H = \sigma_\varepsilon^2$ and $G = \sigma_\xi^2$ (Harvey et al., 1998). As another example, the *local linear trend model*, can be put into state space form in (3.2.1) as follows:

$$y_t = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \mu_t \\ l_t \end{pmatrix} + \varepsilon_t,$$

$$\begin{pmatrix} \mu_{t+1} \\ l_{t+1} \end{pmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} \mu_t \\ l_t \end{pmatrix} + \begin{pmatrix} \xi_t \\ \zeta_t \end{pmatrix}$$

with $Z = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $\alpha_t = \begin{pmatrix} \mu_t \\ l_t \end{pmatrix} \alpha_t = \begin{pmatrix} \mu_t \\ v_t \end{pmatrix}$, $T = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ and $\eta_t = \begin{pmatrix} \xi_t \\ \zeta_t \end{pmatrix}$. Similarly, all structural time series models can be written in such state space representation.

3.7.2 Multivariate Structural Time Series Models

Structural time series models can be generalized to accommodate multivariate time series. As an example, consider a local level model for an N x 1 vector of observations y_t, that is

$$y_t = \mu_t + \varepsilon_t$$
$$\mu_t = \mu_{t-1} + \eta_t,$$

where μ_t, ε_t and η_t are N x 1 vectors such that

$$\varepsilon_t \sim N(0, \sum_{\varepsilon}), \qquad \eta_t \sim N(0, \sum_{\eta})$$

with N x N variance matrices $\sum_{\varepsilon}$ and $\sum_{\eta}$. This model is commonly known as the 'seemingly unrelated time series equations.' Each series in y_t is modelled as in the univariate case, but the disturbances may be correlated instantaneously across series. In the case of the augmented model in (3.7.2) with trend, cyclical and seasonal components, the disturbances associated with the components become vectors, which have N x N variance matrices. The link across the N different time series is through the correlations of the disturbances driving the components; $\sum_{\varepsilon}$ and $\sum_{\eta}$ are thus non-diagonal matrices. Consider a multivariate local level model with the assumption that, for example, the rank of $\sum_{\eta}$ is $r < N$. The model then contains only r underlying level components, also known as common levels. Reordering the series, the model can be written as

$$y_t = a + A\mu_t^* + \varepsilon_t,$$
$$\mu_{t+1}^* = \mu_t^* + \eta_t^*,$$

where μ_t^* and η_t^* are r x 1 vectors, a is a N x 1 vector and A is a N x r matrix. Also

$$a = \begin{pmatrix} 0 \\ a^* \end{pmatrix}, \qquad A = \begin{bmatrix} I_r \\ A^* \end{bmatrix}, \qquad \eta_t^* \sim N(0, \sum_{\eta}^*)$$

where a^* is a (N-r) x 1 vector and A^* is a (N-r) x r matrix of nonzero values where variance matrix $\sum_{\eta}^*$ is an r x r positive definite matrix. The matrix A may be interpreted as a factor-loading matrix. When there is more than one common factor ($r > 1$), the factor loadings are not unique.

3.8 Bayesian Analysis

Thus far this thesis has only considered the analysis of observations generated by a linear Gaussian state space model from a frequentist point of view. This section describes the analysis from a Bayesian point of view. For a more detailed discussion see Durbin (1987) and Durbin (1988).

According to Petris et al. (2007), Bayesian inference has an appealing coherence and simplicity, as it accounts for the uncertainty about the true values of model parameters in a natural way. Inference on the unknown model parameters, ψ, is solved by computing the conditional distribution given the sampling results. Prior knowledge about ψ is expressed through the prior, $p(\psi)$, and the likelihood is expressed by $p(y \mid \psi)$. Using Bayes theorem, one can compute the conditional density $p(\psi \mid y)$ though $p(\psi \mid y) = p(y \mid \psi)p(\psi)/m(y)$, where $m(y)$ is the marginal density of $y_1,...,y_n$. In contrast to the Bayesian approach, the frequentist statistical inference considered in Section 3.5 does not have a probability distribution for unknown parameters, and inference on ψ is based on the determination of estimators with good properties, confidence intervals and hypothesis testing. Since the value of the parameter ψ does not vary, it is not interpretable as a random variable in a frequentist sense; neither can the probability that ψ takes values in a certain interval have a frequentist interpretation. Adopting subjective probability instead, ψ is a random quantity simply because its value is uncertain, and one should formalize information on ψ by means of probability.

3.8.1 Posterior Analysis of State Vector

For the model specified in (3.2.1), where the parameter vector ψ is specified, the posterior analysis of the model is straightforward. The Kalman filter and smoother provide the posterior means, variances and covariance's of the state vector α_t given the data. Since the model is Gaussian, posterior densities are normal, so these can be estimated easily from standard properties of the normal distribution.

However, when the parameter vector ψ is not fixed and known, the analysis becomes more complex. In this case ψ is treated as a random vector with a prior density $p(\psi)$, a proper prior. The problem of parameter estimation amounts to calculating the mean of the posterior density $p(\psi \mid y)$, denoted as

$$\bar{x} = E[\alpha \mid y]$$

Additionally,

$$\bar{x}(\psi) = E[\alpha \mid \psi, y]$$

is the conditional expectation of α given ψ and y, where for the purposes of this thesis $\bar{x}(\psi)$ is calculated using the Kalman filter and smoother. For a detailed discussion when this is not the case, see Durbin and Koopman (2001, Chapter 13). It is seen that

$$\bar{x} = \int \bar{x}(\psi) p(\psi \mid y) d\psi$$

By Bayes theorem, the posterior density is calculated though $p(\psi \mid y) = Kp(\psi)p(y \mid \psi)$ where K is the normalising constant defined by

$$K^{-1} = \int p(\psi) p(y \mid \psi) d\psi$$

Therefore

$$\bar{x} = \frac{\int \bar{x}(\psi) p(\psi) p(y \mid \psi) d\psi}{\int p(\psi) p(y \mid \psi) d\psi} \tag{3.8.1}$$

Now $p(y \mid \psi)$ is the likelihood for which linear Gaussian models are calculated by the Kalman filter, as calculated in Sections 3.5.2 and 3.5.3. The integrals in (3.8.1) can be computed by numerically in cases where the dimensionality of ψ is not large, but this is seldom.

The main technique used for Bayesian analysis is however simulation, the approach used in this thesis. In principle, simulation could be applied directly to formula (3.8.1) by drawing a random sample $\psi^{(1)},...,\psi^{(N)}$ from the distribution with density $p(\psi)$ and then estimating the numerator and denominator of (3.8.1) by the sample means of $\bar{x}(\psi)p(y \mid \psi)$ and $p(y \mid \psi)$ respectively. However the estimator is inefficient in many cases. To overcome this, one can use a simulation

technique known as importance sampling to achieve greater efficiency. A full discussion on importance sampling can be seen in Ripley (1987) or Geweke (1989). Another approach to Bayesian analysis based on simulation is provided by the Markov Chain Monte Carlo method, the approach used in this thesis.

3.8.2 Markov Chain Monte Carlo

While for very few cases it is possible to compute the posterior distribution of states and unknown parameters in closed form, one generally has to resort to Monte Carlo methods to draw a sample from the posterior distribution of interest. One way of obtaining a sample from a joint posterior of parameters and unobservable states is to run a Gibbs sampler, alternating draws from the full conditional distribution of the states and from the full conditionals of the parameters. Generating the parameters is model dependent, and a draw from the full conditional distribution of the states can be obtained using the forward filtering backward sampling algorithm (FFBS), developed independently by Carter and Kohn (1994), Früwirth-Schnatter (1994) and Shephard (1994), and is essentially a simulation version of the Kalman smoother. This is a technique where random draws are generated from the conditional densities $p(\varepsilon \mid y,\psi)$, $p(\eta \mid y,\psi)$ and $p(\alpha \mid y,\psi)$ for a given parameter vector ψ. For details on how these conditional densities are calculated using the FFBS algorithm, see Durbin and Koopman (2011, §4.7). The basic idea of a Gibbs sampler, as described by Petris et al (2007), is to evaluate the posterior mean of α or of the parameter vector ψ via simulation by choosing samples from a joint density $p(\psi,\alpha \mid y)$; sampling from this joint density is implemented as a Markov chain. To evaluate the posterior of the parameter vector, ψ, the Gibbs sampling approach consists initialising ψ, say $\psi = \psi^{(0)}$, and then one repeatedly cycles through two simulation steps:

For i=1,…,n

1. Sample $\alpha^{(i)}$ from $p(\alpha \mid y,\psi^{(i-1)})$ using FFBS
2. Sample $\psi^{(i)}$ from $p(\psi \mid y,\alpha^{(i)})$

for *i=1,2*…. After a number of 'burn-in' iterations one is allowed to treat the samples from step (2) as being generated from the density $p(\psi \mid y)$. Implementing

the two steps described above is not straightforward however. Sampling from $p(\psi \mid y,\alpha)$ depends partly on the model for ψ and is usually only possible up to proportionality. To sample under these circumstances accept-reject algorithms such as the Metropolis Hastings algorithm is implemented. Within the Gibbs sampler, when generating the states, the model parameters are fixed at their most recently generated value, and therefore the problem reduces to drawing from the conditional distribution of the states given the observations for a completely specified dynamic linear model. If one is not interested in the states but only in the unknown parameters, keeping the states in the posterior distribution simplifies the Gibbs sampler. This typically happens when there are unknown parameters in the system equation and system variances, since conditioning the states makes those parameters independent of the data and results in known distributions (Petris, 2009). In this thesis, for which the parameter vector consists only of variances of disturbances associated with the components, the distribution of the parameter vector can be modelled such that sampling from $p(\psi \mid y,\alpha)$ in step (2) is relatively straightforward, as explained by Petris (2009). Using the Gibbs sampler, prior distributions for the inverse of unknown variances (precisions) are gamma distributions. Assume for simplicity that the observations are univariate and that the unknown observation and system variance parameters are defined by their precisions $\psi_y,\psi_{\alpha,1},...,\psi_{\alpha,p}$. These observation and system variances are time-invariant and are related to the unknown parameters as follows:

$$H = \psi_y^{-1},$$
$$Q = diag(\psi_{\alpha,1}^{-1},...,\psi_{\alpha,p}^{-1})$$

The parameters have independent gamma distributions, a priori:

$$\psi_y^{-1} \sim IG\left(\frac{a_y^2}{b_y},\ \frac{a_y}{b_y}\right),$$
$$\psi_{\alpha,i}^{-1} \sim IG\left(\frac{a_{\alpha,i}^2}{b_{\alpha,i}},\ \frac{a_{\alpha,i}}{b_{\alpha,i}}\right), \qquad \text{for } i=1,...,p.$$

where a_y, $a_{\alpha,i}$ are the means and b_y, $b_{\alpha,i}$ the variances of these respectively. Each full conditional distribution is proportional to the joint distribution, with the joint distribution of the observations, states and unknown parameters seen to be

$$p(y,\alpha,\psi) = \prod_{t=1}^{n} p(y_t \mid \alpha_t,\psi_y)\cdot \prod_{t=1}^{n} p(\alpha_t \mid \alpha_{t-1},\psi_{\alpha,1},...,\psi_{\alpha,p})\cdot\ p(\alpha_0)\cdot\ p(\psi_y)\cdot \prod_{i=1}^{p} p(\psi_{\alpha,i})$$

A Gibbs sampler draws from the full conditional distribution of the states and from the full conditional distributions of $\psi_y,\psi_{\alpha,1},...,\psi_{\alpha,p}$ in turn. Sampling the states is performed using the FFBS algorithm. The full conditional distribution of ψ_y is derived as:

$$\begin{aligned} p(\psi_y \mid ...) &\propto \prod_{t=1}^{n} p(y_t \mid \alpha_t,\psi_y)\cdot\ p(\psi_y) \\ &\propto \psi_y^{\frac{n}{2}+\frac{a_y^2}{b_y}} \exp\left\{-\psi_y\cdot\left[\frac{1}{2}\sum_{t=1}^{n}(y_t - Z_t\alpha_t)^2 + \frac{a_y}{b_y}\right]\right\} \end{aligned}$$

Therefore the full conditional of ψ_y is again a gamma distribution

$$\psi_y^{-1} \mid ... \sim IG\left(\frac{a^2}{b}+\frac{n}{2},\quad \frac{a}{b}+\frac{1}{2}SS_y\right)$$

with $SS_y = \sum_{t=1}^{n}(y_t - Z_t\alpha_t)^2$. Similarly, it can be shown that the full conditionals of the $\psi_{\alpha,i}{}'s$ are as follows:

$$\psi_{\alpha,i}^{-1} \mid ... \sim IG\left(\frac{a_{\alpha,i}^2}{b_{\alpha,i}}+\frac{n}{2},\quad \frac{a_{\alpha,i}}{b_{\alpha,i}}+\frac{1}{2}SS_{\alpha,i}\right)$$

where $SS_{\alpha,i} = \sum_{t=1}^{T}(\alpha_{t,i} - (T_t\alpha_{t-1})_i)^2$. Implementing the Gibbs sampler with the full conditions described above, and discarding the first few observations as burn in values, generates posterior estimates of the unknown variances.

3.9 Structural Time Series Models with R

Structural time series models can be modelled using a wide variety of packages in R. The package `dlm` (Petris, 2010) was the package used in this thesis to construct, smooth and filter both univariate and multivariate linear Gaussian state space models. Additionally, these models can be rendered time-varying/dynamic.

According to Petris and Petrone (2011), the `dlm` package follows the notation and algorithms used by West and Harrison (1997), who focused on these models from a Bayesian perspective. This thesis has focused more on a classical approach and, as far as notation and algorithms are concerned, follows those of Durbin and Koopman (2001) and Harvey (1987). The model defined in (3.2.1), in the notation of the classical approach, is defined in the `dlm` package with slightly different notation, that is in the notation of the Bayesian approach. These differences are noted in Table 3.1:

Table 3.1: Notation for model (3.2.1) in R package dlm.

Notation in Model (3.2.1)	**Notation in R package 'dlm'**
Z	F
T	G
H	V
Q	W
a_1	m_0
P_1	C_0

3.9.1 The Trend Component

The trend component (μ_t) of a structural time series model in (3.7.1) or (3.7.2), can be modelled using the `dlmModPoly(order, dV, dW)` function, where `dV` defines the observation variance (σ_ε^2) and `dW`, a diagonal matrix, containing the state evolution variances. The `dV` and `dW` components can either be specified or calculated using maximum likelihood. When `order` is equal to one, this defines a *local level model* defined in (3.7.3), with `dW` merely a scalar corresponding to the variance of the trend level component (σ_ξ^2). When `order` is equal to two, this defines the trend with a slope component, forming a (2×2) `dW` matrix. The first diagonal component of this matrix corresponds to the variance of the trend level (σ_ξ^2), and the second diagonal component to the variance of the slope (σ_ζ^2). When `dW` is defined with both the trend level and slope diagonal elements ≥ 0, a *local linear trend model* described in (3.7.4) is defined. When both diagonal elements are fixed to zero, a *deterministic model* described in

(3.7.5) is defined. When the diagonal element of dW corresponding to the slope component is fixed to zero, but the trend level diagonal component is ≥ 0, a *random walk with drift model* (3.7.6) is defined. Lastly, when the diagonal element of dW corresponding to the trend level component is fixed to zero, but the slope component diagonal is ≥ 0, a *smooth trend model* is defined. As an example, a *local linear model* with both observation and state variances specified as 0.1 and 0.2 respectively, is defined as follows:

```
dlmModPoly(order=1, dV=0.1, dW=0.2)
```

The case where the variances are estimated using maximum likelihood or Bayesian analysis is discussed in Sections 3.9.4 and 3.9.9 below respectively.

3.9.2 Seasonal and Cyclical Components

The seasonal (γ_t) or cyclical (c_t) components are modelled using the Fourier representation in (3.7.8) in this thesis. The function used to model these components as such, is the `dlmModTrig(s,q,om,tau,dV,dW)` function, where `s` is the period specified as an integer, `tau` the period specified as a non-integer, `om` the frequency and `q` the number of harmonics. Note that `dV` and `dW` are similarly defined as in Section 3.9.1. For each harmonic, two variance components $\sigma^2_{\tilde{\omega}}$ and $\sigma^2_{\tilde{\omega}^*}$ are specified in the `dW` diagonal matrix. This package assumes $\sigma^2_{\tilde{\omega}} = \sigma^2_{\tilde{\omega}^*}$, with the same variance applied across all harmonics, as only one input is accepted for dW. However, this can be more flexibly defined, as will be described in Section 3.9.4, where variances differ across all harmonics. The assumption of $\sigma^2_{\tilde{\omega}} = \sigma^2_{\tilde{\omega}^*}$ is however kept throughout this thesis, as throughout the literature reviewed for this thesis, this was a standard assumption. Harvey (1985) states that not much is lost by implementing this assumption in terms of fit, and it is very advantageous in terms of numerical optimisation to have only one parameter for estimation instead of two. As an example, a cycle with period equal to 12.2, with 2 harmonics, and a constant state evolution variance across the harmonics is demonstrated:

```
dlmModTrig(tau=12.2, q=2, dV=0.1, dW=0.2)
```

3.9.3 Explanatory Variables

Regression terms are added into the model using the `dlmModReg(X, addInt, dV, dW)` function, where `X` is a matrix with the time series of explanatory variables in columns, and `addInt` is the logical argument of whether an intercept should be added. Intercepts are captured in the trend component of the models constructed in this thesis and are thus not added to regression components. Again `dV` is the observation variance and dW is the state evolution variance corresponding to the explanatory variables (σ_χ^2), following the formulation as seen in (3.7.10). As an example, consider a time series of explanatory variable temperature, modelled with no intercept, and where the observation and state evolution variances are known and specified:

```
dlmModReg(X=Temperature, addInt=FALSE, dV=0.1, dW=0.2)
```

The case where the observation and state evolution variances are estimated is discussed in Section 3.9.4 below. When modelling explanatory variables, the `dlm` package requires for the dependent variable in the model to have corresponding missing values with the explanatory variables in the model. This can be very inconvenient when modelling, especially for the purposes of comparing models, as the data modelled needs to be altered to match the data of the explanatory variables included in terms of missing values.

3.9.4 Unknown Parameter Estimation

Unknown model parameters, where in this thesis these specifically refer to unknown observation and state evolution variances, are estimated via maximum likelihood procedures using the `dlmMLE` function. This function makes use of the "L-BFGS-B" optimisation algorithm discussed in Section 3.5.4 in estimating these unknown parameters. Maximum likelihood described in Section 3.5 suggests that the Kalman filter and smoothing algorithms must first be performed to generate the v_t and F_t matrices, however the loglikelihood function constructed within the `dlmMLE` function estimates these matrices without needing to perform smoothing and filtering beforehand. Unknown parameters can also be estimated through

Bayesian analysis, discussed in Section 3.8. An example of how these variances are estimated is provided below, where the model constructed is a *local linear model*, with a cyclical component of period 12.2 and 2 harmonics, and an explanatory variable of the time series temperature with no intercept. The explanatory variable and cyclical components are modelled dynamically, with each harmonic in the cyclical component allowed a different variance estimate:

```
buildit <- function (par)
{
mod <- dlmModPoly(order=1)+dlmModTrig(tau=12.2,q=2)
+dlmModReg(X=Temperature, addInt=FALSE)
V(mod) <- exp(par[1])
diag(W(mod))[1] <- exp(par[2])
diag(W(mod))[2:3] <- exp(par[3])
diag(W(mod))[4:5] <- exp(par[4])
diag(W(mod))[6] <- exp(par[5])
return(mod)
}
fit2 <- dlmMLE (data, par = c(1,1,1,1,1), buildit )
dlmY <- buildit(fit2$par)
```

3.9.5 Filtering and Smoothing

Once the model is fully specified with all parameters estimated, the filtering and smoothing procedures discussed in Section 3.3, are performed using the `dlmFilter(y, mod)` and `dlmSmooth(y, mod)` functions, where `y` denotes the data to which the fully specified model, `mod`, must be fit. In the case of the above example, `mod=dlmY`.

Section 4.8 of this thesis presents the model selected for modelling Kapenta catch over the period 1 January 1995 to 31 December 2003. This is referred to as model D2. The code for specifying this model, estimating unknown parameters, filtering and smoothing is provided in Appendix A. This model uses catch per unit effort (CPUE) as the variable of interest, and possesses a weekly sampling frequency.

Reasons for this will all be discussed in Chapter 4.

3.9.6 Model Diagnostics, Comparisons and Goodness-of-Fit

Many of the model diagnostics and fit comparisons, as discussed in Section 3.7, were not available in `dlm` and code was therefore written specifically for these. The code constructing model diagnostics and goodness-of-fit measures discussed here for the model built in Appendix A, is given in Appendix B.

Measures of fit, namely R_D^2 and prediction error variance (P.E.V) measures, were calculated using the methodology described in Harvey (1989), described in Section 3.6.2. Code showing how to calculate these are presented in Appendix B. The P.E.V is derived by calculating the variance of the one-step ahead forecast errors, f_t, as calculated in (3.3.1), and evaluating where this value converges as $t \to \infty$. The R_D^2 value uses this P.E.V in its calculation.

Model diagnostics, namely the autocorrelation function and scatter plot of the standardized residuals are generated in R through the `acf` and `plot` functions. Standardised residuals are calculated by generating raw residuals, through the `residuals` function in `dlm`, and dividing this by the square root of the one-step ahead error variance, f_t.

To compare models, AIC measures are used, and are calculated as in (3.6.1). This thesis, however, produces two AIC values due to different methodologies for calculating the loglikelihood. Although the likelihood of a model can be calculated through a direct function (`dlmLL`) in the `dlm` package, it is not calculated in the means discussed by Durbin and Koopman (2001) and Section 3.6 of this thesis. The AIC calculated by the `dlm` package, denoted as AIC_1, makes use of the loglikelihood calculated by:

$$\log L(y) = -\frac{1}{2}\sum_{t=d+1}^{n}\log|f_t| - \frac{1}{2}\sum_{t=d+1}^{n} v_t' f_t^{-1} v_t \qquad (3.9.1)$$

where v_t and f_t are defined in (3.3.1). The AIC preferred, denoted AIC_2 in this thesis, makes use of the loglikelihood described by Durbin and Koopman (2001) and in Section 3.6 of this thesis, as:

$$\log L(y) = -\frac{(n-d)N}{2}\log 2\pi - \frac{1}{2}\sum_{t=d+1}^{n}\log|f_t| - \frac{1}{2}\sum_{t=d+1}^{n}v_t' f_t^{-1} v_t$$

Note the above likelihood is denoted for the diffuse case. In calculating the loglikelihood for AIC_2, the methodology used in the `KFAS` (Helske, 2011) package is used, as the `KFAS` package is more in the tradition of Durbin and Koopman (2001) and Harvey (1989). The function in the `KFAS` package was adapted to include model vectors and matrices developed in the `dlm` package; see Appendix B for details. It must be noted that sometimes AIC_2 measures are calculated as 'NA'. One possible reason for the 'NA' occurrence is when f_t equals zero at some $t=c+1,...,n$, where c is defined as the last index of the diffuse phase. As will be seen in Chapter 4, models are built using both daily and weekly data. The AIC_2 values could not be calculated when using daily data, for reasons not fully understood but this could perhaps be attributed to the large amount of unexplained variation when using daily data.

3.9.7 Code-Checking

The code given in Appendices A and B was initially tested and checked against results generated by Harvey (1985), where structural time series models were constructed and assessed, using the diagnostic, fit and model comparison measures discussed in Section 3.9.6. The Nelson and Plosser original data used is available in R, called `nporg` in the `urca` (Pfaff, 2006) package. Parameter estimates and goodness-of-fit values were generated by the code in Appendices A and B, and are seen to be very close to those provided in Harvey (1985), where `STAMP` (Commandeur et al., 2011) was used instead of `dlm`. Table 3.2 shows a sample of these comparisons, and compares the estimates calculated in Harvey (1985), to estimates calculated using the code of this thesis. The estimates in Table 3.2 generated by Harvey (1997) can be observed in Table 2 of his paper, for the GNP dataset observed between 1909 to 1947. Model (i) refers to a *random walk with drift model*, and Model (ii) refers to a *local linear trend model* with

cycle, where period is equal to 7 with 1 harmonic. Parameters *d* and *q*, discussed in Section 3.6.3, are also specified for modes (i) and (ii).

Table 3.2: Table comparing the results generated by Harvey (1997) to the results generated by this thesis for the Nelson and Plosser dataset

		Estimates ($\times 10^4$)					$\tilde{\sigma}^2 \times 10^3$	
Model	Author	σ_ε^2	σ_ξ^2	σ_ζ^2	$\sigma_{\tilde{\omega}}^2 = \omega_{\tilde{\omega}^*}^2$	Log L	P.E.V	R^2
(i) d=1, q=1	Harvey	0.0	62.2	0.0	-	73.66	6.22	0.00
	Dalmeyer	0.0	62.2	0.0	-	73.66	6.39	-0.05
(ii) d=3, q=3	Harvey	0.0	23.7	6.1	3.3	76.49	5.88	0.05
	Dalmeyer	0.0	24.5	5.7	3.3	70.49	6.23	0.03

Results are seen to be very close in value, and lead to the conclusion that code written for the purposes of this thesis is correct. Additionally, AIC values were checked with values generated by the `dlmodeler` (Szymanski, 2012) package to ensure parameter specifications were correct.

3.9.8 Confidence Intervals

In deciding whether to model the trend component of Kapenta catch as a *local linear model*, or with a slope component in a model such as a *local linear trend model*, the presence of a slope component is tested for significance. This is tested through the use of confidence intervals for both the slope component and for changes in the mean trend level (smoothed estimate of each trend level component in the time series less the final smoothed trend level estimate). Confidence intervals are constructed using the covariances for the smoothed estimates, discussed in Section 3.3.3. These were constructed using adapted code written by Dr Birgit Erni, seen in Appendix C, which makes use of the `U.R./D.R` and `U.C./D.C`. output values from the Kalman filter function, `dlmFilter`, and the `U.S./D.S`. output values from the smoothing function, `dlmSmooth`, in the `dlm` package. `U.R./D.R.` together give the singular value decomposition of the

variances of the prediction errors, `U.C./D.C` together give the singular value decomposition of the variances of the estimation errors and `U.S./D.S.` together give the singular value decomposition of the variances of the smoothing errors.

3.9.9 Bayesian Analysis

In addition to calculating unknown parameters through maximum likelihood procedures using the `dlmMLE` function, this thesis also discusses the estimation of unknown parameters from a Bayesian perspective. The `dlm` package does have available functions for this, such as the `dlmGibbsDIG` function that implements a Gibbs sampler for a univariate model having one or more unknown variances. This function, however, proved difficult to use as it could not handle missing values in the data. For this reason Bayesian estimates of unknown parameters were estimated through code written for this thesis, following the example provided by Lavine (n.d.). Here a Gibbs sampler was implemented, where inverse independent gamma priors are assumed for unknown variances. The code for estimating unknown parameters for the model specified in Appendix A is seen in Appendix D.

3.9.10 Multivariate Analysis

Multivariate models are implemented in the `dlm` package through the `dlmSum` function. This function can alternatively be specified as `%+%` the between models for each individual time series. This function cannot include regression components into multivariate models, and returns errors stating that this function does not support time-varying dynamic linear models. This error is however returned whether explanatory variables are included dynamically or not, and thus explanatory variables are not included into multivariate models. For reasons not understood this function does however work with certain time-varying components, such as trend components, even though this function is described to support only non time-varying models. To see how the dlmSum function works suppose, for example there are two time series, the first following a *local linear trend model*, and the second a *local linear model* plus a cyclical component with period equal to 12.2 and 2 harmonics, where observation and state evolution variances are specified. One can construct the multivariate model as follows:

```
dlmModPoly(order=2, dV=0.1, dW=c(0, 0.2))  %+%
(dlmModPoly(order=1, dV=0.1, dW=0.2) +
dlmModTrig(tau=12.2, q=2, dV=0, dW=0.2))
```

Unknown observation and state evolution variances can again be estimated using maximum likelihood. When components within the multivariate model are assumed independent, the covariance matrix $\sum_{\eta}$ (see §3.7.2) is a diagonal matrix and is estimated fairly simply, as seen in Appendix E(1). Here, the univariate model built in Appendix A, less the explanatory variable, is extended into a multivariate model. However, when components within the multivariate model are assumed non-independent, estimating unknown parameters for multivariate models is much more complex. When this is the case, the covariance matrix $\sum_{\eta}$, a non-diagonal matrix, is parameterised using log-Cholesky. Defining θ to be the minimal set of parameters to determine $\sum_{\eta}$, the positive definite matrix may be factored as $\sum_{\eta} = U^T U$, where U is an upper triangular matrix. Setting θ to be the upper triangular elements of U gives the Cholesky parameterisation. If it is required for the diagonal elements of U in the Cholesky factorization to be positive then U is unique, and to avoid constrained estimation one can use the logarithms of the diagonal elements of U. This is known as the log-Cholesky parameterisation, and further details of this methodology can be seen in Pinheiro and Bates (1996). When components within the multivariate model are assumed non-independent, the covariance matrix $\sum_{\eta}$ is specified using this methodology as seen in Appendix E(2). Here, the univariate model built in Appendix A, less the explanatory variable, is again extended into a multivariate model as in Appendix E(1), but the assumption of independent model components is no longer imposed.

3.10 Chapter Conclusion

This chapter has provided the theoretical background to linear Gaussian state space models. The formulation of these models is provided and describes how the Kalman filter and smoothing algorithms are used to provide information on the state vector. Unknown model parameters can be estimated from both a frequentist

and Bayesian approach, and both approaches are investigated in this thesis. Model diagnostics assess standardised residuals for normality, homoscedasticity and independence, through the use of autocorrelation functions, scatter and QQ-plots. Model fits are viewed and compared through R_D^2, AIC and prediction error variance (P.E.V) values, as defined by Harvey (1989). Structural time series models, formulated as linear Gaussian state space models, allow for trend, seasonal, cyclical and regression components to be modelled explicitly. Each of these components can be defined to be time varying by modelling the component stochastically. This chapter concludes with a description of the implementation of structural time series models in R using the `dlm` package. A more detailed description of the `dlm` package is provided in Petris et al. (2007). The application of these models to the Kapenta catch data is now described in Chapter 4.

Chapter 4

Structural Time Series Models Applied to Kapenta Fishing Data

4.1 Introduction

The background to the Kapenta case study was discussed in the introductory chapter. This chapter deals with the application of structural time series models to the Kapenta catch data. Details of the data and an exploratory analysis are presented in Sections 4.2 and 4.3 respectively. Hypotheses on which components will be significant in modelling Kapenta catch are presented in Section 4.4. The model building process is described in Section 4.5, and results for models built using daily Kapenta catch data, along with model-checking and goodness-of-fit results, are presented in Section 4.6. Problems experienced modelling daily data are described in Section 4.7, and weekly Kapenta catch data is modelled instead. The results for models using weekly Kapenta catch data are described in Section 4.8. Unknown model parameters are generally estimated using maximum likelihood, but can alternatively be estimated from a Bayesian perspective; these results are presented in Section 4.9. This chapter concludes with a brief extension of the univariate structural time series model into a multivariate model.

4.2 Data

The data used in this study consists of Kapenta catch data for each of the five basins of Lake Kariba for the period 1 January 1986 to 31 December 2003. Additionally, covariate data such as minimum and maximum temperature, precipitation, lake level and moonlight was also included in this dataset. This data was received from Dr. Altwegg and Dr. Musil from SANBI. SANBI received this data from Ms. Mzime Ndebele-Murisa who used it for her work on the paper *Implications of a changing climate on the Kapenta fish stocks of Lake Kariba, Zimbabwe*, written in collaboration with Trevor Hill and Emmanuel Mashonjowa in 2011.

For the purposes of this thesis, daily catch data per vessel (in kg) and the daily number of hauls per vessel are both summed to produce total catch per day (in kg), and the total number of hauls per day. A daily variable known as the 'catch per unit effort' (in kg), referred to as CPUE, is calculated by dividing the daily total catch by the daily total number of hauls. There are two possible variables of interest to model, total daily catch or daily CPUE. Only CPUE is modelled, and the justification for modelling this variable is described in Section 4.5. The pelagic Kapenta fishery is license-controlled, mechanized and performed with light attractions on lift nets from pontoon rigs. On the Zimbabwean side of the lake, each company returns statistics with landings and the number of nights fished. The daily Kapenta catch and haul data per vessel was, according to Ndebele-Murisa et al. (2011), primarily collected by the Lake Kariba Fisheries Research Station. However after 2002 their records were inconsistent or irretrievable due to technical difficulties. The focus of this thesis is on modelling the catch data for basin 5 of Lake Kariba, and was chosen due to the greater availability of catch data available for this basin in terms of total catch, with less missing values. Additionally basin 5 is closest to the town of Kariba where covariate data are recorded, making this data most applicable to this basin.

Covariate data consist of climatic data, such as monthly average minimum and maximum temperatures (°C) and monthly average precipitation (mm), originally

collected from the Kariba weather station and the Meteorological Society of Zimbabwe. Due to the fact that these are monthly and not daily values, these values were recollected by myself from Tutiempo weather for the purposes of this paper. Tutiempo receives this information from the weather station situated in Kariba, Zimbabwe. These variables are now defined as mean daily temperature (°C) and daily precipitation (mm). It must be noted that the availability of this data was poor over the earlier years of this study, with observations erratically available, and for this reason was only collected for the period 1 January 1995 to 31 December 2003. Additionally daily lake level data (metres above sea level) and daily moonlight data (a variable on a scale of 0 to 1 indicating how much light was emitted from the moon) is also available. According to Ndebele-Murisa et al. (2011) lake level data was sourced from the Zambezi River Association (ZRA), and the daily moonlight data from the Astronomical Applications Department. One aim of this study is to assess the impact of daily evening cloud cover on Kapenta catch, since fishing occurs in the evenings. These data were, however, very expensive and due to budget constraints only three years could be purchased, from 1 January 1986 to 31 December 1988. This was sourced from the Meteorological Society of Zimbabwe

The most important features of the data described are summarised in Table 4.1.

4.3 Exploratory Data Analysis

This section provides an exploratory analysis of the Kapenta catch data for basin 5 of Lake Kariba. A plot of the daily Kapenta CPUE for basin 5 from 1 January 1986 to 31 December 2003 is presented in Figure 4.1. It is clear from the graph that Kapenta catch has a cyclical component, corresponding to a period of one year. This pattern is more clearly observed over the second half of the data, from 1 January 1995 to 31 December 2003, than over the first half of the data, from 1 January 1986 to 31 December 1994. The Kapenta CPUE dataset appears to follow different underlying patterns over the two halves of the data; this can be seen in Figure 4.1 where the second half of the data appears to be more stationary overall and possesses a more obvious cyclical component. Non-parametric smoothing using a moving average with a window of 30 days is presented in Figure 4.3(a).

Table 4.1: Table showing characteristics of CPUE and covariate data

Variable (daily)	Period of Availability	Units of measurement	% of Missing Values	Data Source	Notes on Variable
Total Catch Basin 5	01/01/86 - 31/12/03	Kg	1.55%	Lake Kariba Fisheries Research Station	The number of hauls per basin, and therefore CPUE, has the same % availability and data source as Total Catch data per basin
Total Catch Basin 4	01/01/86 - 31/12/03	Kg	7.97%		
Total Catch Basin 3	01/01/86 - 31/12/03	Kg	3.92%		
Total Catch Basin 2	01/01/86 - 31/12/03	Kg	4.81%		
Total Catch Basin 1	01/01/86 - 31/12/03	Kg	68.6%		
Mean Temperature	01/01/95 - 31/12/03	°C	13.84%	Tutiempo Weather	Sourced to Tutiempo from Kariba Weather Station Available at: http://www.tutiempo.net/en/Climate/Kariba/01-1995/677610.htm
Precipitation	01/01/95 - 31/12/03	mm	16.55%	Tutiempo Weather	
Moonlight	01/01/86 - 31/12/03	Proportion [0,1]	0%	Astronomical Applications Department	Available at:: http://aa.usno.navy.mil/data/docs/MoonFraction.php
Cloud Cover	01/01/86 - 31/12/88	No of Octaves	0%	Zimbabwe Meteorological Services	Limited availability due to cost constraints
Lake Level	01/01/86 - 31/12/03	m's above sea level	0%	Zambezi River Authority (ZRA)	

The presence of a strong yearly cycle is again seen, particularly over the second half of the data. Figure 4.3(b) shows 2 years of the moving average series taken over the first half of the data, and Figure 4.3(c) shows this for the second half of the data. Here, the yearly cyclical components can be graphically seen for both halves of the data. In Chapter 1 it was noted that Kapenta catch is thought to be seasonal, as only in winter do Kapenta move out into the open waters after breeding in the summer months. The yearly cycle is thought to correspond to this phenomena. To investigate this seasonal effect further, Figures 4.4(a) and 4.4(b) show the daily average of CPUE per month over the first and second halves of the data respectively. Here again it is seen that CPUE has a seasonal effect over both halves of the data, with average CPUE increasing over the cold months of July/August/September, and low over the warmer months starting from October. Figures 4.4(a), 4.3(a) and 4.3(b) indicates that this yearly cyclical component may be present over the first half of the data, even though it is not graphically visible from Figure 4.1.

Figure 4.2 shows daily CPUE over a 1-year period, making it possible to look at finer-scale patterns in the data. It is observed that an additional cyclical component is present in the data corresponding to a period of approximately 1 month. In Chapter 1 it was noted that Kapenta fishing takes place in the evenings with light attractions. The moon cycle corresponds to a period of 29.53 days and the observed monthly cycle in the data is thought to be related to this phenomenon, where the amount of light emitted from the moon possibly affects the efficacy of the light attractions. Figure 4.5 shows a 1-year plot of CPUE with moonlight (where both variables are standardised through subtracting their means and dividing by their standard deviations). It is seen that there is a strong relationship between these two variables. As the phase of the moon moves toward full moon, CPUE decreases and visa versa. This confirms the attribution of the monthly cycle to moon phase. Figure 4.6 displays the autocorrelation function for the daily CPUE data. From Figure 4.6. evidence of a strong monthly cycle is again seen. Taking the mean over every month in the data to remove the effect of the monthly cycle from the autocorrelation function, the autocorrelation function in Figure 4.7 again confirms the presence of a strong yearly cycle in the data.

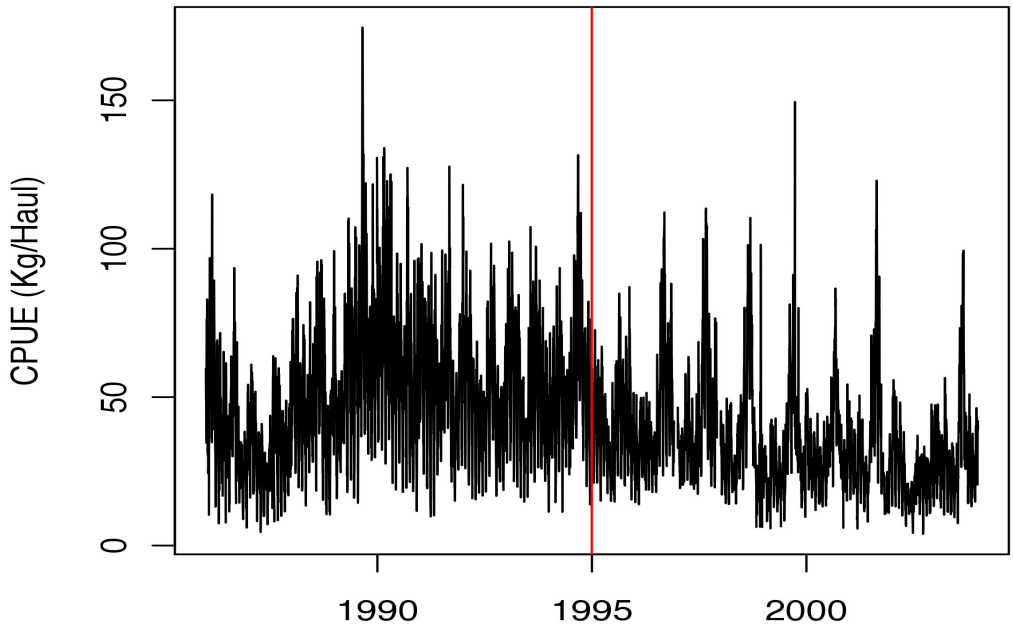


Figure 4.1: Plot showing daily CPUE for 1 January 1986 to 31 December 2003

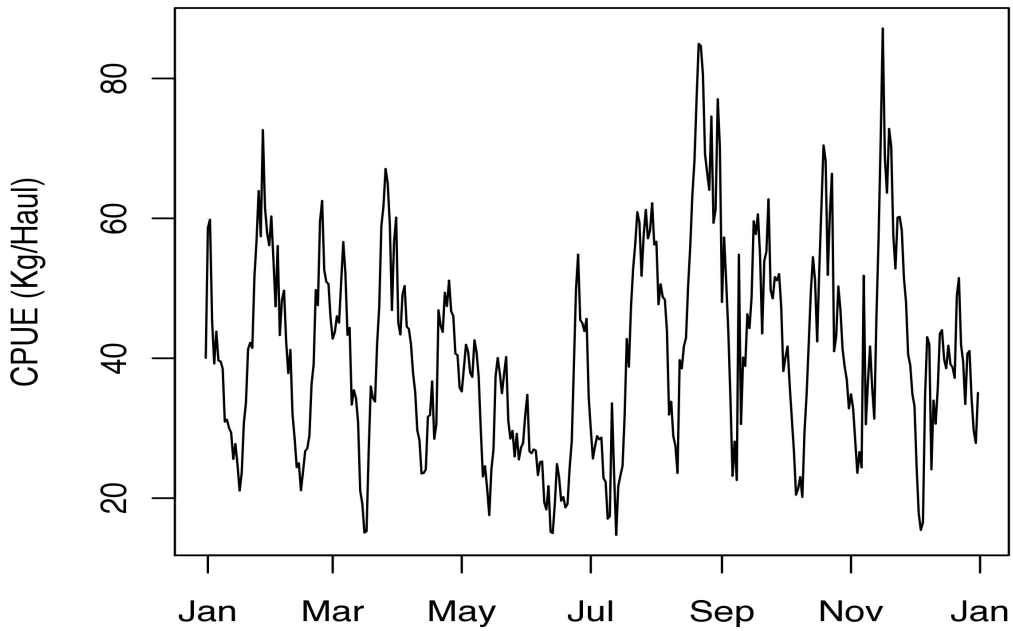


Figure 4.2: Plot showing 1 year sample of daily CPUE data

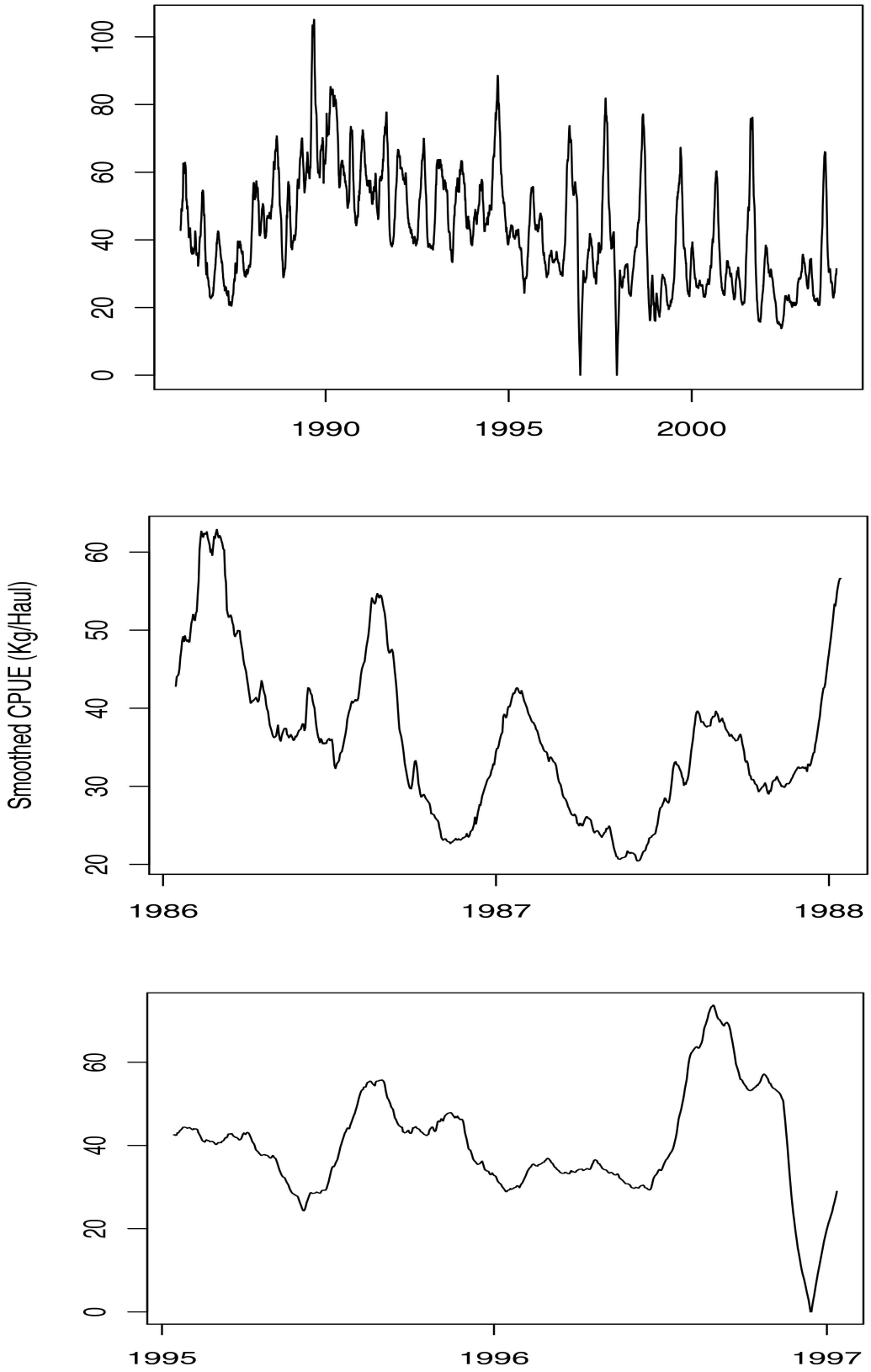


Figure 4.3: (a) Plot showing non-parametric smoothing using a moving average with a window of 30 days over period I January 1986 to 31 December 2003
(b) Plot showing 2-years of non-parametric smoothing using a moving average with a window of 30 days sampled from the period 1 January 1986 to 31 December 1994
(c) Plot showing 2-years of non-parametric smoothing using a moving average with a window of 30 days sampled from the period 1 January 1995 to 31 December 2003

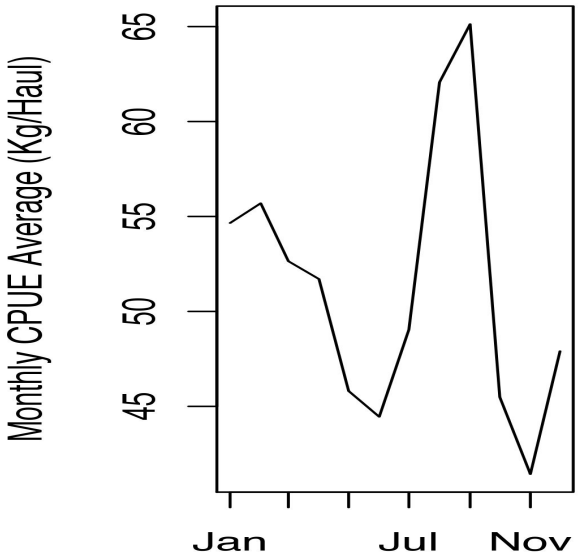


Figure 4.4(a): Plot showing average daily CPUE per month over the period 1 January 1986 to 31 December 1994.

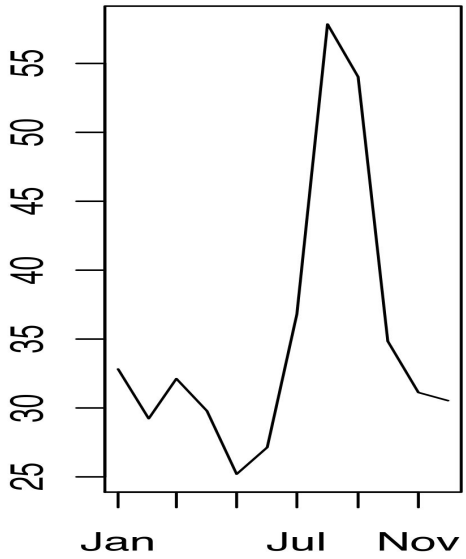


Figure 4.4(b): Plot showing average daily CPUE per month over the period 1 January 1995 to 31 December 2003.

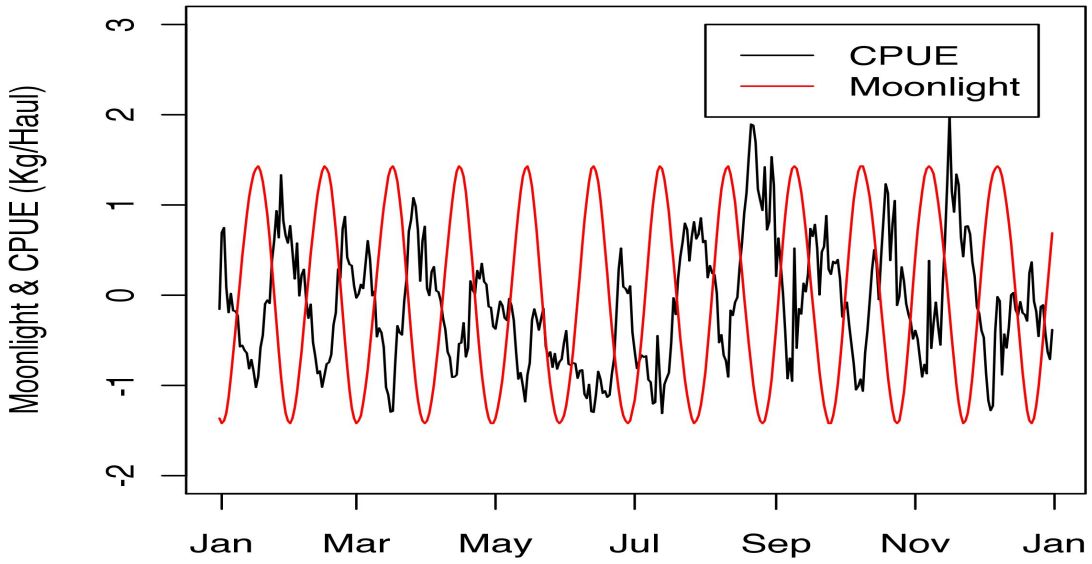


Figure 4.5: Plot showing 1year of daily CPUE with daily Moonlight data

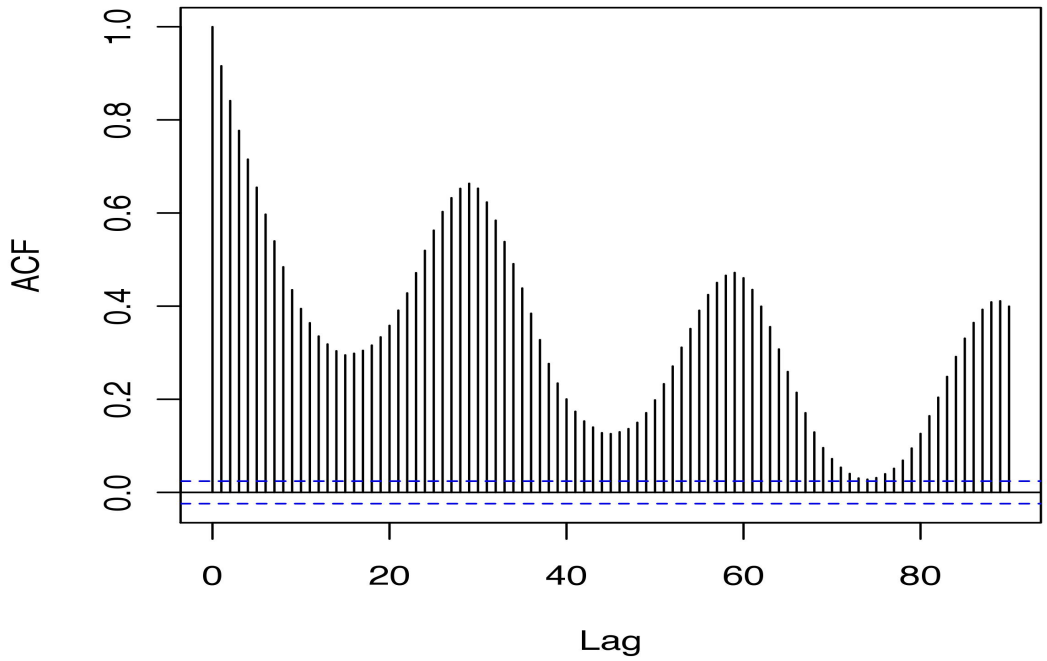


Figure 4.6: Autocorrelation function of daily CPUE data

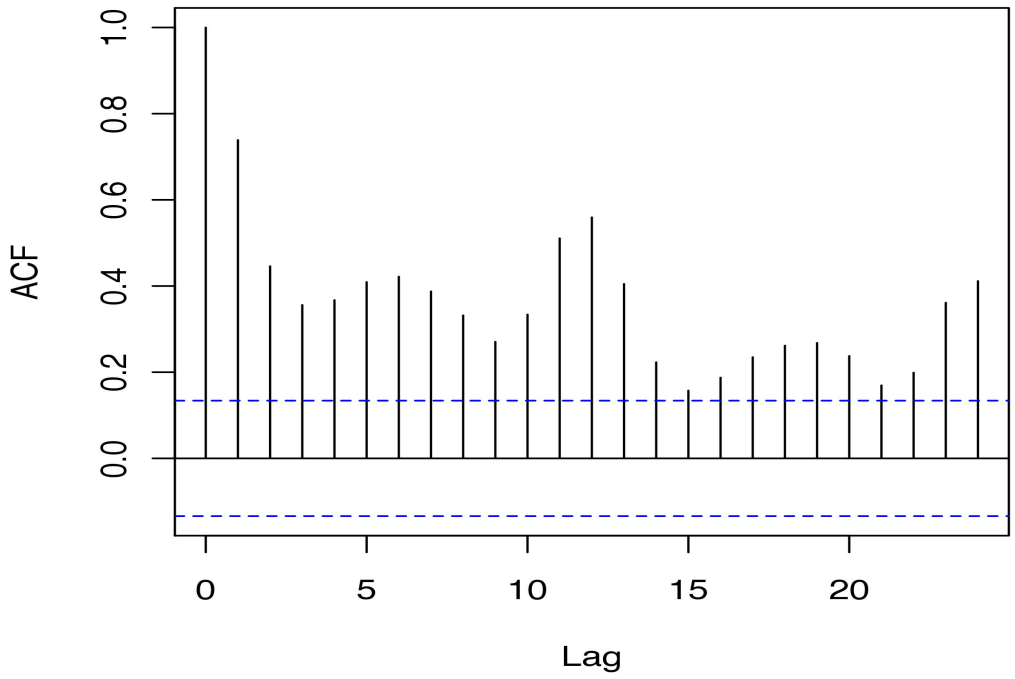


Figure 4.7: Autocorrelation function of monthly averaged CPUE data

From Figure 4.1, there is no clear indication of an overall long-term increasing or decreasing trend in CPUE, but rather changes in the mean level of CPUE with alternating increasing and decreasing trends over different periods of the data. This is confirmed in Figure 4.3(a). An increasing trend is seen from 1987 to 1990, but large decreasing trends are seen from 1990 to 1993 and 1995 to 2002.

Table 4.2: Table showing means, medians and variances for daily data from 1 January 1986 to 31 December 1994 and 1 January 1995 to 31 December 2003 respectively

Dataset	Mean	Median	Variance
01/01/86-31/12/03	43.43	38.92	480.50

Table 4.2 shows the mean, median and variance for the daily CPUE. A large variance is observed for this data, and this may present problems for building models with strong explanatory ability.

The relationships between CPUE and the various covariates described in Table 4.1 are also inspected. In order to compare the patterns of CPUE and covariate data, CPUE and each covariate is overlayed on the same system of axes, where both variables are standardised. From Figure 4.8 a relationship between CPUE and temperature can be seen to exist; the two variables seem to move in the same way with temperature peaking after CPUE. This makes intuitive sense, as CPUE is seen to peak around July/August/September from Figures 4.4(a) and 4.4(b), when temperatures are low, with CPUE decreasing when temperatures peak around October (Madamombe, 2002). Figure 4.9(a) shows no visible relationship between lake level and CPUE. A 3-year plot of lake level, shown in Figure 4.9(b), does however show a yearly cycle in lake level. Overlaying the monthly aggregate of lake level on the monthly aggregate of CPUE (where both these variables are standardised), as seen in Figure 4.9(c), shows that the yearly cycles of lake level and CPUE do not seem to correspond; lake levels peak earlier on average than CPUE. Figure 4.10 does not show any relation between precipitation and CPUE; with the rainy season occurring over October to February (Madamombe, 2010), where precipitation is seen to be high, but CPUE is low. Precipitation appears to be consistently around zero over the winter months when CPUE increases. Figure

4.12 shows no visible relationship between CPUE and cloud cover; even though it was thought that CPUE should increase with decreased cloud cover (and visa versa) as described in Chapter 1. This relationship is however not visibly present.

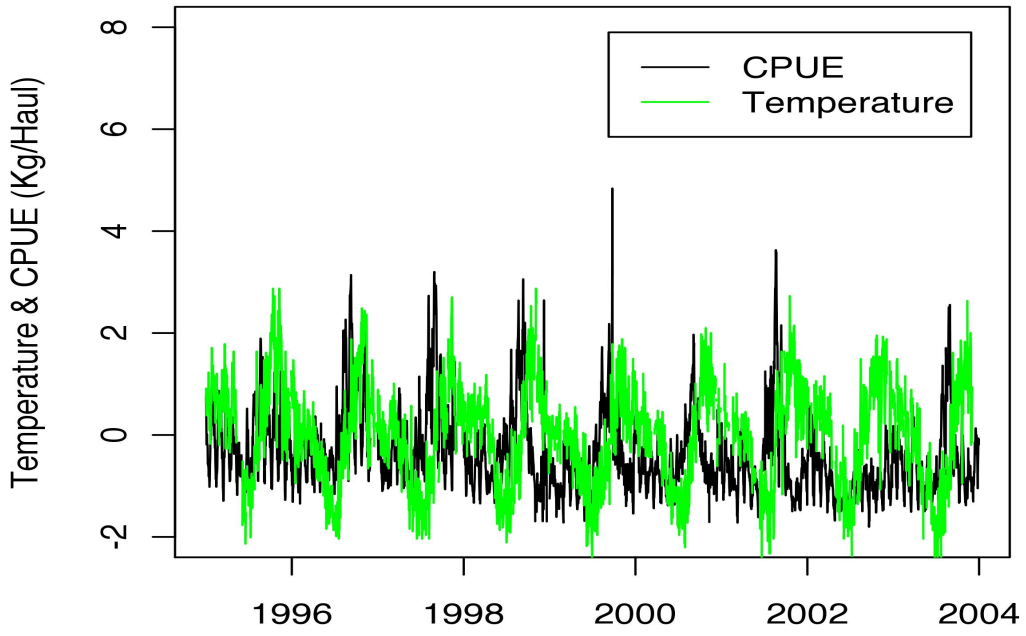


Figure 4.8 Plot showing daily CPUE with daily temperature over 1 January 1995 to 31 December 200

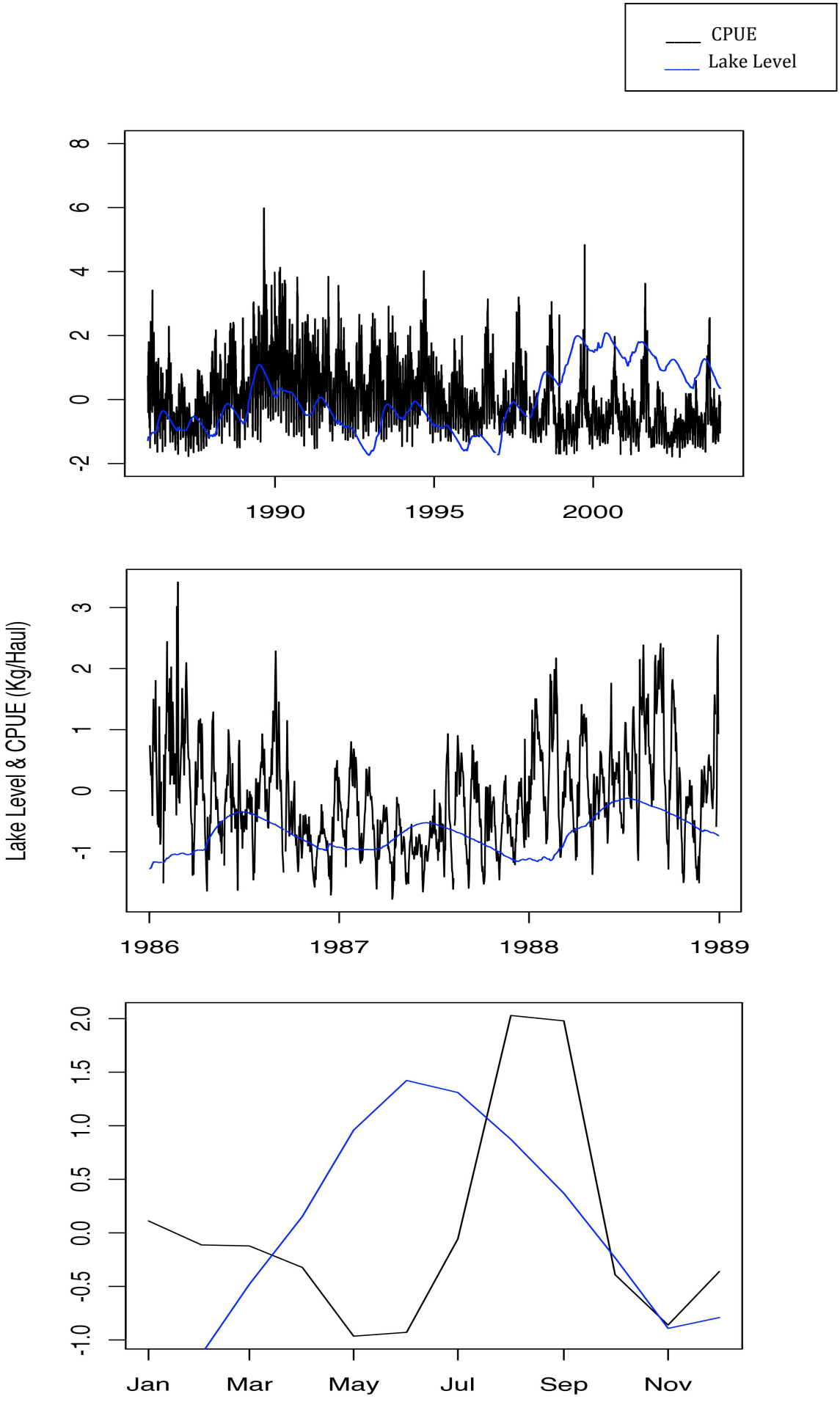


Figure 4.9: (a) Plot showing daily CPUE and daily lake level data over the period 1 January 1986 to 31 December2003.
(b) Plot showing 3 years of daily CPUE and daily lake level data
(c) Plot showing monthly averages of CPUE and lake level data observed over the period 1 January 1986 to 31 December 2003

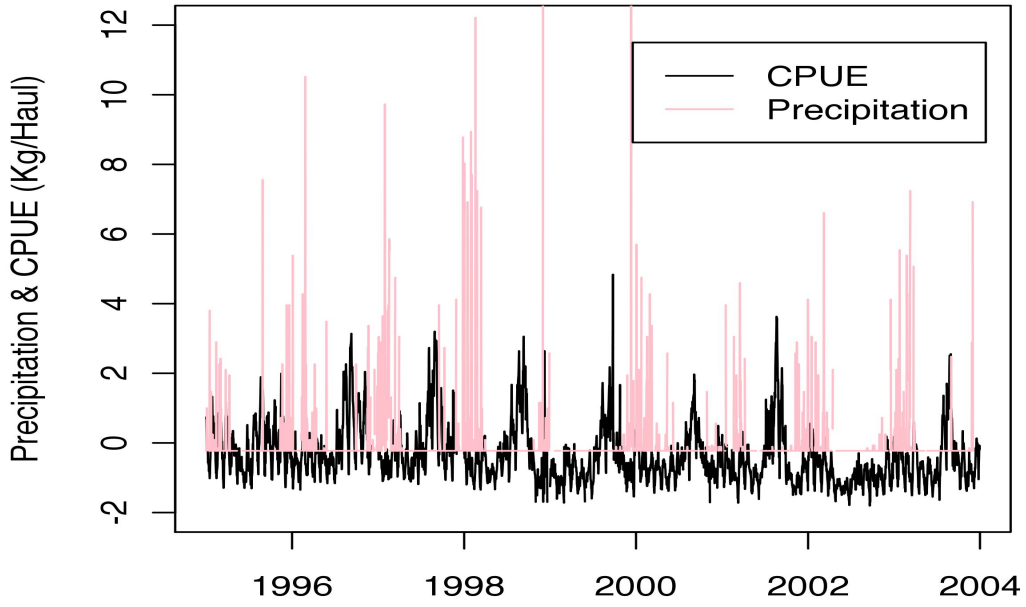


Figure 4.10 Plot showing daily CPUE with daily precipitation over 1 January 1995 to 31 December 2003

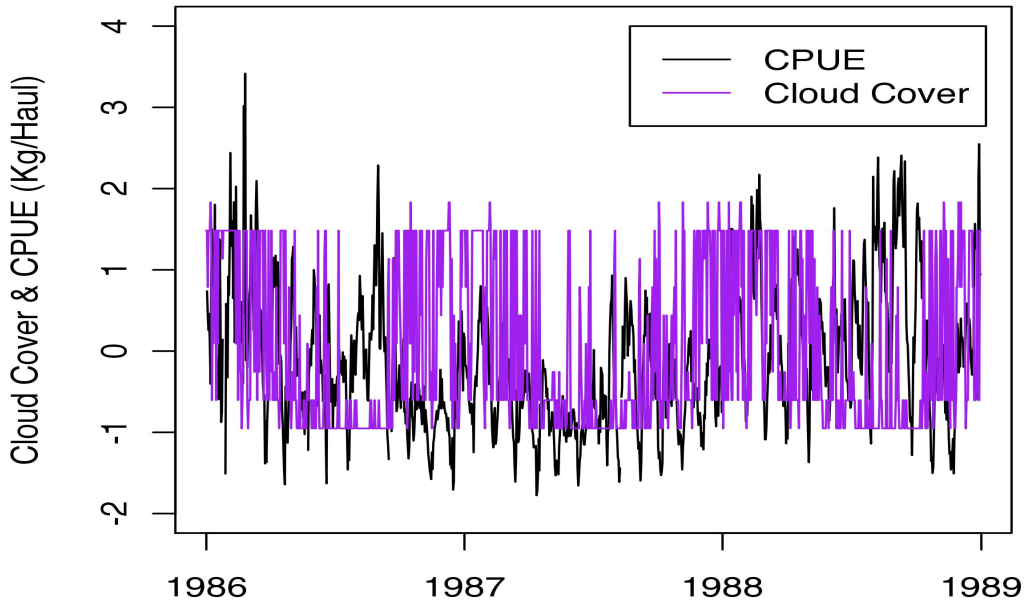


Figure 4.11 Plot showing daily CPUE with daily cloud cover over 1 January 1995 to 31 December 2003

4.4 Model Hypotheses

Chapter 1 introduced biological and climatic information about Lake Kariba and how these factors are thought to interact with the Kapenta fishing stocks. Additionally, having performed the above exploratory analysis with the available CPUE and covariate data, hypotheses about which components will be important in determining daily CPUE are developed. The following factors are addressed one at a time and their hypothesized significance into the model discussed:

- Yearly cyclical component: The movement of Kapenta fish to protected bays over summer months for breeding, and back into open waters over winter months, when Kapenta CPUE are seen to increase, is hypothesized to be a significant component in the model for CPUE. This yearly seasonal component is predominantly seen in Figures 4.1, 4.3(a)-(c), 4.4 and 4.7.
- Monthly cyclical component: Figures 4.2 and 4.6 show the clear presence of monthly cyclical components throughout the Kapenta CPUE data. The presence of this component is attributed to the effect of moonlight on CPUE, where the cycle of the moon is 29.53 days, Figure 4.5 shows how closely daily CPUE relates to moonlight. Kapenta fishing takes place in the evenings with light attractions. When moon phase is close to full moon and the moon emits most light, these light attractions loose efficacy and CPUE decreases. The strong presence of this component, due to moonlight, is hypothesized to be a significant component in the model for CPUE.
- Lake Level: Other literary works, such as Ndebele-Murisa et al (2011), Marshall (1982) and Magadza (1996), find lake level to be significant factors in determining Kapenta stocks. Increased water levels result in increased nutrient concentrations in the water, increasing phytoplankton biomass and production. This has favorable impacts on higher trophic levels, such as the Kapenta fish. This thesis, however, aims to understand factors that influence frequently sampled (daily) CPUE values, and due to the fact that lake level values change slowly and only very slightly, it is hypothesized that lake level will not be a significant component in the

model for Kapenta CPUE. Additionally Figures 4.9(a) to 4.9(c) show no relationship between the movement of CPUE with lake level, and the cyclical movement of lake level does not correspond to the cyclical movement of CPUE.

- Temperature: Literary works such as Ndebele-Murisa et al. (2011) and Chifamba (200) show temperature to influence Kapenta stocks. Increased water temperatures adversely affect fish stocks, as higher temperatures allow more nutrients to become locked up in the bottom layer of the lake. This adversely affects the phytoplankton in the upper layer of the lake upon which Kapenta feed, and this decreases Kapenta stocks. Figure 4.8 shows an existing relationship between temperature and Kapenta CPUE, with Kapenta catches decreasing as soon as temperatures peak. However, since the yearly cyclical component of CPUE corresponds so closely to the yearly cycle of temperature, much of the temperature effect will most likely be captured by the yearly cyclical component. However, a temperature covariate may still be significant in explaining CPUE values corresponding to unusual changes in temperature. It is thus uncertain as to whether this component will be significant in the CPUE model.
- Precipitation: Ndebele-Murisa et al. (2011) found precipitation to influence Kapenta stocks, but not as greatly as the temperature and lake level covariates. In a similar method to lake level, precipitation favorably affects Kapenta stocks. Figure 4.10 shows no visual relationship between the two variables, and similarly with lake level shows slow and only slightly changing data, not likely to affect frequently sampled (daily) CPUE values. Thus precipitation is not hypothesized to affect CPUE values.
- Cloud Cover: Due to the fact that moonlight had such a clear relationship with CPUE, it was hypothesized that cloud cover over the evenings could similarly affect Kapenta catch. Dense cloud cover can diminish the amount of moonlight and favorably affect CPUE, as the efficacy of light attractions are improved. Figure 4.11 shows no clear relationship between CPUE and cloud cover, but the presence of this component will be reliably tested in the model for CPUE

4.5 Building Structural Time Series Models For Kapenta Catch Data

The model building process employed in this thesis is a step-by-step process described in Section 4.5.1 The form of the structural time series model is first selected through adding models components, such as trend, cyclical and regression components, into the model one at a time and assessing how the addition of these factors affect model diagnostics and measures of fit. These diagnostics and measures of fits are discussed in Section 3.6. The goodness-of-fit measures include AIC, prediction error variance (P.E.V) and R_D^2 values, but note that there are two AIC values provided, AIC_1 and AIC_2, which are described in Section 3.9.6. The unknown parameters of a specified model are calculated through the maximum likelihood algorithms discussed in Section 3.5, and filtering and smoothing algorithms are performed as described in Section 3.3. Bayesian analysis can alternatively be used to estimate unknown parameters, but this will be demonstrated in Section 4.8.

Structural time series models are fitted to the CPUE data through the `dlm` package described in Section 3.9, using code seen in Appendices A to E. The choice between modelling CPUE and total catch was not an easy choice, as total catch at times were seen to provide stronger R_D^2 values than those built using CPUE data. The choice to model CPUE data ultimately was made, as the number of hauls per day needs to be considered. CPUE is the standard means of considering this variable, and amongst others is a measure used by Ndeldele-Murisa et al. (2011) and Chifamba (2000). Ndeldele-Murisa et al. (2011) describes CPUE as being a better indicator of the actual fish stocks in the lake as the total Kapenta catches depend on the number of fishermen operating at a given time. Similarly, this thesis intends to eliminate the effects of different human efforts on a given day, and focus more on biological factors affecting Kapenta catch. Additionally, diagnostic measures, such as the autocorrelation functions and QQ-plots, were observed to be much poorer overall when building models using total catch data. The assumption of normally distributed errors was particularly poorly upheld when using total catch data. It was a struggle to explain all the variation in total

catch data and more harmonics would have to be added to all cyclical components in an attempt to remove autocorrelation between lags than seen with CPUE data.

Having decided to model CPUE, it was further decided to model the log of CPUE and then to model the two halves of the log of CPUE data separately. The two halves of the data, the periods 1 January 1986 to 31 December 1994 and 1 January 1995 to 31 December 2003, will be referred to as datasets I and II respectively throughout this thesis. Datasets I and II were modelled separately for two reasons: firstly, graphically it appears that the two halves of the data might follow different underlying patterns, as seen in Figure 4.1. Secondly, temperature and precipitation data from Tutiempo weather only become consistent or regular over the second half of the data and their relationship with CPUE can only be reliably tested over dataset II. It was decided to model the log of CPUE, as this normalizes the data and removes the positive skew present in the CPUE data. This is demonstrated in Figures 4.12(a) to 4.12(d). Subsequently, modelling the log of CPUE data continuously allowed for models with more favorable R^2_D values. Diagnostic autocorrelation and QQ-plots also appeared more favorable when modelling the log of datasets I and II. Note that when this thesis refers to modelling CPUE data, it is referring to the log of CPUE data. The model building process for both CPUE datasets I and II, is described in more detail in Section 4.5.1 below.

4.5.1 The Model Building Process

The process of building a model describing daily CPUE, for datasets I and II, involves the following:

1. The structure of the model is decided on. This involves selecting which and in what form trend, cyclical and regression components will be present in the model for CPUE. This in itself is a 4-step process:

   i) The structure of the trend component is first decided on. This includes deciding between modelling the trend as a *local level model*, a *local linear trend model*, a *deterministic model*, a *smooth trend model* or a *random walk with drift model*. The various ways of modelling these trend components are described in

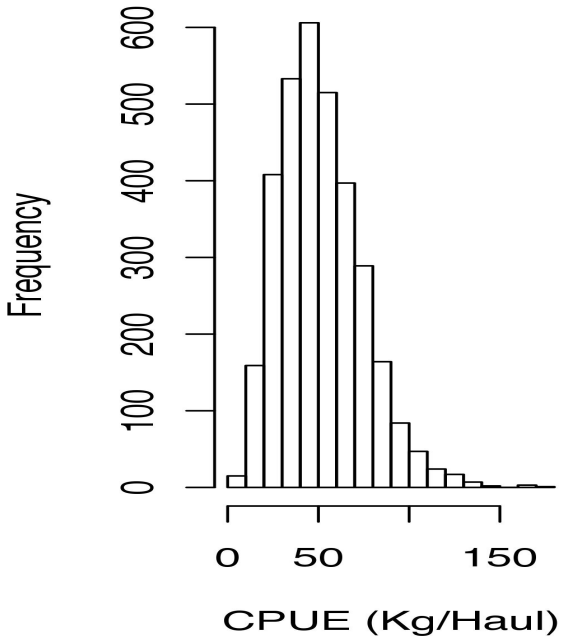


Figure 4.12 (a): Histogram of CPUE over dataset I

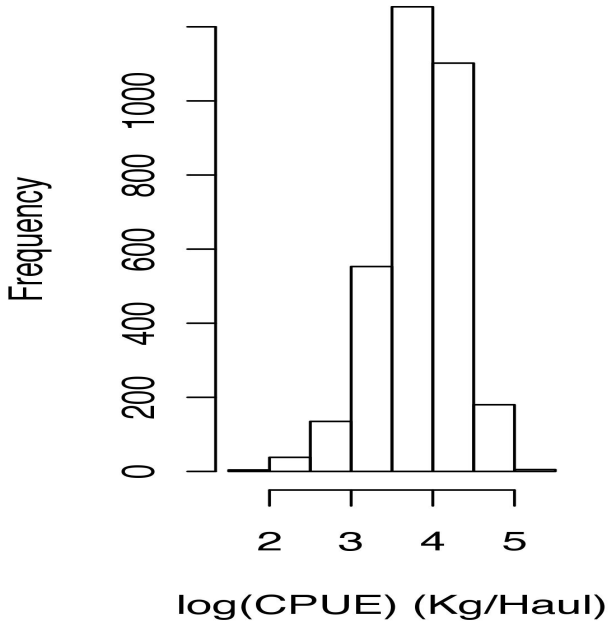


Figure 4.12 (b): Histogram of logged CPUE over dataset I

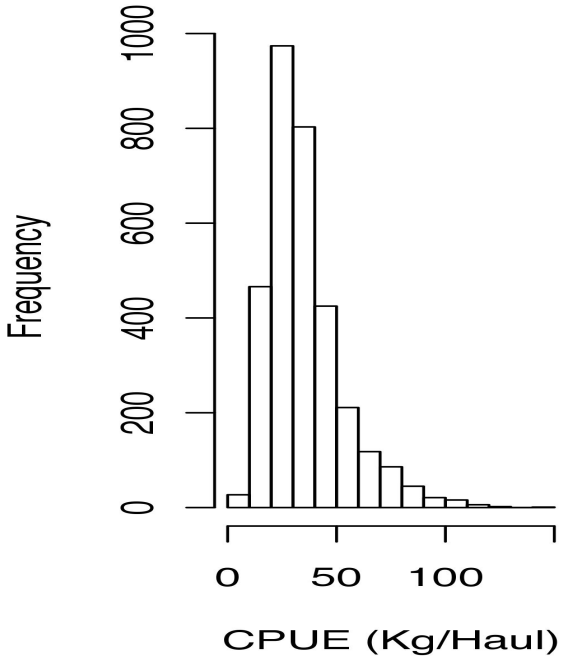


Figure 4.12 (c): Histogram of CPUE over dataset II

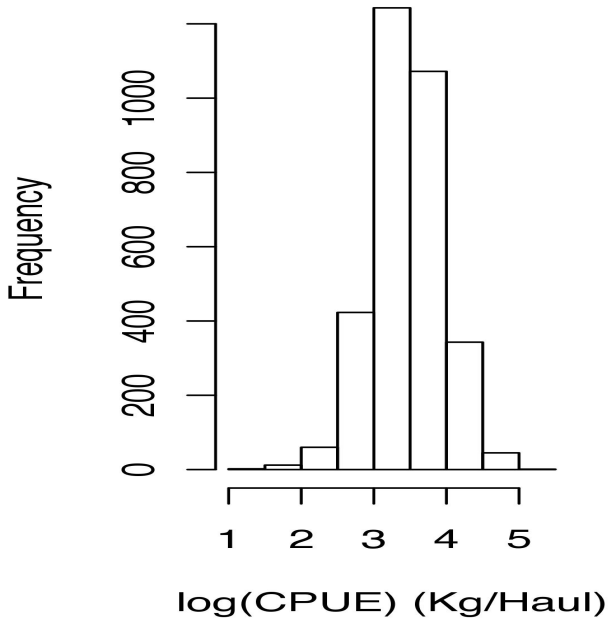


Figure 4.12 (d): Histogram of logged CPUE over dataset II

equations (3.7.3) to (3.7.6) in Section 3.7.1.1, but for convenience are restated and renumbered:

Table 4.3: Table showing formulae for trend components of structural time series models

Local Level Model	$y_t = \mu_t + \varepsilon_t, \quad \varepsilon_t \sim N(0,\sigma_\varepsilon^2)$ $\mu_t = \mu_{t-1} + \xi_t, \quad \xi_t \sim N(0,\sigma_\xi^2)$	(4.5.1)
Local Linear Trend Model	$y_t = \mu_t + \varepsilon_t, \quad \varepsilon_t \sim N(0,\sigma_\varepsilon^2)$ $\mu_t = \mu_{t-1} + l_{t-1} + \xi_t, \quad \xi_t \sim N(0,\sigma_\xi^2)$ $l_t = l_{t-1} + \zeta_t, \quad \zeta_t \sim N(0,\sigma_\zeta^2)$	(4.5.2)
Deterministic Model	$y_t = \mu_t + \varepsilon_t, \quad \varepsilon_t \sim N(0,\sigma_\varepsilon^2)$ $\mu_t = \mu_0 + lt,$	(4.5.3)
Random Walk with Drift	$y_t = \mu_t + \varepsilon_t, \quad \varepsilon_t \sim N(0,\sigma_\varepsilon^2)$ $\mu_t = \mu_{t-1} + l + \xi_t, \quad \xi_t \sim N(0,\sigma_\xi^2)$	(4.5.4)
Smooth Trend Model	$y_t = \mu_t + \varepsilon_t, \quad \varepsilon_t \sim N(0,\sigma_\varepsilon^2)$ $\mu_t = \mu_0 + l_t,$ $l_t = l_{t-1} + \zeta_t, \quad \zeta_t \sim N(0,\sigma_\zeta^2)$	(4.5.5)

ii) The effect of moonlight on CPUE is investigated. This effect can be accounted for through adding in moonlight as an explanatory variable, or through modelling it as a cyclical component with period equal to 29.53 days. Modelling an explanatory variable is described in Section 3.7.1.3, but is restated and renumbered here for convenience:

Table 4.4: Table showing formulae for regression components of structural time series models

Explanatory Variable	$y_t = \sum_{j=1}^{k} \beta_{j,t} X_t + \varepsilon_t,$ $\beta_{j,t} = \beta_{j,t-1} + \chi_{j,t},$ $\varepsilon_t \sim N(0,\sigma_\varepsilon^2)$ $\chi_{j,t} \sim N(0,\sigma_\chi^2)$	(4.5.6)

Modelling a cyclical component is described in equations (3.7.8), but is also restated and renumbered here for convenience

Table 4.5: Table showing formulae for cyclical components of structural time series models

Fourier Form Cyclical Components	$y_t = c_t + \varepsilon_t,$ $c_t = c_{t-1}\cos\lambda_c + c^*_{t-1}\sin\lambda_c + \tilde{\omega}_t,$ $c^*_t = -c_{t-1}\sin\lambda_c + c^*_{t-1}\cos\lambda_c + \tilde{\omega}^*_t,$ $\varepsilon_t \sim N(0,\sigma^2_\varepsilon)$ $\tilde{\omega}_t \sim N(0,\sigma^2_{\tilde{\omega}})$ $\tilde{\omega}^*_t \sim N(0,\sigma^2_{\tilde{\omega}^*})$	(4.5.7)

The benefit of using a cyclical component is that adding smaller harmonics of the cycle to the model is easily performed and showed to improve model fit, diagnostics and flexibility of the model. Alternatively, adding increasing powers of the explanatory variable to the model performs a similar function. Both methods were tested in this thesis and the complexity of harmonics was restricted to four harmonics fitted per cycle. Improving model fit, diagnostics and flexibility can also be performed through making the cyclical components or explanatory variables dynamic, where the error terms in equations (4.5.6) and (4.5.7) are then greater than zero. The decision of whether or not to model this explanatory variable or cyclical component of moonlight dynamically was a difficult choice. This component may change over time, especially as the time series is long and behaviour changes. However the danger is that with very variable data, such as the Kapenta CPUE data, dynamic components may absorb much of the variability and then loose it's simple interpretation. Based on visual inspection of graphical representations of fitted model terms for various components considered for inclusions into the model, including both cyclical and regression components, this thesis decided not to model any components dynamically. It was too frequently observed that components absorb too much of the variability in the data, where after these components made no interpretive or biological sense. As an example of this, models including dynamic components will be demonstrated in Section 4.6.

iii) The effect of the yearly cycle is fitted to the data, where the period of the cycle is equal to 365 days. The number of harmonics in this cycle can also be increased to improve model fit and diagnostics. For the same reasoning

explained in the modelling of the moonlight effect in (2), the yearly cyclical components are not modelled dynamically, as the same problems discussed where experienced.

iv) Explanatory variables temperature, lake level and precipitation are added to the model to see if these variables help explain any of the variation in the data. Note that the variables temperature and precipitation are only added to dataset II due to data availability. Again, for the same reasoning explained in the modelling of the moonlight effect in (2), the above explanatory variables are not modelled as dynamic components.

2. The models defined in (1.) are essentially in state space form, as described in Section 3.7.1.4. The Kalman filtering and smoothing algorithms are used to estimate the mean and covariance matrix of the normally distributed state vector α_{t+1} based on information through to time *t*. The Kalman filter estimates this based on observations through to time *t*, while the smoothing algorithm estimates are based on the entire data sample. Therefore the Kalman filter estimates $a_{t+1} = E(\alpha_{t+1} \mid y_{1:t})$ and $P_{t+1} = \mathrm{var}(\alpha_{t+1} \mid y_{1:t})$, while the smoother estimates $\hat{\alpha}_t = E(\alpha_t \mid y_{1:n})$ and $V_t = \mathrm{var}(\alpha_t \mid y_{1:n})$. The filtering and smoothing algorithms are discussed in Section 3.3, but for convenience the recursions are restated below:

Table 4.6: Table showing formulae for Kalman filtering and smoothing algorithms

Kalman Filter	$v_t = y_t - Za_t,$ $F_t = ZP_tZ' + H,$ $K_t = TP_tZ'F_t^{-1},$ $L_t = T - K_tZ,$ $a_{t+1} = Ta_t + K_tv_t,$ $P_{t+1} = TP_tL_t' + Q,$	(4.5.9)
Smoother	$r_{t-1} = Z'F_t^{-1}v_t + L_t'r_t,$ $\hat{\alpha}_t = a_t + P_tr_{t-1}$ $N_{t-1} = Z'F_t^{-1}Z + L_t'N_tL_t,$ $V_t = P_t - P_tN_{t-1}P_t$	(4.5.10)

2. Unknown model parameters are estimated using maximum likelihood, as described in Sections 3.5.2 and 3.5.3. The loglikelihood is defined as defined as follows:

Table 4.7: Table showing formulae for loglikelihood of structural time series models

Loglikelihood	$\log L(y) = -\frac{(n-d)N}{2}\log 2\pi - \frac{1}{2}\sum_{t=d+1}^{n}\log\lvert F_t\rvert - \frac{1}{2}\sum_{t=d+1}^{n} v_t' F_t^{-1} v_t$ where d is the number of diffuse elements, n is the number of observations in the time series, and N the number of series observed	(4.5.8)

The loglikelihood is maximised by means of a numerical maximisation process, the "L-BFGS-B" method, described in Section 3.5.4.

4. Once the model is fully specified model diagnostics are assessed through the use of autocorrelation functions, scatter plots and QQ-plots, to assess the assumptions of independent, homoscedastic and normally distributed residuals. Additionally, goodness-of-fit measures and measures for model comparison are constructed. Goodness-of-fit measures include the prediction error variance (P.E.V) and R_D^2 measures, while models are compared using AIC measures. These measures are discussed in more detail in Sections 3.6.1 to 3.6.3, but for convenience purposes, goodness-of-fit measures and model-comparison measures are restated below

Table 4.8: Table showing formulae for Goodness-of-Fit Measures for Structural Time Series Models

P.E.V	$\sigma^2 = \sigma_*^2 \bar{f}$ where $\bar{f} = \lim_{t\to\infty} f_t$,	(4.5.9)
R^2	$R_D^2 = 1 - SSE / \sum_{t=2}^{n} (\Delta y_t - \overline{\Delta y})^2$, where $SSE = (n-s)\tilde{\sigma}^2$, s is the number of constant elements in state vector, n is the number of observations in the time series	(4.5.10)
AIC	$AIC = -2\log L(y \mid \psi) + 2w$, $w=d+q$, d is the number of diffuse elements in the state vector and q is the total number of elements in the state vector	(4.5.11)

Note the AIC calculated using loglikelihood described in (4.5.8) produces the

AIC_1 measure, whereas using the loglikelihood in (3.9.1) produces the AIC_2 measure. The diagnostics and goodness-of-fit measures for each of the models constructed using the process described in (1.) are viewed, and the model describing the most variation within the CPUE data is selected.

4.6 Results for Structural Time Series Models using Daily CPUE Data

The model building process described in Section 4.5.1 is summarised in Tables 4.9 and 4.10, for datasets I and II respectively. Modelling the trend component is first considered, and it appears from fit values that the *local level model* performs best for both datasets. Goodness-of-fit values show lower AIC_1, AIC_2, P.E.V and higher R_D^2 values. Alternative models for the trend component are a *local linear trend, smooth trend, random walk with drift* or *deterministic trend* model. These all incorporate a slope component into the trend. The smoothed slope estimates (the estimation of the slope component in the state vector, given the entire time series) shown in Figures 4.12 and 4.13 for datasets I and II respectively, show that in these models for trend, a slope component may be unnecessary, and the initial indication to use a local linear model appears satisfactory. For models containing a slope, smoothed slope estimates are approximately all zero. The *smooth trend model* shows smoothed slope estimates slightly greater in value than those observed in the other kinds of models for trend, however this is expected as all the variance is transferred into the slope component. Additionally, the value of the slope component is not strictly in one direction, but oscillates around zero, suggesting *a local linear model* is still satisfactory. The *deterministic trend* model is useful in the way that it allows one to see the overall direction of CPUE through the smoothed mean level component of the trend. For dataset I, the *deterministic trend model* shows that the mean level of CPUE is increasing overall. For dataset II, however, the *deterministic trend model* shows that the mean level of CPUE is decreasing overall. The presence of an overall increasing or decreasing mean level of CPUE indicates that perhaps a small value for slope may be meaningful, as even though this increase or decrease appears small per week, over 9 years it may prove to be significant. If there is an indication that a slope component, although

Table 4.9: Table showing model building process and fit results using daily data for dataset I

	Model	**AIC_1**	**AIC_2**	**P.E.V**	**R^2_D**
Trend	*Local Level Model*	-6444.69	-6428.58	0.051	-0.015
	Smooth Trend Model	-5541.76	-5509.54	0.067	-0.334
	Deterministic Trend Model	-1660.75	-1628.51	0.219	-3.358
	Random Walk Model with Drift Model	-6431.57	-6399.34	0.051	-0.015
	Local Linear Trend Model	-6429.57	-6397.34	0.051	-0.015
Moon Cycle	*Local Level Model + Explanatory Variable (Moonlight)*	-6655.49	-6623.26	0.048	0.051
	Local Level Model + Explanatory Variable (Moonlight^2)	-6853.91	-6805.55	0.045	0.107
	Local Level Model + Explanatory Variable (Moonlight^3)	-6924.90	-6860.43	0.044	0.127
	Local Level Model + Explanatory Variable (Moonlight^4)	-6974.99	-6894.39	0.043	0.140
	Local Level Model + Cycle (period=29.53, harmonics=1)	-6666.69	-6618.34	0.048	0.056
	Local Level Model + Cycle (period=29.53, harmonics=2)	-6865.43	-6784.84	0.044	0.117
	Local Level Model + Cycle (period=29.53, harmonics=3)	-6922.00	-6809.18	0.043	0.137
	*Local Level Model + Cycle (period=29.53, harmonics=4)***	-6952.35	-6807.29	0.043	0.151
Yearly Cycle	*Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=1)*	NA	-6576.98	0.048	0.055
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=1)	NA	-6742.85	0.045	0.116
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=1)	-6958.39	-6766.96	0.044	0.136
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=1)	-6986.49	-6764.96	0.043	0.150
	Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=2)	NA	-6534.09	0.048	0.054
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=2)	NA	-6699.93	0.045	0.115
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=2)	NA	-6724.03	0.044	0.136
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=2)	NA	-6722.03	0.043	0.149
	Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=3)	NA	-6489.18	0.048	0.053
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=3)	NA	-6654.72	0.045	0.114
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=3)	NA	-6678.73	0.044	0.135
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=3)	NA	-6676.69	0.044	0.149
	Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=4)	NA	-6442.48	0.048	0.052
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=4)	NA	-6607.00	0.045	0.113
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=4)	NA	-6631.23	0.044	0.134
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=4)	NA	-6629.09	0.044	0.148
Explanatory Variables	*Local Level Model + Cycle (period=29.53, harmonics=4)+Explanatory Variable (Lake Level)*	-6959.41	-6783.07	0.043	0.151
Dynamic Models	*Local Level Model +Dynamic Explanatory Variable (Moonlight)*	-7017.28	-6985.05	0.025	0.051
	Local Level Model + Dynamic Cycle (period=29.53, harmonics=1)	-6666.69	-6618.34	0.048	0.056

Table 4.10: Table showing model building process and fit results using daily data for dataset II

	Model	**AIC_1**	**AIC_2**	**P.E.V**	**R^2_D**
Trend	*Local Level Model*	-6770.75	-6754.64	0.044	-0.026
	Smooth Trend Model	-5934.26	-5902.02	0.057	-0.325
	Deterministic Trend Model	-1845.63	-1813.39	0.205	-3.765
	Random Walk Model with Drift Model	-6757.27	-6725.04	0.044	-0.027
	Local Linear Trend Model	-6755.27	-6723.04	0.044	-0.026
Moon Cycle	*Local Level Model + Explanatory Variable (Moonlight)*	-6942.18	-6909.94	0.042	0.029
	Local Level Model + Cycle (period=29.53, harmonics=1)	-6945.51	-6897.15	0.042	0.033
	Local Level Model + Cycle (period=29.53, harmonics=2)	-7070.69	-6990.11	0.040	0.075
	Local Level Model + Cycle (period=29.53, harmonics=3)	-7083.66	-6970.84	0.039	0.085
	*Local Level Model + Cycle (period=29.53, harmonics=4)***	-7069.91	-6924.85	0.039	0.087
Yearly Cycle	*Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=1)*	NA	-6855.90	0.042	0.032
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=1)	NA	-6948.67	0.040	0.075
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=1)	-7101.34	-6929.36	0.039	0.084
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=1)	-7101.33	-6883.37	0.039	0.086
	Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=2)	NA	-6815.23	0.042	0.032
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=2)	NA	-6908.60	0.040	0.075
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=2)	NA	-6889.50	0.039	0.085
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=2)	NA	-6843.54	0.039	0.087
	Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=3)	NA	-6771.73	0.042	0.032
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=3)	NA	-6865.65	0.040	0.075
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=3)	NA	-6846.70	0.039	0.085
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=3)	NA	-6800.78	0.039	0.087
	Local Level Model + Cycle (period=29.53, harmonics=1)+Cycle (period=365, harmonics=4)	NA	-6724.20	0.042	0.031
	Local Level Model + Cycle (period=29.53, harmonics=2)+Cycle (period=365, harmonics=4)	NA	-6817.65	0.040	0.074
	Local Level Model + Cycle (period=29.53, harmonics=3)+Cycle (period=365, harmonics=4)	NA	-6798.61	0.039	0.084
	Local Level Model + Cycle (period=29.53, harmonics=4)+Cycle (period=365, harmonics=4)	NA	-6752.67	0.039	0.086
Explanatory Variable	*Local Level Model + Cycle (period=29.53, harmonics=4)+Explanatory Variable (Lake Level)*	NA	-6900.49	0.039	0.087
Dynamic Models	*Local Level Model +Dynamic Explanatory Variable (Moonlight)*	-7295.24	-7263.00	0.041	0.052
	Local Level Model + Dynamic Cycle (period=29.53, harmonics=1)	-6945.51	-6897.15	0.041	0.033

small, is significant, it will be added to the model, for increased flexibility. For this purpose confidence intervals for the smoothed slope estimates in the models for dataset I and II were constructed. In order to construct these confidence intervals, yearly cyclical, monthly cyclical and covariate effects are removed from the data, by incorporating these components into the model.

Confidence Intervals assessing the presence of slope components for datasets I and II were tested for models constructed with *local linear trends*. This is the most flexible form for modelling trend with a slope component, and with cyclical and covariate effects provided the best overall model diagnostics and fits from all trend models containing a slope. Additionally, confidence intervals were constructed for differences in the levels of the trend components for these models; the difference between each smoothed trend level estimate and the final smoothed trend level estimate in the time series. The variance of the slope components for both datasets I and II are estimated to be approximately zero, and thus the confidence intervals remain constant over time.

Table 4.11: Confidence Intervals for Slope Components with Daily CPUE data

Dataset I	0.0000 ± 0.0044
Dataset II	0.0000 ± 0.0039

These confidence intervals show slope components to be unnecessary, with the intervals containing zero. The confidence intervals showing differences in the mean trend level, seen in Figures 3.14 and 3.15, show these differences to oscillate from being significantly negative to significantly positive for both datasets; more indicative of a random walk type model. The prior belief of local level models is thus confirmed, and the trend for both datasets is modelled as such. Note that cycles are still present in smoothed trend level differences, as all models despite the number of harmonics added to cyclical components could not capture much of the variation of the within the data. Much more variation could be explained using *local linear models*.

Tables 4.9 and 4.10 show that the effect of moonlight on CPUE is better accounted for through adding a cyclical component with period equal to 29.53

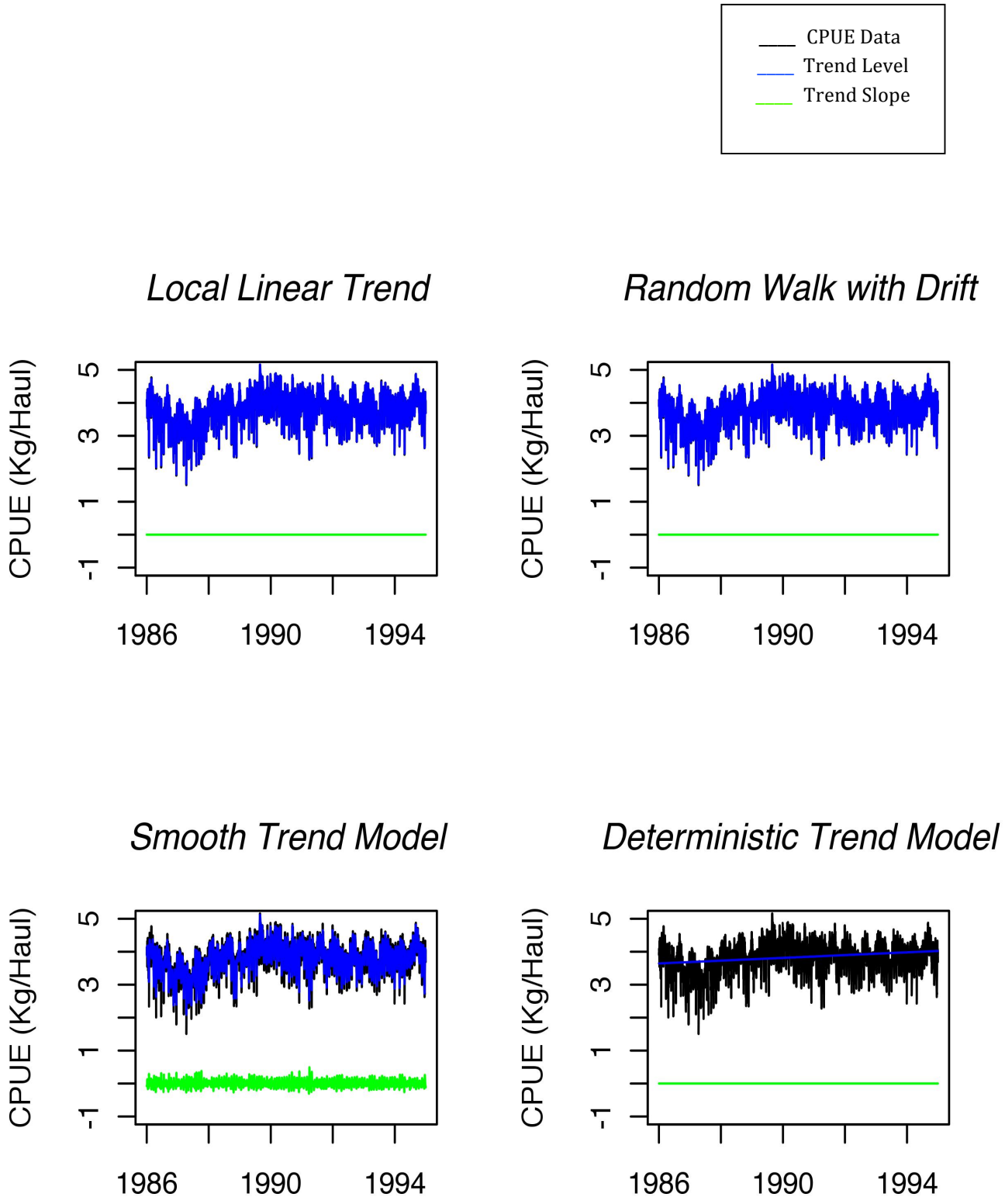


Figure 4.13: Plots showing daily CPUE data with smoothed estimates of trend level and slope components, for the various model structures shown above for dataset I

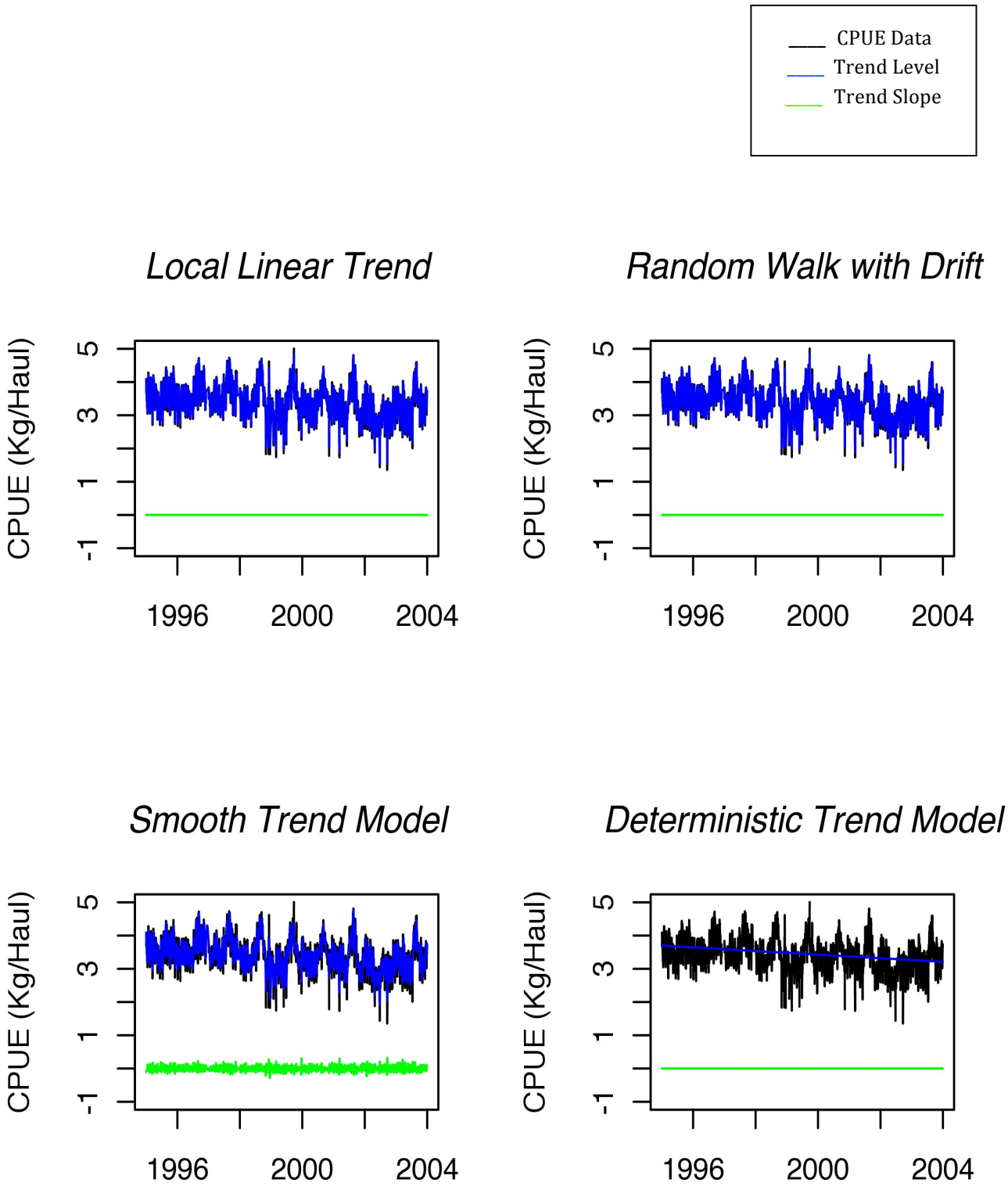


Figure 4.14: Plots showing daily CPUE data with smoothed estimates of trend level and slope components, for the various model structures shown above for dataset II

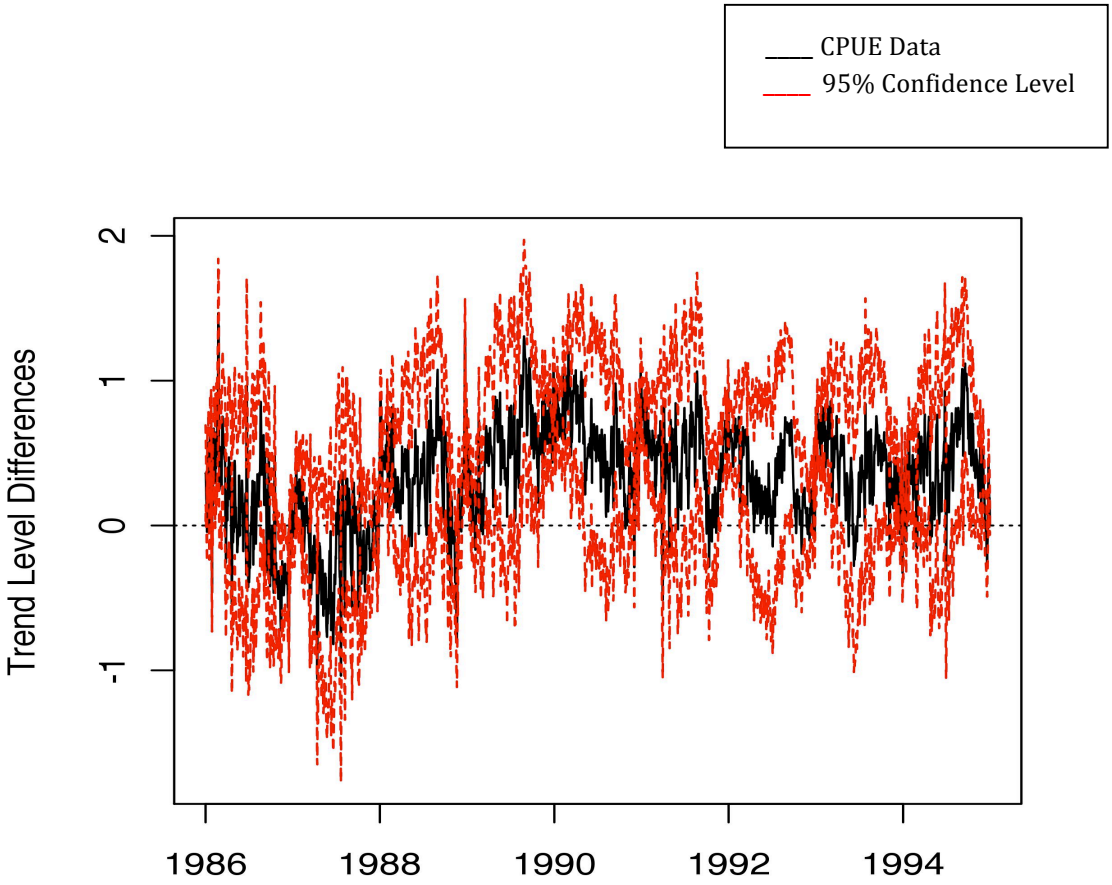


Figure 4.15: Plot showing confidence interval for the changes in trend level for a model using daily data from dataset II

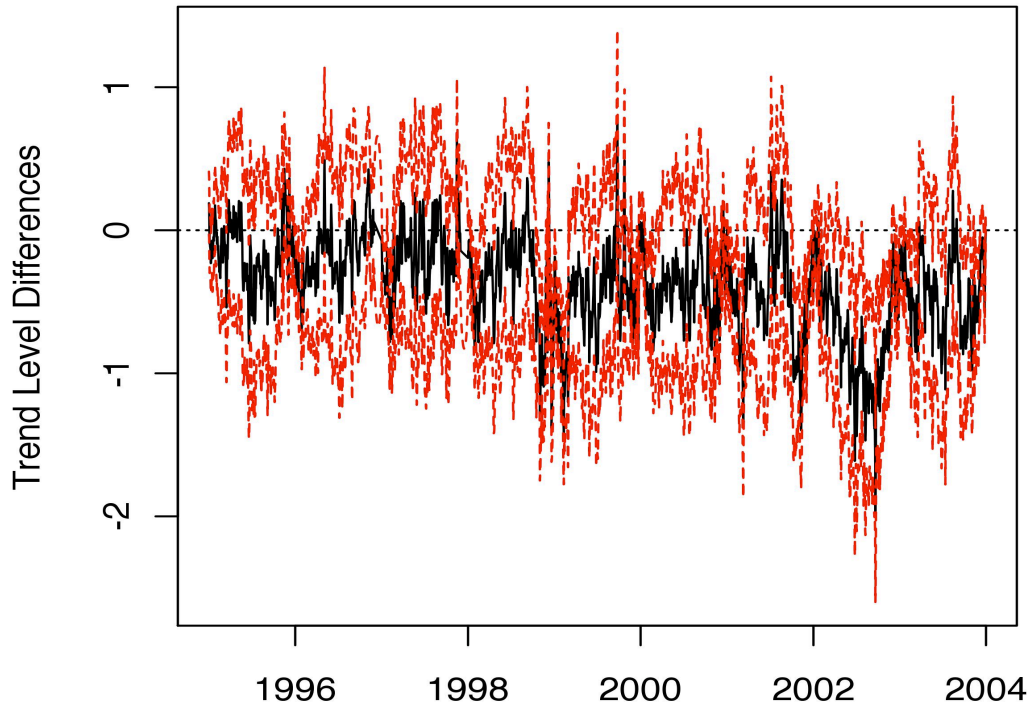


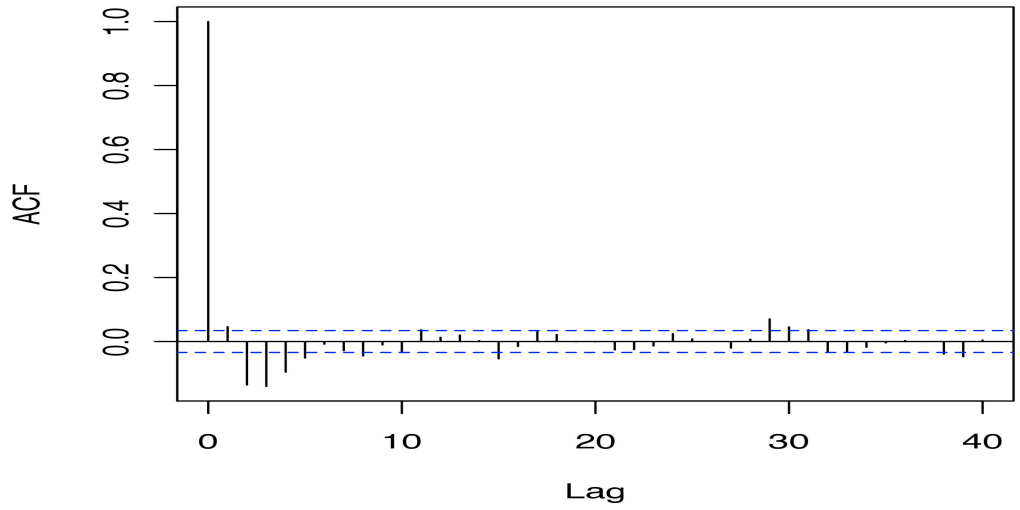
Figure 4.16: Plot showing confidence interval for the changes in trend level for a model using daily data from dataset II

days, than adding an explanatory variable in both datasets. Increasing the number of harmonics in the cycle further improves model fits, more so than adding increasing powers of the moonlight explanatory variable to the model. Thus the moonlight component is better modelled through sines and cosines.

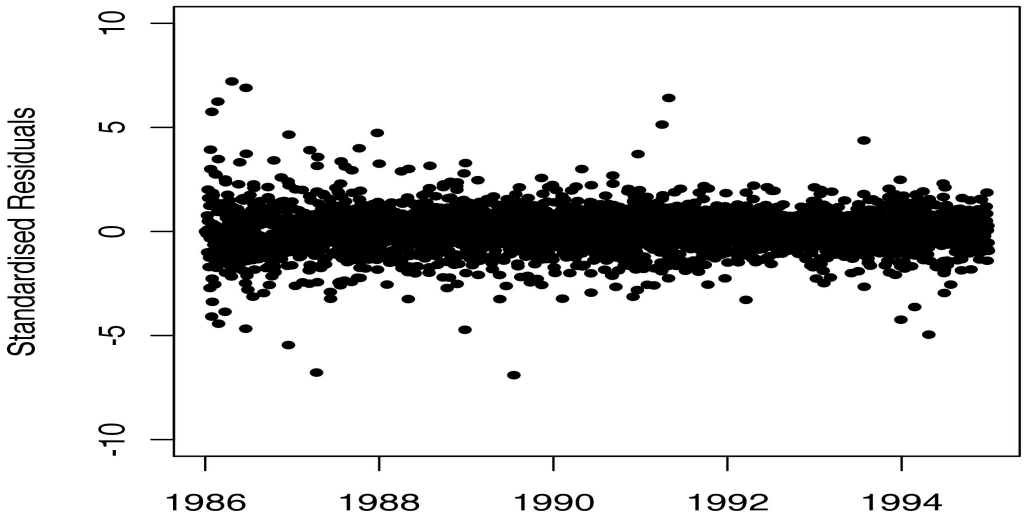
Cyclical components with period equal to 365 days are also added to the model to capture the yearly movement of Kapenta fish in and out of protective bays for breeding. This component is also modelled with increasing harmonics. Explanatory variables temperature and precipitation are tested for inclusion into the model for dataset II (due to data availability), but these produced non-interpretable results. Due to the fact that many daily values were missing from the temperature and precipitation datasets, the CPUE dataset had to be altered to have corresponding missing values with that of the covariate data (§3.9.3). This makes directly comparing these models with those not including the explanatory variables impossible. Additionally, models including temperature and/or precipitation with this altered CPUE dataset, produced results with extreme negative R_D^2 values, and large P.E.V, AIC_1 and AIC_2 values. Graphical diagnostics, the autocorrelation and quartile plots, were also poor. Negative R_D^2 values are known to exist for very poor model fits. For both these reasons temperature and precipitation were not considered for addition into the model for daily CPUE. Lake level was added into the models for both datasets I and II, but its seen that this does not improve any fit values, or model diagnostics.

For dataset I, increasing the number of harmonics of the moonlight cyclical component keeps improving model fits and diagnostics, and this extends beyond four harmonics. However, increasing the number of harmonics in the yearly cyclical component tends to worsen model fits and diagnostics, even though it is graphically evident from Figures 4.1, 4.3(a), 4.4 and 4.7 that there should be a yearly cycle. All models show poor fit values as seen through low R_D^2 values, as only approximately 15% of the variance in the data can be explained through the best models built. Although it is difficult to select one model as best, it appears that the *local level model* with a cyclical component of period equal to 29.53 and four harmonics (** in Table 4.9) is best amongst all models fitted, seen by better

a)



b)



c)

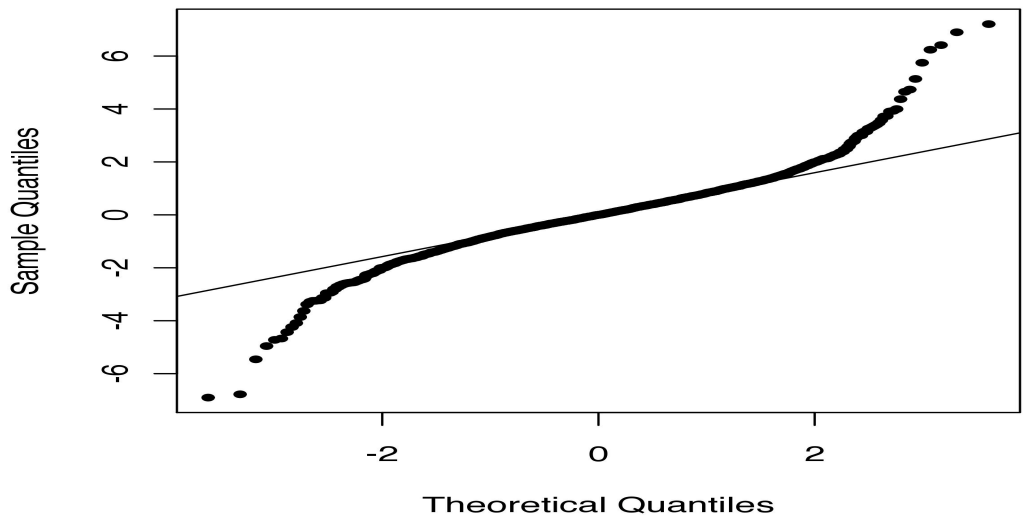
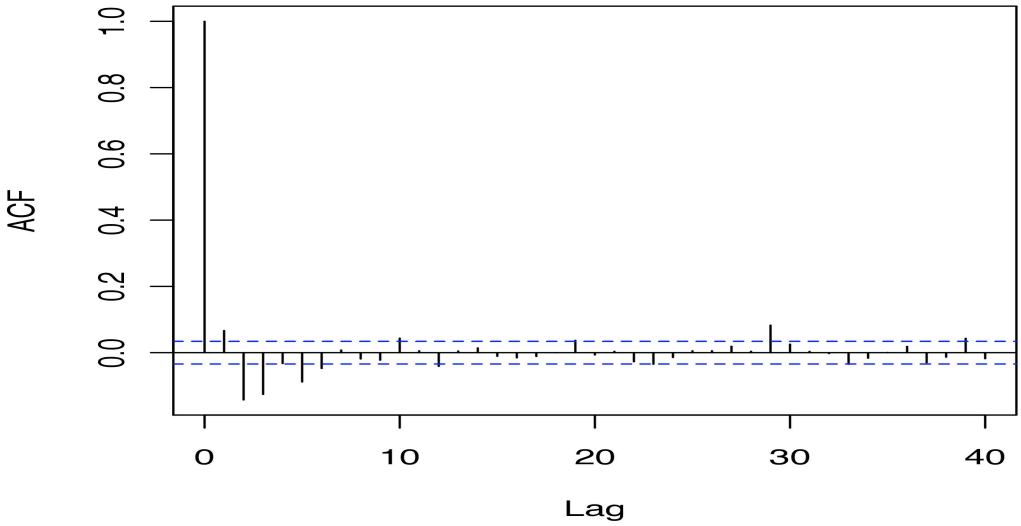
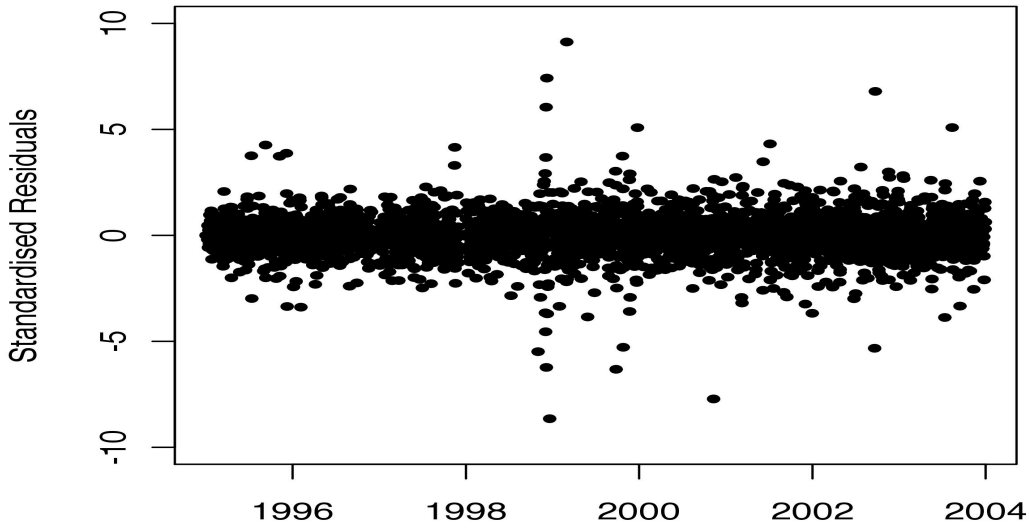


Figure 4.17: (a) Autocorrelation function, (b) Scatter plot ,and (c) QQ-plot of the standardized residuals for the best model using daily data from dataset I

a)



b)



c)

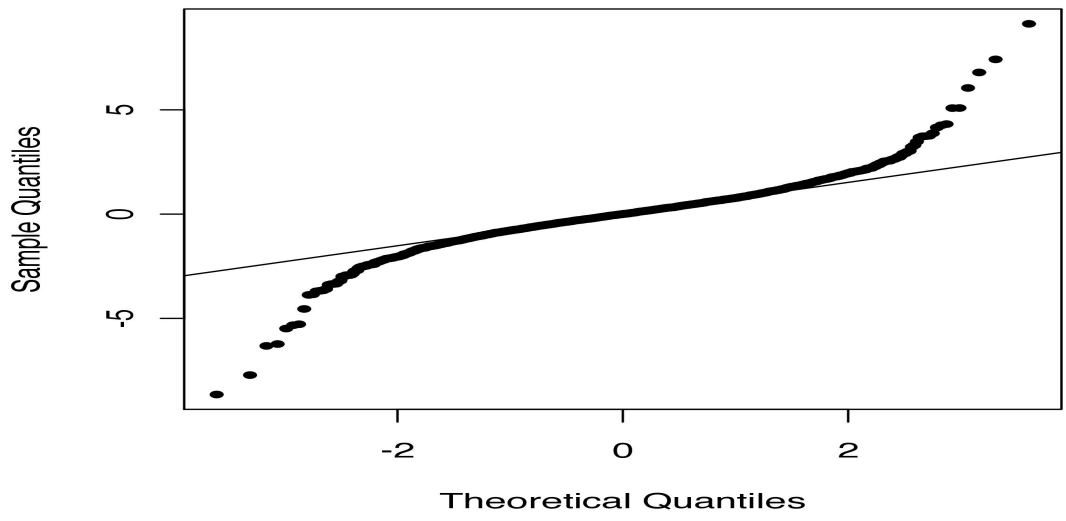


Figure 4.18: (a) Autocorrelation function, (b) Scatter plot ,and (c) QQ- plot of the standardized residuals for the best model using daily data from dataset II

overall diagnostics and goodness-of-fit values. Another worrying result is that diagnostic measures for these models, even the best model** selected above, are quite poor. This is seen in Figure 4.16. The autocorrelation function shows positively auto-correlated residuals, with more than 5% of the lags showing significant correlations. The scatter plot, checking the assumption of homoscedastic standardised innovations, does not appear majorly problematic. Residuals may appear to be slightly larger in earlier years, but smoothed values are known to jump up and down initially. The QQ- plot seems problematic, as it doesn't show the linear relationship indicating normally distributed standardized residuals.

Dataset II exhibits the same problems as dataset I. However, in this case the fact that adding a yearly cyclical component does not improve the model fit is even more perturbing, as dataset II had an even more obvious yearly cyclical component (Figure 4.1, 4.3(a), 4.4 and 4.7). Additionally, the fit values showed to be even poorer than that of dataset I, with the best models showing R_D^2 values of only approximately 8.7%. Although it is again difficult to choose one model as best, a local level model with cyclical component of period equal to 29.53 and four harmonics is the better of models observed (denoted ** in Table 4.10). Figure 4.17 shows diagnostics are also poor. The assumption of homoscedasticity does not seem to uphold badly, however there is a period in the middle of the dataset where residuals seem larger. The QQ- plot shows standardized residuals that are not normally distributed and the autocorrelation function shows positively auto-correlated residuals.

As discussed in Section 4.5.1 of this thesis, there is a danger in modelling components dynamically when using structural time series models, as these components may absorb much of the variability in the data and lose its simple interpretation. However, in an attempt to improve model fits, this approach was considered. Allowing the moonlight variable to be dynamically included, seen in equation (4.5.6) where the error component (σ_χ^2) is greater than zero, improves model fit for dataset I but not for dataset II, and fails to generate good diagnostics

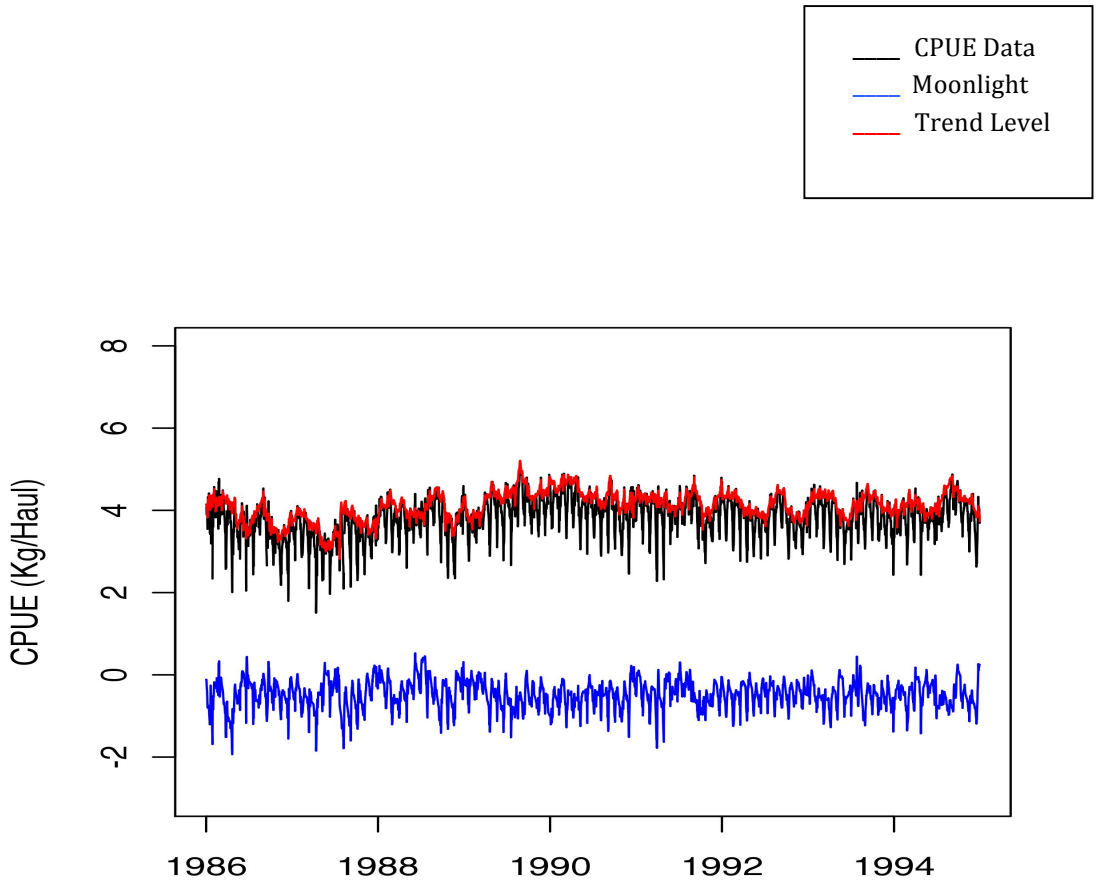


Figure 4.19: Plot showing daily CPUE with smoothed estimates of trend level and dynamic moonlight components for dataset I

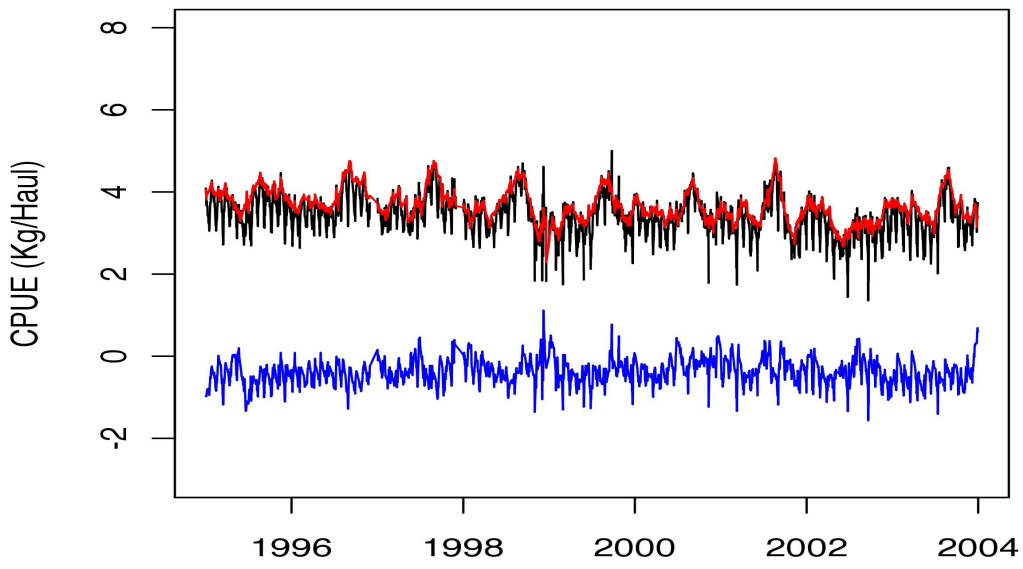


Figure 4.20: Plot showing daily CPUE with smoothed estimates of trend level and dynamic moonlight components for dataset II

for either dataset I or II. These goodness-of-fit values are seen in Tables 4.9. and 4.10, and diagnostic plots showed highly significant correlations, homoscedastic and non-normally distributed standardised residuals. Figures 4.18 and 4.19 show the dynamic moonlight component to absorb much of the variability of the data, as no clear cyclical movements are observed. Thus modelling the moonlight explanatory variable dynamically is not considered, despite the fact that model fit is improved for dataset I. Modelling moonlight through dynamic cyclical components with period equal to 29.53 days, as seen in equations (4.5.7) where the error components are greater than zero, did not improve models for either datasets I or II. As modelling components dynamically allows for more model flexibility, ideally one would want to uncomplicate the model, and include the cyclical component with only one harmonic. The results for this are also presented in Tables 4.9 and 4.10, and show poor goodness-of –fit measures for both datasets. Additionally, poor diagnostic plots were also observed, showing non-normally distributed, highly correlated and homoscedastic standardised residuals. The variance of the cyclical components for both datasets were estimated to be approximately zero, so modelling the moonlight cyclical component dynamically does not make much a difference to fit and diagnostic plots observed when modeling cyclical components non-dynamically. Increasing the number of harmonics in the dynamic cyclical components, undesirably complicating the model, similarly estimates variances to be approximately zero. Thus model fits and graphical diagnostics are not seen to improve much either when compared with corresponding non-dynamic models. Thus model components are not dynamically included.

4.7 Problems with Daily Data

Using daily data proved to be problematic when fitting structural time series models as described in Section 4.6. It was discovered that these poor results were attributed to the sampling interval of the data. As described by Brown (2004) “it is possible to sample so often that there is essentially nothing to look at.” Sampling intervals must be long enough for there to be a reasonable chance for some change to occur since the last observation. If the sampling rate is too high, as seen with

daily data, it is difficult to see a pattern in the data and hence is harder to forecast into the future. The longer the interval, the clearer the average level of the process (Brown, 2004). With daily data, highly variable data, there are short-term patterns for which there are no means of modelling or covariates available for predicting values. Patterns that emerge over aggregated data are seen to be more predictable and more easily modelled. As it was the aim of this thesis to find long-term trends and patterns in the data, it was decided to sum daily data to weekly data. Weekly CPUE and covariate data is calculated through the sum of daily values observed over a 7-day period. The sum of a week containing missing values is not calculated and left as a missing value for data accuracy purposes. One missing day may over- or under-estimate the weekly average greatly, due to the highly cyclical nature of the data. Since basin 5 contains very few missing data, the quality of the data was still high with few missing values.

4.8 Results for Structural Time Series Models using Weekly CPUE Data

Following the same approach outlined in Section 4.5.1, the model building process and fit results for models built using datasets I and II can be seen in Tables 4.12 and 4.13 respectively. For both datasets it appears from overall fit values that a local level model is best once again. Additionally, Figures 4.20 and 4.21 also show all smoothed slope estimates to be approximately zero. The *smooth trend model* for dataset II shows smoothed slope estimates slightly greater in value than those observed in the other kinds of models for trend, however this is expected as all the variance is transferred into the slope component. Additionally, the value of the slope component is not strictly in one direction, but oscillates around zero, suggesting *a local linear model* is still satisfactory. The *deterministic trend* model is again used to see the overall direction of CPUE through the smoothed mean level component of the trend. For dataset I, the *deterministic trend model* shows that the mean level of CPUE is slightly increasing overall, and for dataset II it is decreasing overall. In order to construct these confidence intervals, yearly cyclical, monthly cyclical and covariate effects are removed from the data,

Table 4.12: Table showing model building process and fit results using weekly data for dataset I

	Model	**AIC_1**	**AIC_2**	**P.E.V**	**R^2_D**
Trend	*Local Level Model*	-479.39	-463.27	0.128	0.236
	Smooth Trend Model	-441.12	-408.88	0.136	0.188
	Deterministic Trend Model	-345.06	-312.83	0.165	0.012
	Random Walk Model with Drift Model	-466.09	-433.86	0.128	0.233
	Local Linear Trend Model	-464.09	-431.86	0.128	0.235
Moon Cycle	*Local Level Model + Explanatory Variable (Moonlight)*	-795.48	-763.25	0.063	0.622
	Local Level Model + Explanatory Variable (Moonlight)+ Explanatory Variable (Moonlight^2)	-854.73	-806.37	0.055	0.672
	Local Level Model + Explanatory Variable (Moonlight)+…+ Explanatory Variable (Moonlight^3)	-853.87	-789.40	0.055	0.672
	Local Level Model + Cycle (period=29.53/7, harmonics=1)	-874.28	-825.92	0.052	0.690
	Local Level Model + Cycle (period=29.53/7, harmonics=2)	-947.89	-867.30	0.042	0.747
	Local Level Model + Cycle (period=29.53/7, harmonics=3)	-928.55	-815.72	0.042	0.746
Yearly Cycle	*Local Level Model + Cycle (period=29.53/7, harmonics=1)+Cycle (period=365/7, harmonics=1)*	-859.07	-778.48	0.052	0.687
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=1)	-932.98	-820.15	0.043	0.747
	Local Level Model + Cycle (period=29.53/7, harmonics=3)+Cycle (period=365/7, harmonics=1)	-913.64	-768.57	0.043	0.744
	Local Level Model + Cycle (period=29.53/7, harmonics=1)+Cycle (period=365/7, harmonics=2)	-877.75	-764.92	0.049	0.709
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=2)	-946.86	-801.80	0.040	0.760
	Local Level Model + Cycle (period=29.53/7, harmonics=3)+Cycle (period=365/7, harmonics=2)	-927.57	-750.27	0.040	0.759
	Local Level Model + Cycle (period=29.53/7, harmonics=1)+Cycle (period=365/7, harmonics=3)	-895.90	-735.82	0.047	0.722
	*Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)***	-948.17	-770.87	0.039	0.769
	Local Level Model + Cycle (period=29.53/7, harmonics=3)+Cycle (period=365/7, harmonics=3)	-928.64	-719.11	0.039	0.769
Expl. Variable	*Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Lake Level)*	NA	-746.42	0.039	0.769

Table 4.13: Table showing model building process and fit results using weekly data for dataset II

	Model	**AIC_1**	**AIC_2**	**P.E.V**	**R^2_D**
Trend	*Local Level Model*	-566.90	-550.79	0.103	0.052
	Smooth Trend Model	-495.43	-463.19	0.118	-0.910
	Deterministic Trend Model	-342.60	-310.36	0.163	-0.515
	Random Walk Model with Drift Model	-555.00	-522.77	0.103	0.048
	Local Linear Trend Model	-553.00	-520.77	0.103	0.050
Moon Cycle	*Local Level Model + Explanatory Variable (Moonlight)*	-856.53	-824.29	0.053	0.513
	Local Level Model + Explanatory Variable (Moonlight)+ Explanatory Variable (Moonlight^2)	-913.75	-865.40	0.046	0.576
	Local Level Model + Explanatory Variable (Moonlight)+…+ Explanatory Variable (Moonlight^3)	-913.98	-849.51	0.046	0.578
	Local Level Model + Cycle (period=29.53/7, harmonics=1)	-882.74	-834.39	0.049	0.550
	Local Level Model + Cycle (period=29.53/7, harmonics=2)	-942.51	-861.91	0.041	0.624
	Local Level Model + Cycle (period=29.53/7, harmonics=3)	-926.41	-813.58	0.040	0.627
Yearly Cycle	*Local Level Model + Cycle (period=29.53/7, harmonics=1)+Cycle (period=365/7, harmonics=1)*	-875.60	-795.01	0.048	0.553
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=1)	-934.62	-821.80	0.040	0.626
	Local Level Model + Cycle (period=29.53/7, harmonics=3)+Cycle (period=365/7, harmonics=1)	-918.56	-773.50	0.040	0.628
	Local Level Model + Cycle (period=29.53/7, harmonics=1)+Cycle (period=365/7, harmonics=2)	-892.97	-780.14	0.045	0.583
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=2)	-947.30	-802.24	0.038	0.646
	Local Level Model + Cycle (period=29.53/7, harmonics=3)+Cycle (period=365/7, harmonics=2)	-931.24	-753.94	0.038	0.648
	Local Level Model + Cycle (period=29.53/7, harmonics=1)+Cycle (period=365/7, harmonics=3)	-923.22	-764.67	0.042	0.612
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)	-960.25	-782.95	0.036	0.668
	Local Level Model + Cycle (period=29.53/7, harmonics=3)+Cycle (period=365/7, harmonics=3)	-513.25	-303.71	0.220	-1.039
Expl. variables	*Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Temperature)***	-968.20	-774.79	0.034	0.681
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Precipitation)	-945.74	-752.33	0.036	0.664
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Temperature Lag 1)	-948.34	-754.92	0.036	0.664
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Temperature Lag 2)	-520.59	-327.17	0.220	-1.036
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Temperature Lag 3)	-520.78	-327.36	0.219	-1.035
	Local Level Model + Cycle (period=29.53/7, harmonics=2)+Cycle (period=365/7, harmonics=3)+ Explanatory Variable (Lake Level)	NA	-757.37	0.036	0.667

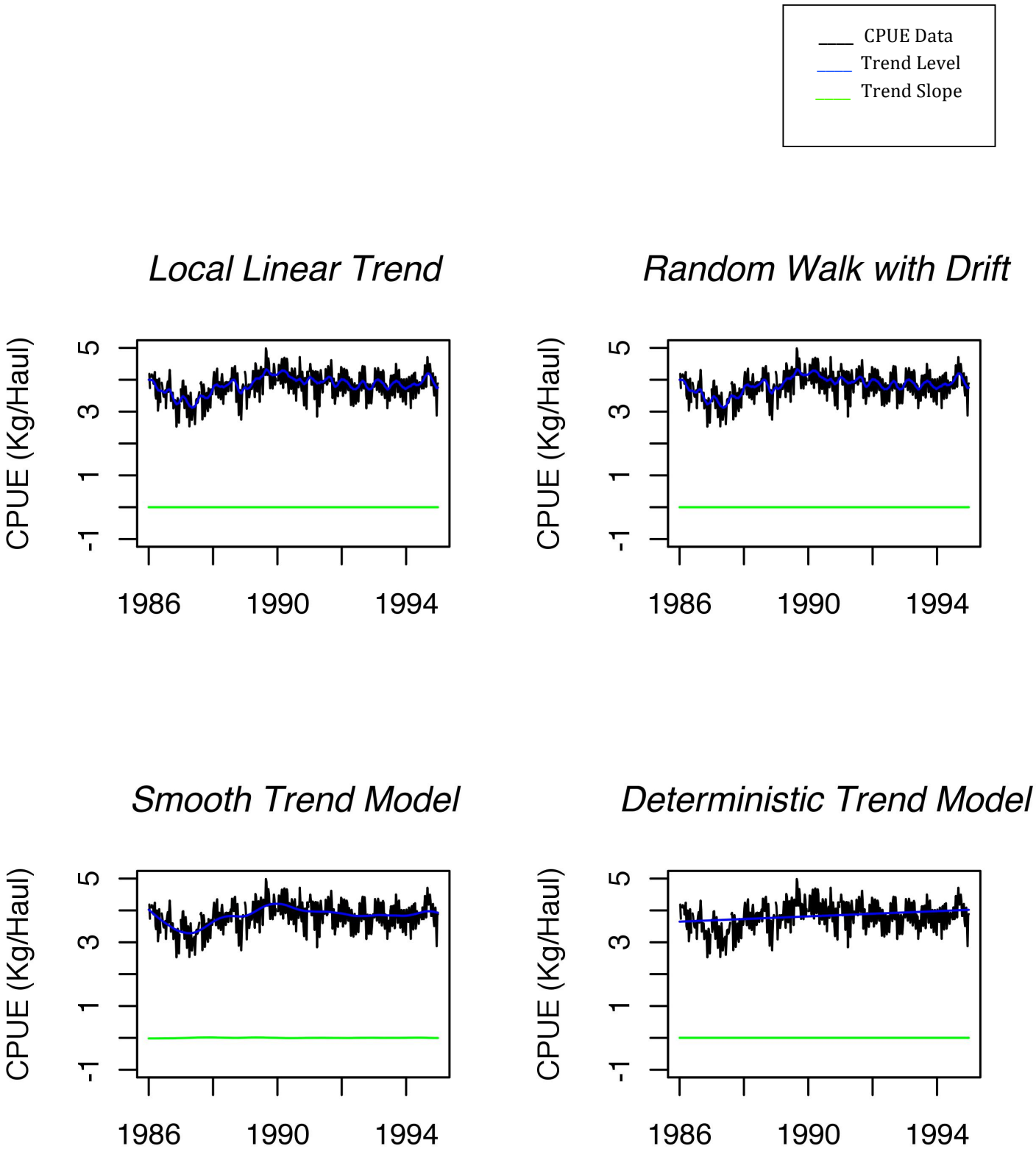


Figure 4.21: Plots showing weekly CPUE data with smoothed estimates of trend level and slope components, for the various model structures shown above for dataset I

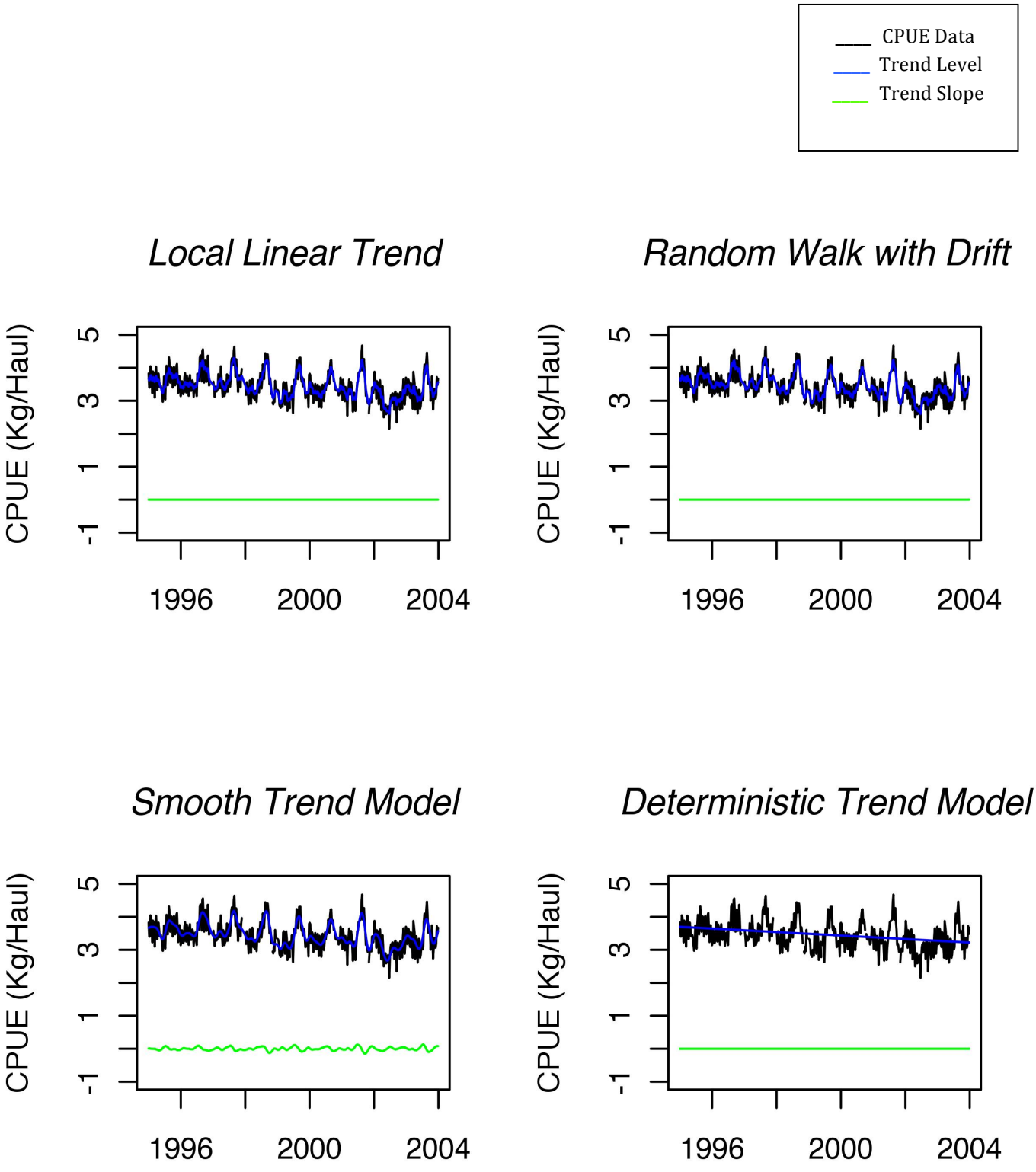


Figure 4.22:Plots showing weekly CPUE data with smoothed estimates of trend level and slope components, for the various model structures shown above for dataset II

by incorporating these components into the model. The presence of slope components into models are again tested through confidence intervals for the smoothed slope estimates in the model. The presence of trend slope components are assessed within *local linear trend* models for datasets I and II. This is the most flexible form for modelling trend with a slope component, and with cyclical and covariate effects provided the best overall model diagnostics and fit values of models including trend slope components. Additionally, confidence intervals were constructed for differences in the smoothed trend level estimates; the differences between each smoothed trend level estimate and the final smoothed trend level estimate in the time series.

The variance of the slope components in the constructed models for both datasets I and II are estimated to be approximately zero, and thus the confidence intervals remain constant over time.

Table 4.14: Table showing confidence Intervals for Slope Components with Weekly CPUE data

Dataset I	-0.0006 ± 0.0088
Dataset II	-0.0006 ± 0.0088

These confidence intervals show slope components to be unnecessary, with the intervals containing zero. The confidence intervals showing differences in the mean trend level, seen in Figures 4.22 to 4.23, show these differences to oscillate from being significantly negative to significantly positive for both datasets; more indicative of a random walk type model. Note that cycles still seem present in the series of differences between trend level components, as these components could not be completely eliminated in the model, as adding more harmonics into the cyclical components in an attempt to remove these cycles started generating models with very poor model fits and diagnostic measures. The prior belief of a local level model is again confirmed for models built using weekly CPUE data.

It is preferable to include moonlight through a cyclical component with period equal to 4.22 (29.53/7) weeks than through modelling moonlight as an explanatory variable. Increasing the number of harmonics in the cyclical components, or alternatively adding increasing powers of the explanatory variable

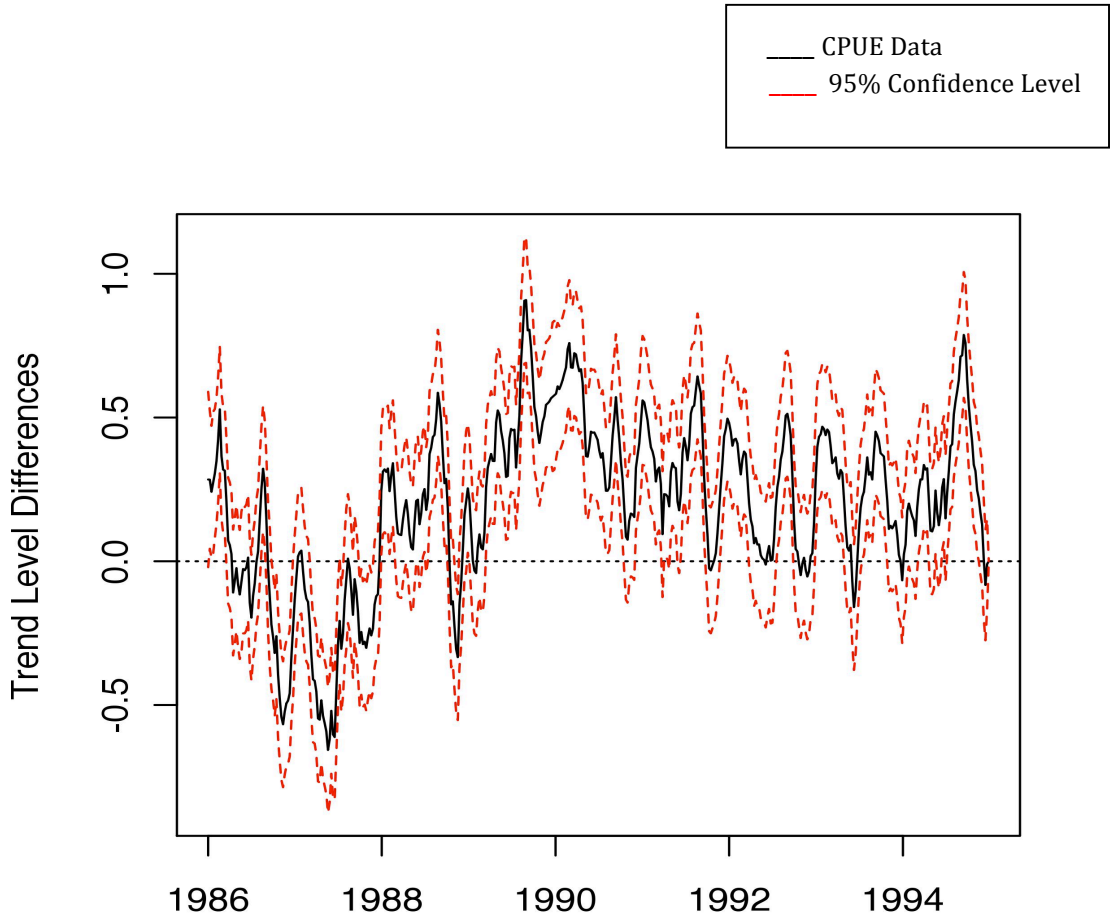


Figure 4.23: Plot showing confidence interval for the changes in trend level for a model using weekly data from dataset I

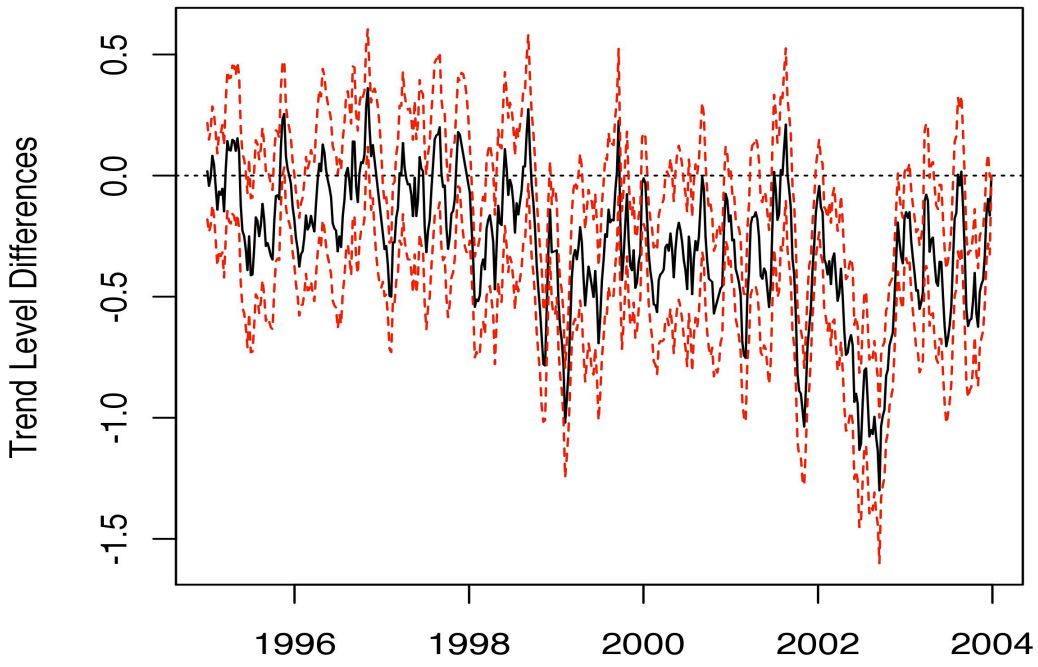


Figure 4.24 Plot showing confidence interval for the changes in trend level for a model using weekly data from dataset II

to the model, for increased flexibility improves model fits and diagnostics. However, this is better modelled through sines and cosines (cyclical components), seen through better goodness-of-fit measures, in Tables 4.12 and 4.13. Additionally, for both datasets, model fit does not continuously seem to improve as more harmonics are added to this component.

The presence of yearly cyclical components also improves goodness-of-fit measures and diagnostic plots, corresponding to the data exploratory analysis findings in Section 4.3. Up to three harmonics were tested for inclusion into both cyclical components, as including more than 3 was thought to be over fitting the data.

Lake level was tested for inclusion in the model build building process for both datasets I and II. Model fits or diagnostic plots did not seem to improve with the addition of this explanatory variable to either dataset. Temperature and precipitation data are available for dataset II only, and these were tested for inclusion as explanatory variables to the model for dataset II. Although few, some weekly values for these covariates are missing. To avoid having to alter the CPUE dataset so that missing covariate values correspond to missing CPUE values, for programming purposes (§3.9.3), missing values were replaced by smoothed average values. Additionally, two new variables defined as temperature with lags of two and three months respectively were also added into the model, as graphically (Figure 4.8) a lag in the effect of temperature on CPUE may be present. From Table 4.13 it is seen that only temperature (with no lag) appeared to improve model fit. From the goodness-of-fit measures seen in Tables 4.12 and 4.13, and through the use of diagnostic plots, the best models for datasets I and II were selected. It is seen that local level models added with two cyclical components where period equals 4.218 (=29.53/7) and 52.143 (=365/7) weeks respectively, the first being fit with two harmonics and the second with 3 harmonics, provides the best overall fit measures for both datasets. Additionally the best model for dataset II includes temperature as an explanatory variable. These models possess the smallest AIC_1 measures, the smallest P.E.V measures and the highest R_D^2 measures. Models fit to weekly data provide strong R_D^2

measures, with the best models explaining approximately 77% and 67% of the variance in datasets I and II respectively. These best-selected models for datasets I and II (denoted by ** in Tables 4.12 and 4.13) will be referred to in the remainder of this thesis as Model D1 and Model D2 respectively. Additionally, maximum likelihood estimates for the unknown observation (σ_ε^2) and evolution (σ_ξ^2) variances are provided below for Models D1 and D2.

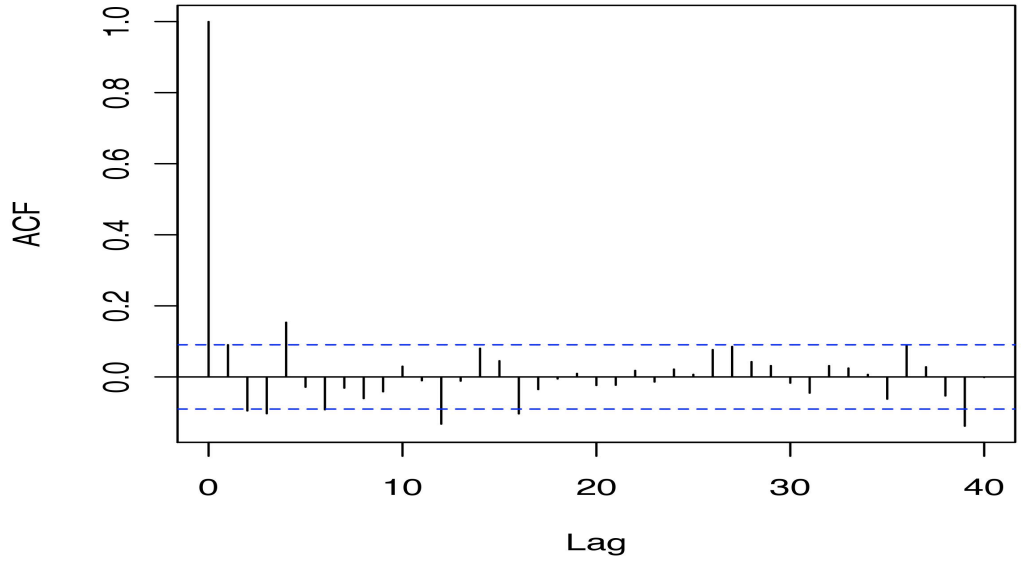
Table 4.15: Table showing unknown parameter estimates for models D1 and D2 using weekly CPUE data

	Model D1	**Model D2**
σ_ε^2	0.023	0.012
σ_ξ^2	0.006	0.014

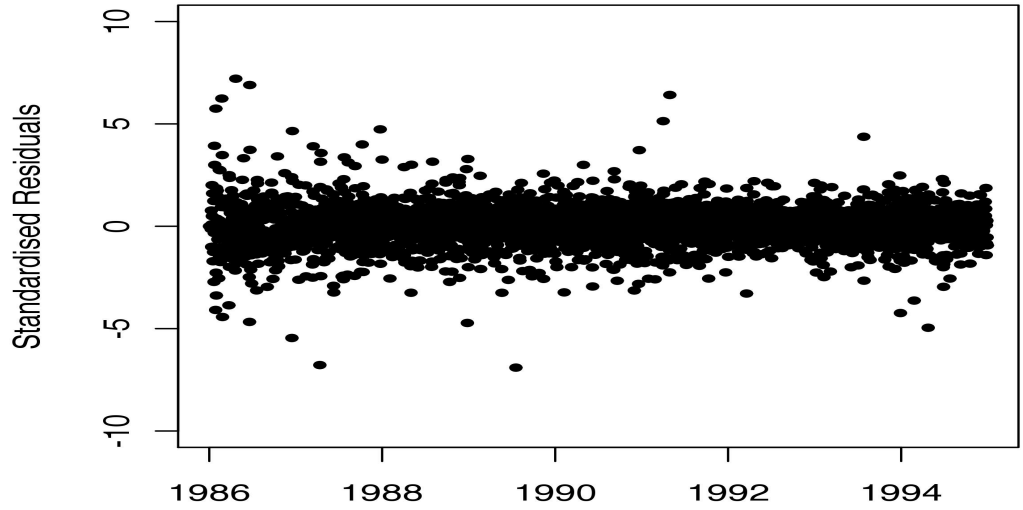
Diagnostic checks performed on models D1 and D2 for datasets I and II respectively aim to assess whether residuals produced from chosen models are random and normally distributed; these are seen in Figures 4.24 and 4.25. The autocorrelation functions show very few lags with significant correlations, however after 4 weeks there are still slight correlations left for both datasets. Four weeks correspond to a period of 28 days, and these correlations may be attributed to the moonlight effect on CPUE, not accounted for by the cyclical components with 3 harmonics. Adding additional harmonics to these cyclical components may eventually eliminate this correlation, however fit values and diagnostic plots do not change significantly or quickly past 3 harmonics. Residuals are seen to be homoscedastic, with the variance of the residuals appearing constant in both scatter plots for datasets I and II. The assumption of normality is well upheld through the almost perfectly linear presence of the QQ-plots for datasets I and II. This shows great improvements to the models described in Section 4.6, when models were built for daily CPUE.

The model results described above for datasets I and II are pleasing, not only because they show reasonable fit values and diagnostics, but because they justify biological expectations outlined in Chapter 1. Additionally these results coincide with expectations in Section 4.3, under the exploratory data analysis. These biological and graphical expectations are all outlined in Section 4.4 of this thesis.

a)



b)



c)

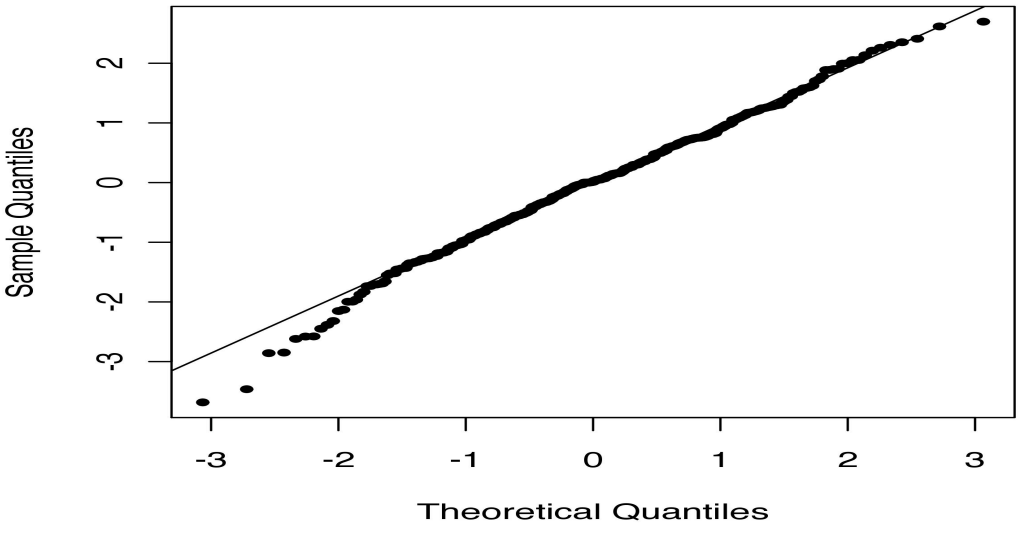
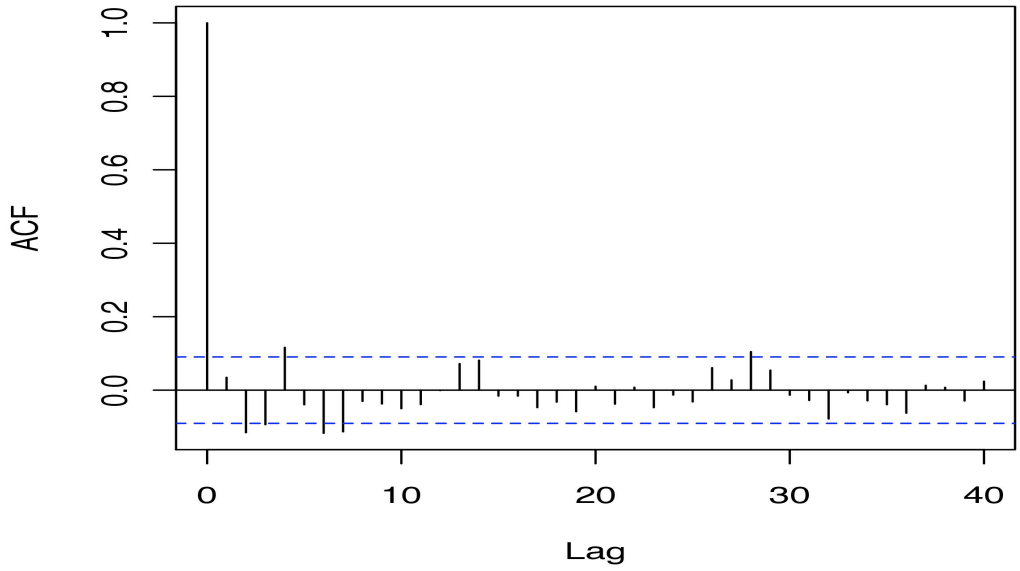
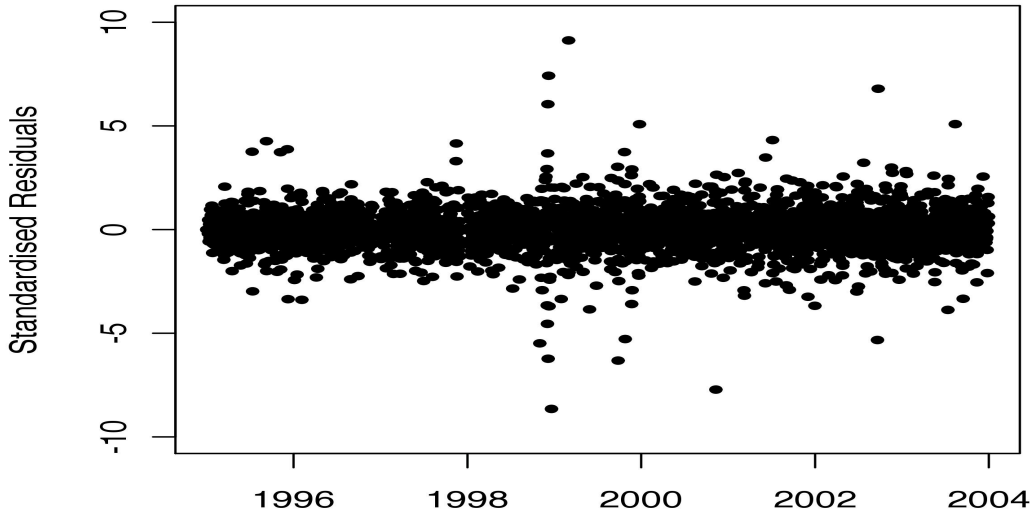


Figure 4.25: (a) Autocorrelation function, (b) Scatter plot ,and (c) QQ- plot of the standardized residuals for the best model using weekly data from dataset I

a)



b)



c)

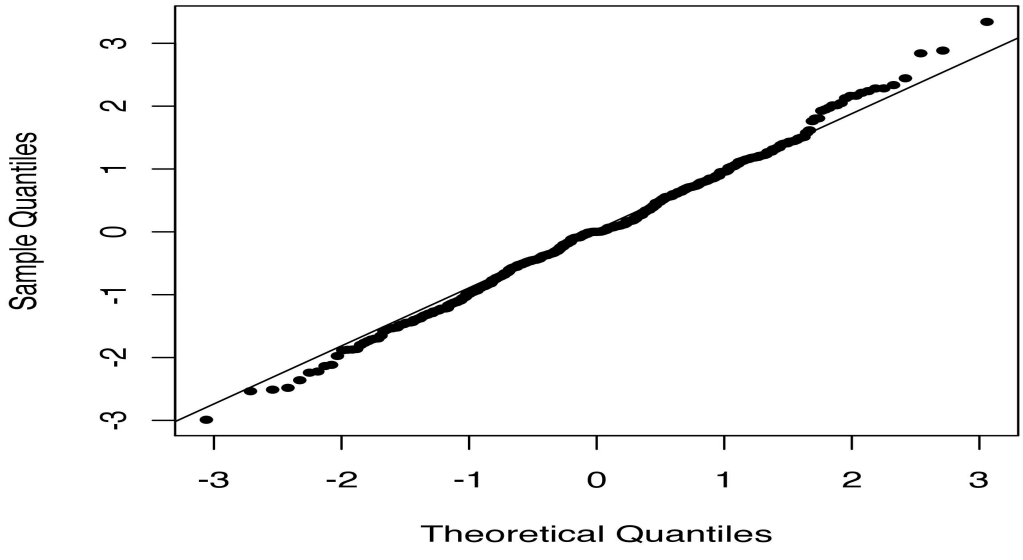


Figure 4.26: (a) Autocorrelation function, (b) Scatter plot ,and (c) QQ plot of the standardized residuals for the best model using weekly data from dataset II

The presence of a cyclical component with period equal to 29.53 days, corresponding to the cycle of the moon, was clearly seen to exist in the data in Section 4.3 (Figures 4.2, and 4.6). This component was hypothesized to exist in Chapter 1, due to light attractions being used in Kapenta fishing. The yearly cyclical components were also thought to exist in Section 4.3 (Figures 4.1, 4.3(a), 4.4 and 4.7). It was noted that CPUE grew larger over the cold months of July/August, and CPUE was lowest over the hotter months of the year starting from October. This effect was also hypothesized in Chapter 1, since Kapenta fish move inshore to breed over summer months and then back into open waters over the winter months. It was noted in Section 4.4 that water level or precipitation was not expected to effect CPUE levels. These covariates change so slowly, and so slightly that it was thought these variables would not influence any frequently observed changes in CPUE. Temperature, however, was a covariate that was expected to influence CPUE, as biologically (see §4.4) and graphically (Figure 4.8) there is a relationship between these two variables. However, it must be noted that much of the temperature effect on CPUE is most likely captured by the yearly cyclical component, as the moving of Kapenta fish in and out of open waters correspond closely with summer and winter months. Temperature is seen to provide a slight improvement in model fit, but not much on diagnostic plots. Due to this improvement in fit, however, it is retained in the model for dataset II. This improvement in model fit is thought to account for any temperature variations out of the ordinary that may affect CPUE, decreasing CPUE effort when it is unusually hot and visa versa. Since moonlight is seen to provide such a strong effect on CPUE, it was thought that cloud cover would similarly affect CPUE. Chapter 1 describes how cloud cover may diminish the amount of light emitted from the moon, and so increases the efficacy of the lights used during fishing to attract Kapenta. In order to see whether cloud cover affects CPUE, it is added to model D1. However, as only 3 years of the cloud cover data is available, it is modelled using the corresponding subset from dataset I. Fit values for model D1 over this new, smaller dataset is provided first in Table 4.16, then fit values for model D1 with cloud cover as an explanatory variable is shown for comparison. For the same reasons outlined in Section 4.4, the cloud cover explanatory variable was not rendered dynamic.

Table 4.16: Table showing model results for model D1 including cloud cover

Model	AIC_1	AIC_2	P.E.V	R^2_D
Model D1	-237.52	-60.22	0.048	0.679
Model D1 + Explanatory Variable (Cloud Cover)	-228.11	-34.69	0.048	0.679

From Table 4.16 it is seen that cloud cover does not improve model fit, as seen through lower AIC values and unchanged P.E.V and R^2_D values. Additionally, diagnostic plots did not improve after the addition of cloud cover into the model.

The models D1 and D2 show that the factors affecting weekly CPUE over both periods are essentially the same factors, with the exception of temperature where this data is not available over the period of dataset I. However, looking at the maximum likelihood parameter estimates for the unknown variances in Table 4.15, certain differences between the two datasets are noticed. Dataset I shows larger observation variance than dataset II; this is expected as dataset I was observed in Table 4.2 to show higher variance. Dataset II however shows greater stochastic movements in the trend, as seen by larger evolution variance. This indicates a less smooth trend component, and the mean trend level of CPUE changes more greatly and significantly over the period of model D2.

Looking at the smoothed estimates of the trend, moon cycle and yearly cycle components along with the CPUE data for dataset I, in figure 4.26, shows the fit of model D1 graphically. Figure 4.27 is the same figure, providing only the first 100 observations; this allows one to see the smoothed estimates of the moon phase cycle relative to the data more clearly. Figures 4.28 and 4.29 provide these same figures for model D2 of dataset II, including the smoothed estimates of the temperature variable. Smoothed trend components are seen to capture the mean level of the data, with the trend component appearing more variable for dataset II due to the larger trend variance. The trend component for dataset II in Figures 4.28 and 4.29, are seen to be below the CPUE data. The reason for this is the addition of the temperature covariate; since the smoothed components must be added together to provide the smoothed representation of the data, adding the temperature covariate with positive coefficient, forced smoothed trend component

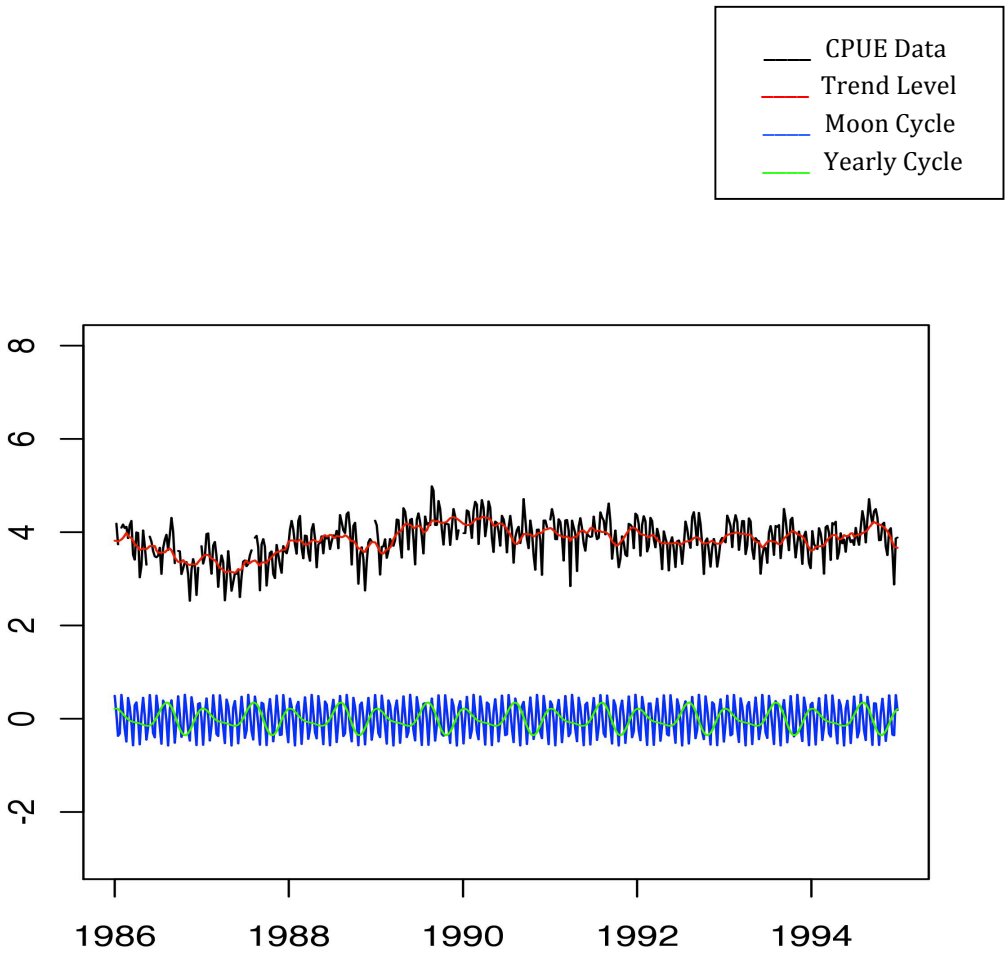


Figure 4.27: Plot showing weekly CPUE data with smoothed estimates of trend level, moonlight cyclical and yearly cyclical components for dataset I

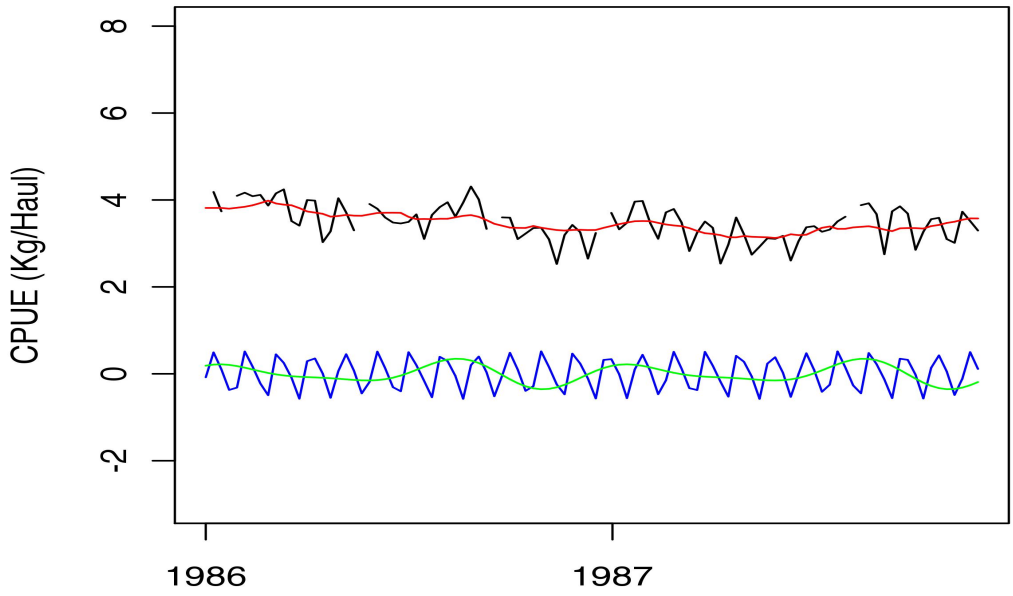


Figure 4.28: Plot showing 2 years of weekly CPUE data with smoothed estimates of trend level, moonlight cyclical and yearly cyclical components for dataset I

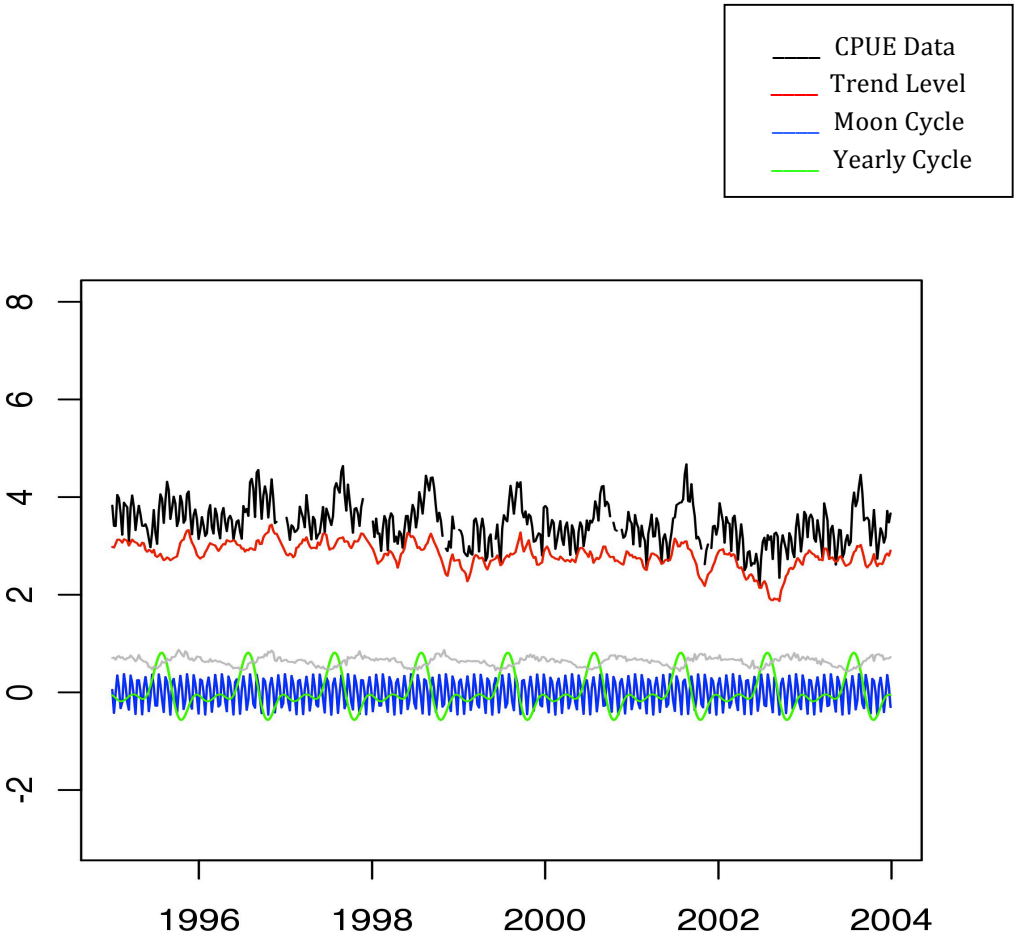


Figure 4.29: Plot showing weekly CPUE data with smoothed estimates of trend level, moonlight cyclical and yearly cyclical components for dataset II

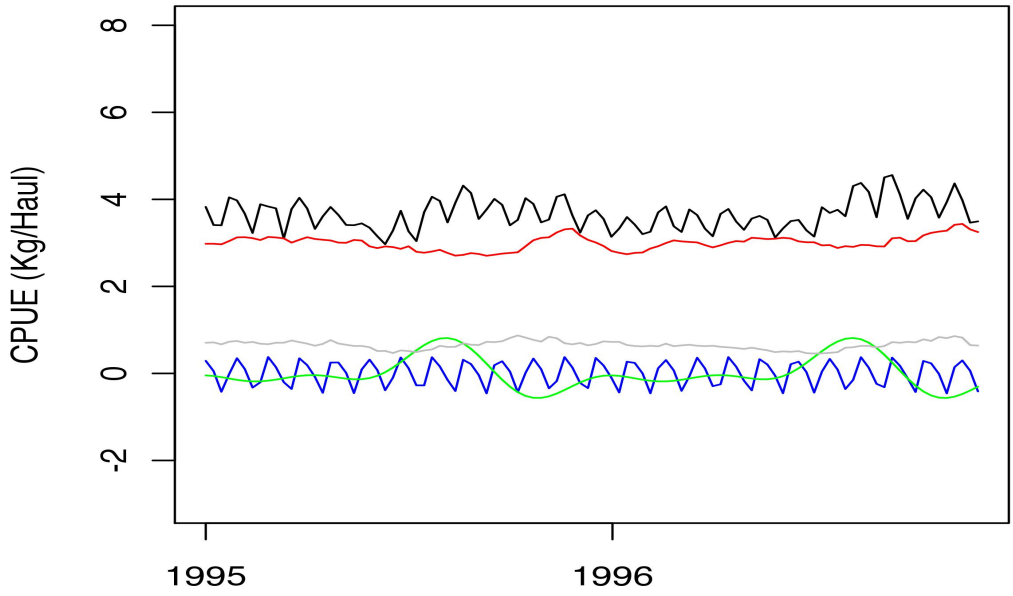


Figure 4.30: Plot showing 2 years of weekly CPUE data with smoothed estimates of trend level, moonlight cyclical and yearly cyclical components for dataset II

to decrease in value. The fit for both models D1 and D2 appears good, with the yearly and moonlight cyclical components in the data being captured well by the cyclical components provided in the model. The temperature covariate shows smoothed estimates that cycles; when temperature increases, CPUE is seen to start decreasing and thus temperature captures any additional cyclical behaviour in CPUE not captured by the yearly cyclical component.

The improvement of models describing CPUE when summing daily data into weekly data is profound, however it must be noted that models are still flawed. Single diagnostic measures such as the *H(h)* measure for homoscedasticity and the Box-Ljung statistic *Q** discussed in Section 3.6.1 continued to show significant heteroscedasticity and correlations for weekly data. Although graphical diagnoses improved and are reasonable to good, model assumptions are not perfectly met as indicated by these measures and imperfections in these graphical diagnoses. Additionally fit values, such as R_D^2, although reasonably strong do not indicate that enough of the variance within the data is explained to warrant forecasting.

4.9 Bayesian Analysis

Bayesian estimation and smoothing is illustrated in this section for the selected models D1 and D2, of datasets I and II, described in Section 4.8. In both these models the unknown parameters are the two error variances $\psi = (\sigma_\varepsilon^2, \sigma_\xi^2)$. Independent inverse-gamma distributions are assumed as priors, $(\sigma_\xi^2)^{-1} \sim Gamma(\alpha_\xi, \beta_\xi)$, where α and β are the shape and rate parameters of the gamma distribution. Gibbs sampling from the joint posterior $\pi(\sigma_\varepsilon^2, \sigma_\xi^2, \alpha_{0:n} \mid y_{1:n})$ is implemented, running 10000 MCMC iterations. The first 500 draws are discarded as burn-in values.

Diagnostic plots of the MCMC output are first observed. Figure 4.30 shows the running sample means and empirical autocorrelation functions respectively for the MCMC samples of the variances for the model D1 of dataset I. Figure 4.32 shows the same for model D2 of dataset II. The convergence of the MCMC output can be

seen in both figures 4.30 and 4.32, with the ergodic means stabilising for both the observation and evolution variances. MCMC samples of the variances show no correlation with previous samples, a pleasing result for samples that are potentially highly correlated. Figure 4.31(a)-(b) shows the posterior distribution of the observation and evolution variances for model D1 of dataset I. Additionally Figure 4.31(c) plots the MCMC samples from their joint posterior density of σ_{ε}^2 and σ_{ξ}^2 for model D1 of dataset I where a low correlation is evident reflecting a faster mixing of the Gibbs sampler. Figure 4.33 represents the same results as Figure 4.31 for model D2 of dataset II. The MCMC samples from the joint posterior density of σ_{ε}^2 and σ_{ξ}^2 for the model D2 of dataset II also shows low correlation and a faster mixing of the Gibbs sampler.

The Bayesian estimates of the unknown variances with respect to quadratic loss, are given by their posterior expectations whose MCMC estimate, together with Monte Carlo standard errors are given below for both Models D1 and D2. Parameter estimates using Bayesian analysis are seen to produce almost exactly the same parameter estimates as those produced using maximum likelihood, and thus model fits, diagnostics and interpretations are very much the same as those discussed in Section 4.8 of this thesis.

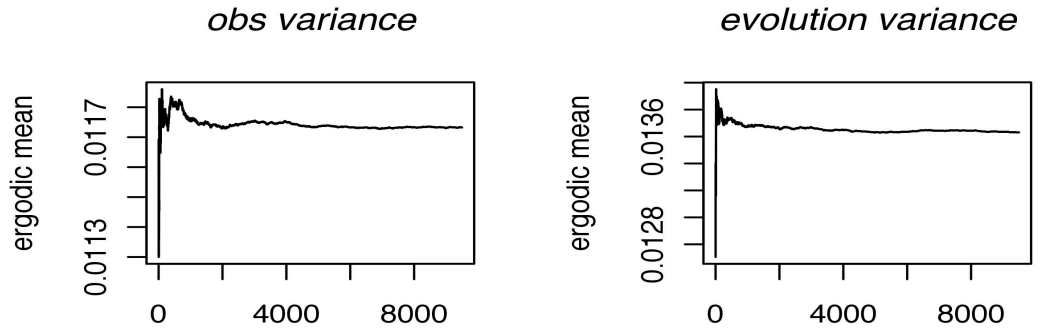
Table 4.17: Table showing parameter estimates using a Bayesian approach with regard to quadratic loss for model D1 using dataset I

Model D1	**Observation Variance** (σ_{ε}^2)	**Evolution Variance** (σ_{ξ}^2)
Parameter Estimate	0.0226	0.0058
Standard Deviation	1.75×10^{-5}	5.44×10^{-6}

Table 4.18: Table showing parameter estimates using a Bayesian approach with regard to quadratic loss for model D2 using dataset II

Model D2	**Observation Variance** (σ_{ε}^2)	**Evolution Variance** (σ_{ξ}^2)
Parameter Estimate	0.0117	0.0136
Standard Deviation	1.01×10^{-5}	1.16×10^{-5}

a)



b)

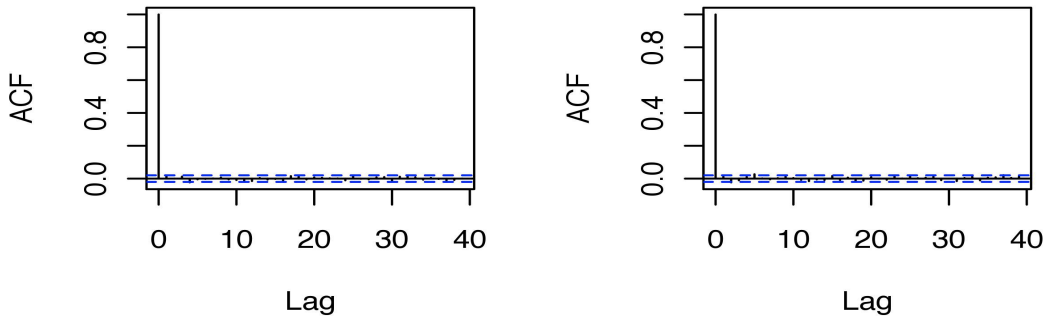


Figure 4.31: (a) Plots showing running sampling means for both the observation and evolution variances of dataset I
(b) Autocorrelation functions for both the observation and evolution variances for model D1 of dataset I

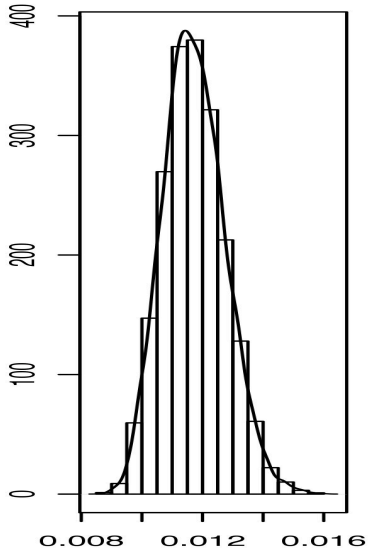


Figure 4.32(a): Posterior density of observation variance for dataset I

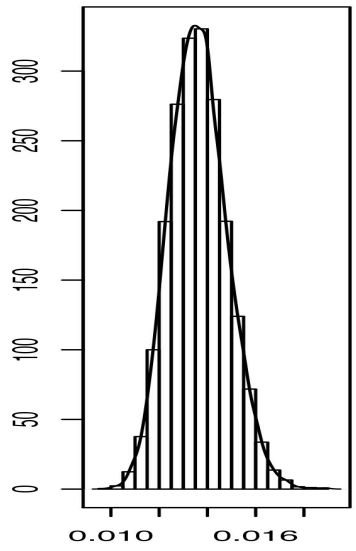


Figure 4.32(b): Posterioir density of evolution variance for dataset I

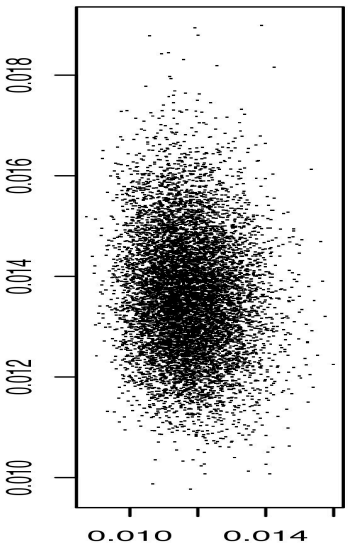


Figure 4.32(c): MCMC samples from the joint posterior of the observation and evolution variances for model D1 of dataset I

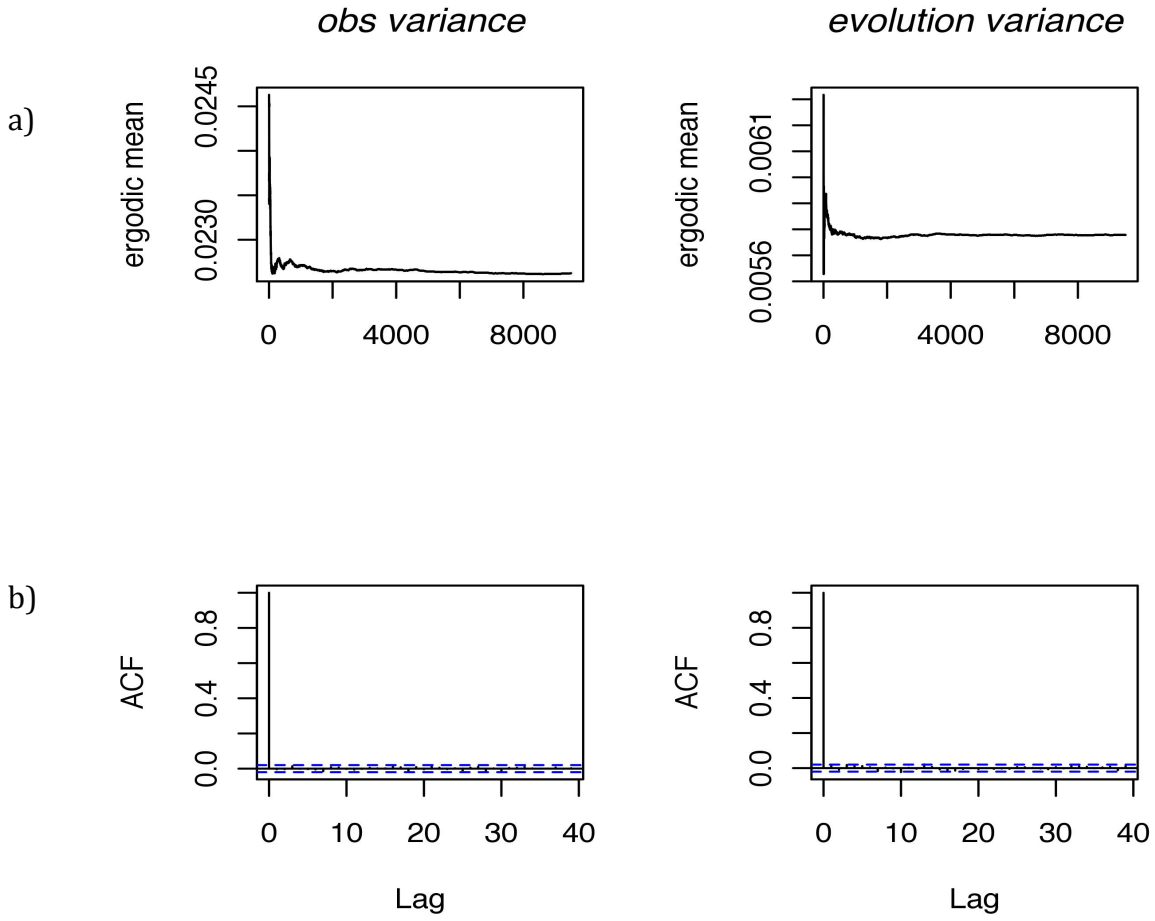


Figure 4.33: (a) Plots showing running sampling means for both the observation and evolution variances of dataset II
(b) Autocorrelation functions for both the observation and evolution variances for model D1 of dataset II

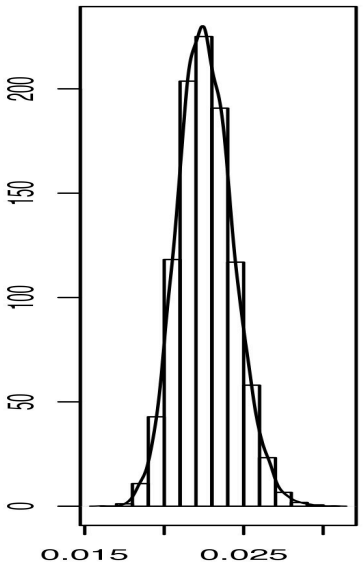


Figure 4.34(a): Posterioir density of observation variance for dataset II

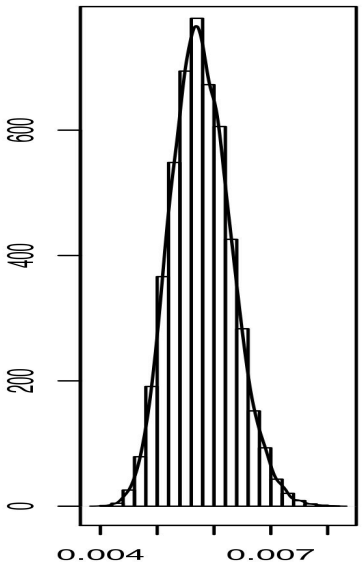


Figure 4.34(b): Posterioir density of evolution variance for dataset I

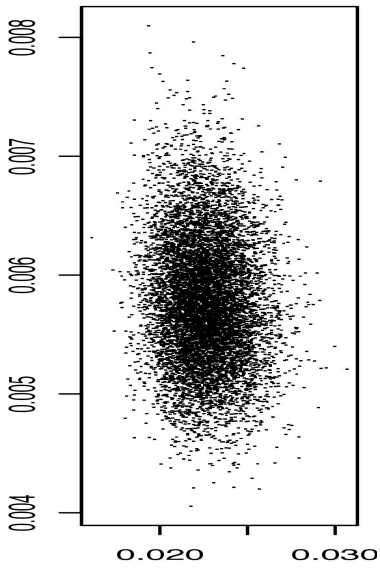


Figure 4.34(c): MCMC samples from the joint posterior of the observation and evolution variances for model D1 of dataset I

4.10 Multivariate Analysis

Finally an extension of univariate models into multivariate models is considered. Univariate models built for log-transformed CPUE data for basin 5 are extended into a multivariate model including the log-transformed CPUE data for basins 2 to 4. Basin 1 is not considered for the multivariate extension, as this data contained too many missing values, as seen in Table 4.1. As discussed in Section 3.9, multivariate models are built using the `dlmSum` function of the `dlm` package in R. Model D1 was extended across basins 2 to 5, for dataset I. The form of model D1 (or model D2 less the temperature explanatory variable) is extended across basins 2 to 5 for dataset II as well. Explanatory variables, modelled using the `dlmModReg` function, cannot be included into the multivariate model using the `dlmSum` function, as this returns errors stating that this function does not include time-varying dynamic linear models. This error is returned whether the explanatory variables are time varying or not.

Figure 4.34 shows plots of weekly CPUE over time in each of the four basins considered in this multivariate analysis. Figure 4.35 shows the same, over a 2-year period. Yearly and monthly cyclical components are similarly observed to exist in basins 2 to 4, as observed for basin 5. Figure 4.34 shows that the trend for basin 4 appears downward sloping over dataset I, and here a *local linear trend* component might be more appropriate for this basin over dataset I. Basin 2 appears more stationary overall than basin 5, with less overall increasing and decreasing components in the data. However, generally it appears that the *local level model* and cyclical components, with periods equal to 29.53 and 365 respectively, selected for basin 5 are mostly relevant for basins 2 to 4. Correlations between the log-transfomed CPUE data for basins 2 to 5 are displayed in Table 4.19 below. This shows how closely CPUE data are related between the basins. If the data between each basin and basin 5 are related, perhaps the form of model selected for basin 5 (models D1) may be appropriate for basins 2 to 4 as well, and multivariate models may be advantageous. The correlations between basins 3 and 5 appear slightly higher, but only slightly more so than that seen between basins 5 and 4.

Table 4.19: Table showing correlations between the weekly CPUE data of basins 2-5 of Lake Kariba

	Basin 5	**Basin 4**	**Basin 3**	**Basin 2**
Basin 5	1			
Basin 4	0.524	1		
Basin 3	0.699	0.575	1	
Basin 2	0.483	0.506	0.512	1

When building a multivariate model, the `dlm` package provides the loglikelihood for the model as a whole, using the formula seen in (3.9.1). From this a single AIC value for the multivariate model can be calculated using (3.6.1). A multivariate model is constructed as described in Section 3.7.2 of this thesis. It contains four observation and four evolution variances, the diagonals of matrices $\sum_{\varepsilon}$ and $\sum_{\eta}$ respectively, each pair corresponding to one of the four basins considered. For models assuming non-independent model components, covariances between the disturbances of components are off-diagonal elements of matrix $\sum_{\eta}$. The form of model D1 is extended across the basin of Lake Kariba to form multivariate models for both datasets I and II, and as only trend level components are dynamically modelled, covariances exist only between disturbances of the trend level components. In this section it is assumed that observation disturbances are uncorrelated, and thus $\sum_{\varepsilon}$ is modelled as a diagonal matrix. This is a reasonable assumption, and simplifies the computation time of the model greatly. These variances are all calculated using maximum likelihood, and multivariate models assuming non-independent model components use the log-Cholesky parameterisation for the covariance matrix $\sum_{\eta}$ discussed in Section 3.9.10. For multivariate models assuming independent model components between different basins, fit measures can be calculated through constructing a univariate model for each of the basins, using the parameter estimates corresponding to that basin in the multivariate model. Goodness-of-fit measures and diagnostics are then constructed as in the univariate case, discussed in Section 3.6.

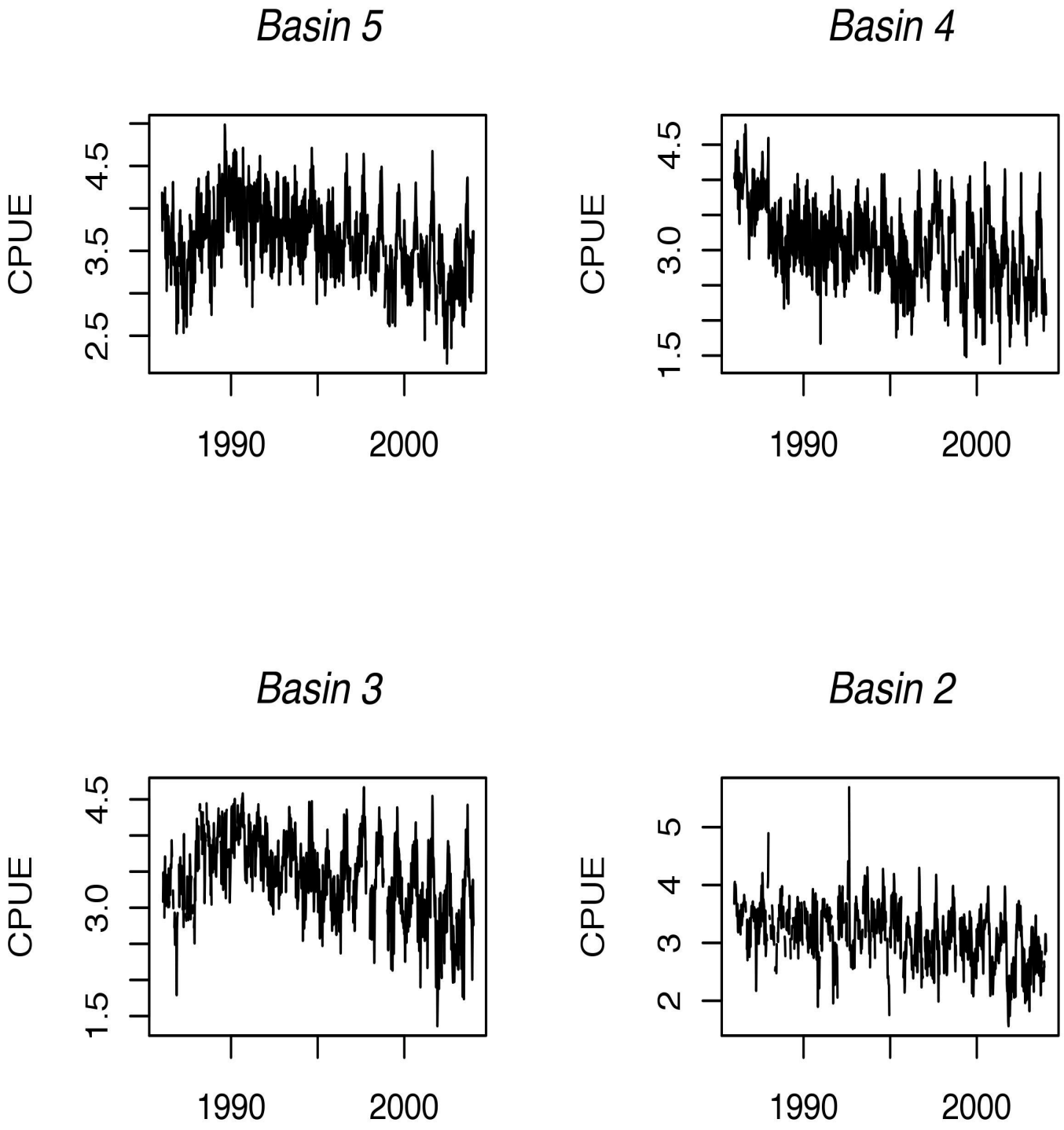


Figure 4.35: Figures showing CPUE for Basins 2 to 4 over the entire dataset

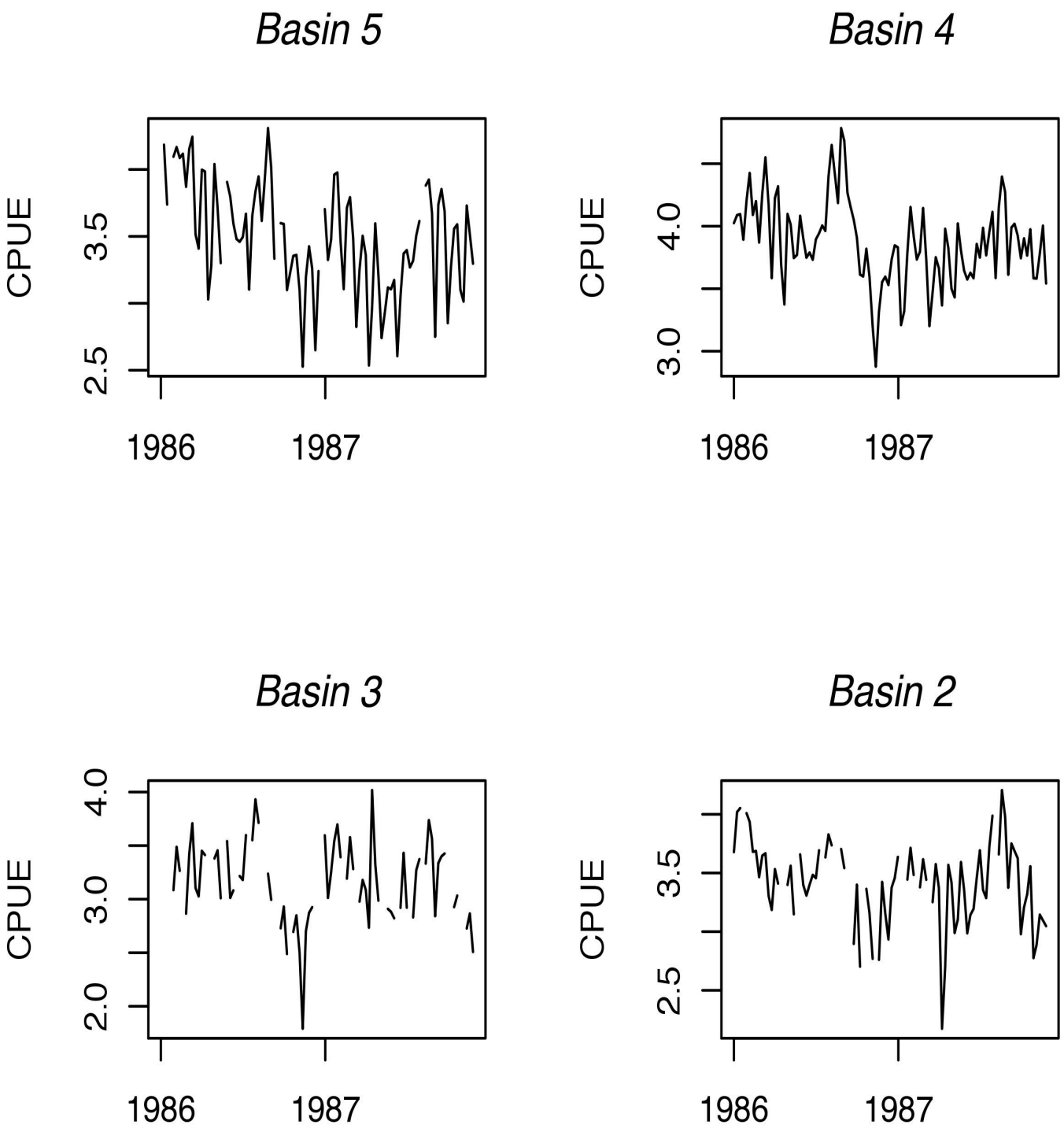


Figure 4.36: Figures showing 2-years of CPUE data for Basins 2 to 5

These results can be viewed to assess the fit of the multivariate model to each basin in particular, and the sum of the loglikelihood values equals the overall loglikelihood for the multivariate model.

Results for extending model D1 across basins 2 to 5 for dataset I, assuming independent models between basins, are displayed in Table 4.23; call this Model M1. Looking at the R_D^2 values, in addition to providing a good fit to basin 5, the form of model D1 is seen to provide a good fit for basin 4 and an average to poor fit for basin 3. This model provides a poor fit for basin 2, producing negative R_D^2 values. This can however be attributed to the fact that the data for basin 2 contained many of its missing values in the first half of the dataset, and this result must thus be interpreted with caution. The R_D^2 values indicate that this multivariate model, M1, provides a reasonable fit across four basins of Lake Kariba, however remaining most applicable to basin 5. Figure 4.36 shows the model diagnostics of model M1 for basins 2 to 4. The diagnostics for basin 5 were discussed in Section 3.8. Diagnostics don't appear too poor. The autocorrelation functions for basins 3 and 4 are good, showing few significant correlations. The autocorrelation function for basin 2 is not as good, showing more significant correlations. The scatter plots show the variance of standardised residuals to remain mostly constant, thus the assumption of homoscedasticity may be well upheld. The QQ-plots appear reasonable, especially for basin 4 and appear mostly linear, except for at the extremes of the quartiles for basins 2 and 3.

Results for extending model D1 across basins 2 to 4 for dataset II, assuming independent models between basins, are displayed in Table 4.24; call this model M2. In addition to providing a good fit for basin 5, the R_D^2 values show that the form of model D1 provides good fits for basins 2 and 3 as well. The model thus appears to become more relevant over the later years of the study to other basins considered. It is noted that basin 4 is not present in the multivariate extension of dataset II. Modelling all four basins in the multivariate model for dataset II returned errors of non-convergence in the maximum likelihood estimates of unknown parameters. Basin 4 also contained many of its missing values in the second half of the dataset and non-convergence occurred as long as basin 4 was

considered in the model. For this reason the multivariate model was run excluding basin 4. The multivariate model, M2, provides a good fit across basins 2,3 and 5. Figure 4.36 shows the model diagnostics of model M2 for basins 2 and 3. The diagnostics for basin 5 were discussed in Section 4.8. Diagnostics appear reasonable. The autocorrelation function for basin 3 is good, showing few significant correlations. The autocorrelation function for basin 2 is however not as good, showing more significant correlations. The scatter plots show the variance of standardised residuals to be mostly constant over time, thus the assumption of homoscedasticity may be upheld. The QQ-plots appear reasonable. They are mostly linear, except for at the extremes of the quartiles where the assumption does not appear to uphold.

The multivariate models M1 and M2, for datasets I and II respectively, constructed in Tables 4.23 and 4.24, allowed for each observation and evolution variance corresponding to the different basins considered, to be estimated individually. It is possible to simplify the multivariate model into one where each basin in the model possesses the same observation and evolution variances. This greatly reduces computation time, but can be inaccurate when these variances are in fact very different. The observation and evolutions variances estimated in models M1 and M2 are displayed in Table 4.20 below:

Table 4.20: Table showing Observation and Evolution variance estimates for Models M1 and M2

		Model M1	**Model M2**
Observation Variance	Basin 5	0.023	0.013
	Basin 4	0.040	-
	Basin 3	0.046	0.040
	Basin 2	0.046	0.021
Evolution Variance	Basin 5	0.006	0.014
	Basin 4	0.007	-
	Basin 3	0.007	0.010
	Basin 2	0.022	0.020

Table 4.20 suggests that these variances are in fact different across the basins, and estimating only one observation and evolution variance across the basins may lead

to inaccurate models, therefore such models were not constructed in this thesis. For dataset I it is seen that Basin 5 has a much smaller observation variance when compared to those of basins 2 to 4, indicating less variable data. Additionally, it is seen that basin 2 has a much smaller evolution variance when compared to basins 3 to 5, indicating a smoother trend component. Similarly for dataset II, it is seen that basin 3 has a much higher observation variance than basins 2 and 5, while basin 2 has a much higher evolution variance.

Thus far model components between basins have been assumed independent, but in reality these models will not be independent, and the dynamic trend level components will be correlated. This correlation is taken into consideration by allowing the off-diagonal covariance elements in the variance matrix, $\sum_{\eta}$, corresponding to the trend level components to be different from zero. These values are also calculated using maximum likelihood. The form of model M1 without the assumption of independent models between basins, call this model N1, generates estimated covariances that imply correlations between the trend components for basins 2 to 5 as follows:

Table 4.21: Table showing correlations between trend level components for basins 2 to 5

	Basin 5	**Basin 4**	**Basin3**	**Basin 2**
Basin 5	1	0.618	0.656	0.409
Basin 4	0.618	1	0.730	0.482
Basin 3	0.656	0.730	1	0.442
Basin 2	0.409	0.482	0.442	1

The overall AIC of model N1 is shown in Table 4.23, and is seen to be better than that seen in model M1, where the same model form for each basin is constructed independently. This is expected due to the reasonably strong correlations observed between basins in Table 4.19, and again Table 4.21 shows strong correlations between trend level components for the different basins. Note that model fit and diagnostic measures per basin as seen for model M1 are not constructed here, due to the fact that models are non-independent and the only measure of fit calculated is the overall AIC measure. Constructing a multivariate model assuming non-

independent model components, such as N1, is seen to be beneficial. It improves model fit and develops a unified way to modelling the basins of lake Kariba through considering correlations between the basins.

The form of model M2 for dataset II, without the assumption of independent model components between basins, call this model N2, generates estimated covariances that imply correlations between the trend components for basins 2, 3 and 5 as follows:

Table 4.22: Table showing correlations between trend level components for basins 2, 3 and 5

	Basin 5	**Basin 3**	**Basin 2**
Basin 5	1	0.518	0.325
Basin 3	0.518	1	0.555
Basin 2	0.325	0.555	1

The overall AIC of model N2 is shown in Table 4.24, and is seen to be better than that seen for model M2. This is again expected due to the reasonably strong correlations observed between basins in Table 4.19, and again Table 4.22 shows reasonably strong correlations between trend level components for these different basins. Thus constructing a multivariate model N2 is again seen to be beneficial, and basins are clearly not independent.

Multivariate models can be useful when modelling systems consisting of more than one time series. It provides a unified approach to modelling a system as a whole. Multivariate models can theoretically be just as flexible as univariate models, as the form of the model does not have to be the same across the series considered in the multivariate model. Multivariate models were, however, much more difficult to implement. The `dlm` package provides the `dlmSum` function in which to build multivariate models, however it is a limited function that does not allow for regression components to be added. Confusingly, this function states it does consider dynamic multivariate models, yet it does include time-varying trend components. Multivariate models often return errors of non-convergence,

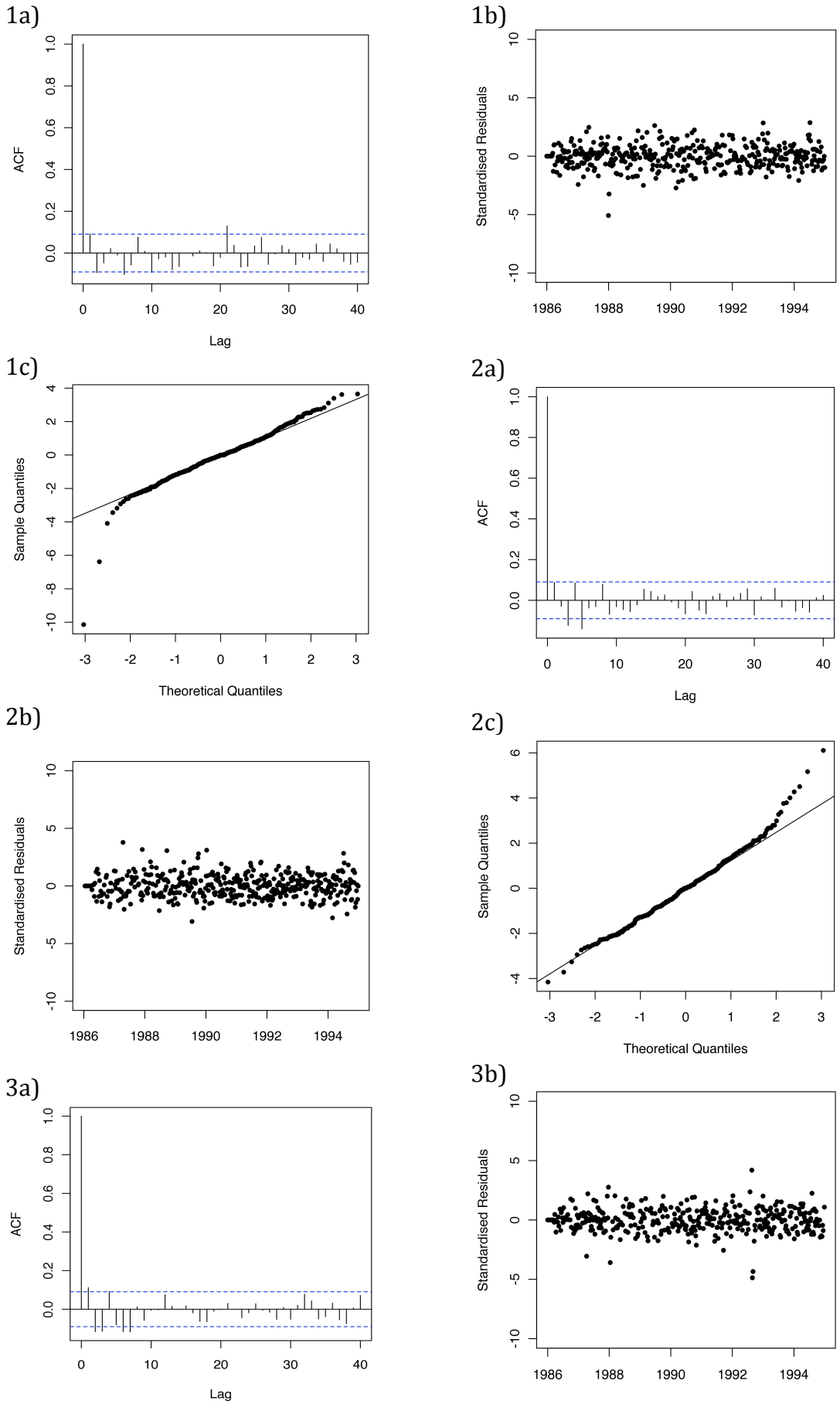


Figure 4.37: 1(a) Autocorrelation function, 1(b) Scatter plot of standardised residuals and 1(c) QQ-plot of Model M1 for basin 4
2(a) Autocorrelation function, 2(b) Scatter plot of standardised residuals and 2(c) QQ-plot of Model M1 for basin 3
3(a) Autocorrelation function, 3(b) Scatter plot of standardised residuals and 3(c) QQ-plot of Model M1 for basin 2

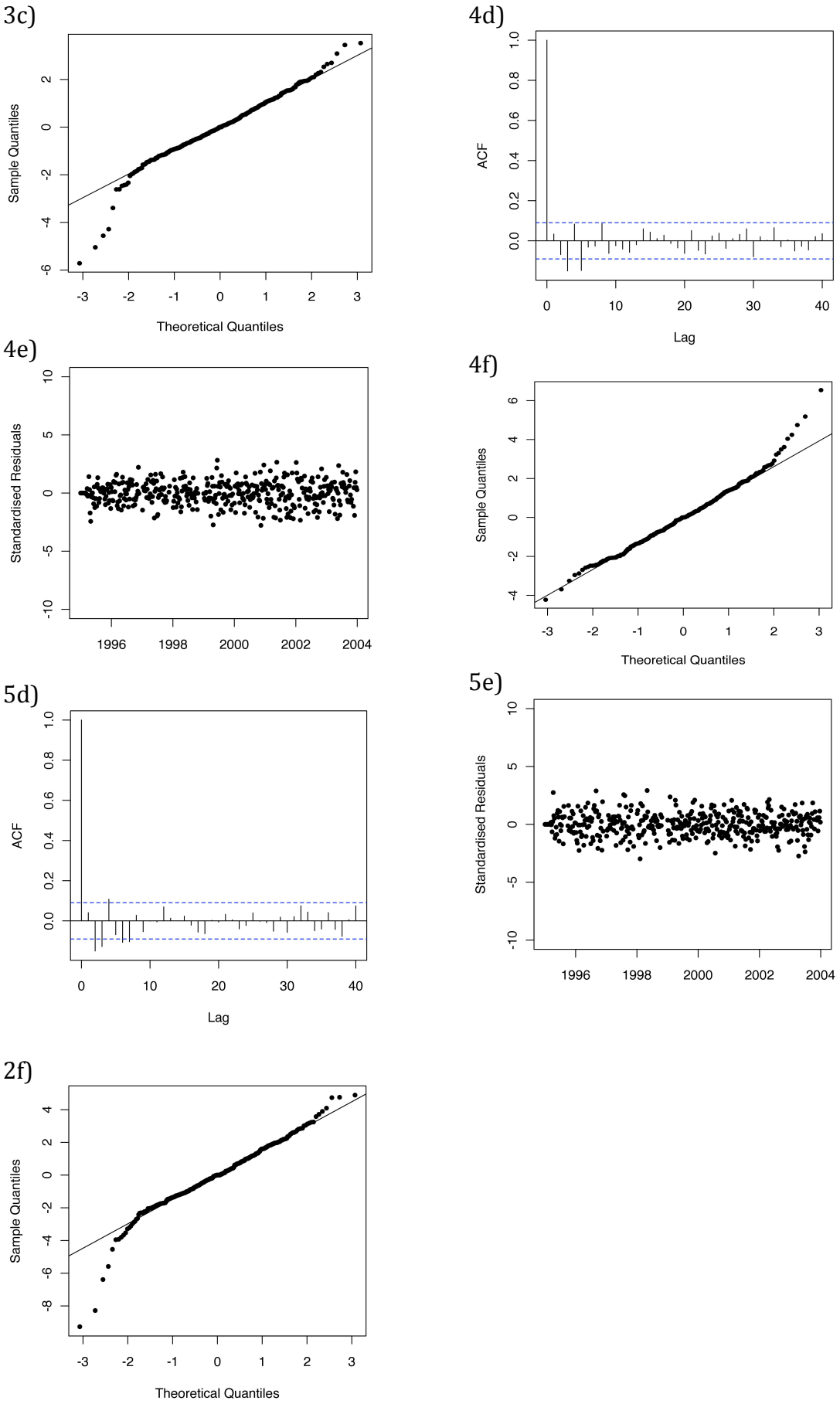


Figure 4.37: 4(d) Autocorrelation function, 4(e) Scatter plot of standardised residuals and 4(f) QQ-plot of Model M2 for basin 3
5(d) Autocorrelation function, 5(e) Scatter plot of standardised residuals and 5(f) QQ-plot of Model M2 for basin 2

Table 4.23: Table showing fit values for multivariate model M1 and N1

Model	Basin	AIC1	AIC2	P.E.V	R^2	Overall AIC
M1	*Five*	-948.17	-770.87	0.039	0.77	-1062.46
	Four	-646.28	-468.98	0.061	0.53	
	Three	-620.66	-443.36	0.069	0.35	
	Two	-456.54	-279.25	0.111	-0.09	
N1						-2002.17

Table 4.24: Table showing fit values for multivariate model M2 and N2

Model	Basin	AIC1	AIC2	P.E.V	R^2	Overall AIC
M2	*Five*	-960.25	-782.95	0.036	0.67	-1837.49
	Three	-654.78	-477.48	0.067	0.56	
	Two	-754.36	-577.06	0.055	0.59	
N2						-1881.28

for reasons not always understood. This limits the number of models that can be constructed as multivariate models. This was never a problem experienced constructing univariate models. Additionally, constructing multivariate models that assume non-independence between model components in `dlm` uses the log-Cholesky parameterisation of the matrix $\sum_{\eta}$, and constructing these models and estimating unknown model parameters became increasingly complex. Goodness-of-fit and diagnostic measures described in Harvey (1989) are all considered for the univariate case only, and can only be applied in the multivariate case across individual independent models. Multivariate models are thus concluded to be advantageous in cases where series are highly correlated, but this may become increasingly complex as more series are added to the analysis.

4.11 Chapter Conclusion

This chapter provided a description of the data used for the application of structural time series models. Total daily Kapenta catch data, divided by the total number of hauls per day are used to generate a variable known as the daily catch per unit effort (CPUE). Additional covariate data, such as daily mean temperature, precipitation, moonlight, cloud cover and lake level data is also available. This chapter focuses predominantly on the modelling of CPUE for basin 5. Preliminary exploratory data analysis shows that CPUE data follows different patterns over the two halves of the data, and thus the two halves are modelled separately. Prominent monthly and yearly cyclical components are also seen. The monthly cycle is thought to correspond with the phase of the moon, as moonlight and CPUE are related. Temperature also showed to be related to CPUE, with other covariates showing no visual evidence of relation.

Models constructed using daily data produced models with poor diagnostics and fit; the reason for this was a sampling frequency that was too high. Summing daily data into weekly data eliminated this problem, and more meaningful patterns could be extracted from the data. Both halves of the CPUE dataset are modelled as local level models, with two cyclical components where period equals 4.218

(=29.53/7) and 52.143 (=365/7) weeks respectively, the first being fit with two harmonics and the second with 3 harmonics respectively. Additionally the second half of the dataset is modelled with a temperature covariate. This is not seen in the model for the first half of the dataset, as precipitation and temperature data is only available over the second half.

Traditionally unknown parameter estimates are calculated through maximum likelihood algorithms, but this thesis also estimates these parameters using a Bayesian approach. Both methods produce similar estimates for unknown parameters.

This chapter concludes with an extension of the univariate model for CPUE of basin 5 into a multivariate model for CPUE of basins 2 to 5. Basin 1 is not considered, as it contains too many missing observations. Non-convergence is a problem experienced frequently when constructing multivariate models using the `dlm` package, making these models difficult to construct. For this reason basin 4 had to be eliminated from the multivariate analysis for dataset II. No diagnostic or goodness-of-fit measures are available for multivariate models in the `dlm` package, with an exception of a loglikelihood value from which the AIC can be calculated, known as overall AIC. Model diagnostics and fit values can be generated for each of the basin in the multivariate model, when covariances between model components are assumed independent. When model components are not assumed independent, a multivariate model is seen to be beneficial in this thesis, as overall AIC fit values decrease. This is due to the reasonably high correlated data between the basins of Lake Kariba. When model components are assumed independent, it is observed that the form of model D1 provides a reasonable fit across all basins considered in the multivariate model, signaling that components determining CPUE for basin 5 are similar for basins 2 to 4 as well.

Chapter 5

Conclusions

5.1 Structural Time Series Models: Summary and Conclusions

The objective of this thesis has been to investigate time series models for non-stationary data, namely ARIMA and structural time series models. These models were each investigated in terms of theory, their advantages and disadvantages. Having investigated these for both model types, structural time series models were selected for modelling Kapenta catch data. When using structural time series models, the data do not need to be rendered stationary beforehand and thus provide superior interpretive power. These models allow for greater flexibility, as all model components can be included as time varying. Additionally, missing values are easily handled within the structural framework. ARIMA models are criticized by Harvey (1997), for not being embedded within any underlying theoretical model or structural relationship. Additionally, identification techniques using autocorrelation functions require experienced individuals. Specifying structural time series models, although complex theoretically, allow for the construction of a model with trend, cyclical and regression components. All these components are separately modelled and directly interpreted, a very intuitive means of constructing a time series model. The main advantage of using structural

time series models over ARIMA models is the fact that meaningful model parameters are specified, and one has control of the manner in which components with these parameters are constructed. The vast literature available on these models, and the associated Kalman filtering and smoothing algorithms, allowed for ease of theoretical understanding and available programming packages in R simplified the implementation of these models. Structural time series models have only basic model assumptions of independent, homoscedastic and normally distributed standardised residuals. These assumptions are easily assessed through the use of autocorrelation functions, scatter and QQ-plots. Various methods provided in Harvey (1989) are available to assess model fits, namely R_D^2, prediction error variance (P.E.V) and AIC values. Unknown model parameters are estimated using maximum likelihood, or alternatively from a Bayesian approach using a Gibbs sampler. This thesis cannot conclude on the superiority of the frequentist versus the Bayesian approach, or visa versa, as both approaches generated the same estimates. Multivariate structural time series models are theoretically easily derived, through a direct extension of univariate model formulations. Model diagnostics and goodness-of-fit values are however only discussed for the univariate case in Harvey (1989), and only AIC values could be derived. Should independence between model components be assumed, parameter estimates corresponding to each of the basins in the multivariate model can be used in constructing a univariate model for each basin. Univariate measures of fit and diagnostics can then be observed for each of the basins, and the form of the model constructed for basin 5 can be tested for relevance for the other basins of Lake Kariba. When the assumption of independent model components is not imposed, model fit as measured by the multivariate AIC value improves when time series in the multivariate analysis are correlated. Note that each basin cannot be assessed using univariate measures of fit and diagnostics when models are no longer assumed independent. The multivariate approach provides a unified way to modelling systems as a whole.

5.2 Kapenta Fishing Application: Summary and Conclusions

The dataset used in this thesis included daily total catch of Kapenta (in kg's) for each of the 5 basins of Lake Kariba over the period 1 January 1986 to 31 December 2003. Additionally the total number of hauls per day for each basin is also provided; dividing total catch by the number of hauls defines the variable to be modelled, daily Kapenta catch per unit effort (CPUE). This thesis focused on modelling the log of Kapenta CPUE for basin 5, the basin with the greatest availability of data and in the area where covariate data (temperature, precipitation, moonlight, lake level and cloud cover) were collected. The data is modelled in two halves, datasets I and II respectively, due to the fact that the two halves of the dataset appeared to follow different underlying patterns and because explanatory variables temperature and precipitation were only consistently available over the second half of the data. Initial exploratory data analysis showed the presence of a yearly cyclical component, a monthly cyclical component (due to a strong relationship between moonlight and CPUE) and a relationship between temperature and CPUE. Using structural time series models to model daily CPUE data of datasets I and II, generated poor models. Due to the high sampling frequency, patterns in the data were difficult to distinguish and daily data were summed to generate weekly data instead. The best models for describing weekly Kapenta CPUE values were selected for both halves of the dataset. The best model in explaining weekly CPUE for dataset I, is a local linear model with Fourier form cyclical components, with periods equal to (29.53/7) and (365/7) weeks respectively. Additionally these cyclical components contain two and three harmonics respectively. The best model, model D2, for dataset II, contains the same components as model D1, but additionally also contains a temperature covariate. Parameter estimates (observation and evolution variances) differ slightly between datasets I and II, with dataset I showing slightly higher observation variance and dataset II showing higher evolution variance. A smoother trend is thus seen for dataset I. The trend component is modelled using local linear models, and no slope component is included. Therefore an overall increase or decrease in Kapenta CPUE is not significantly observed for the period

1 January 1986 to 31 December 2003, and Kapenta stocks appear to be remaining level over these years. Basin 5 is situated on the Zimbabwean side of Lake Kariba, where the fishing industry is very controlled. Kolding suggests that this has allowed for the fishing pressure that has not changed much over the years (Kolding et al, 2003). The yearly cyclical component is attributed to the move of Kapenta fish to protected bays in summer months for breeding, and back to open waters in winter months when catches start to increase again. The moonlight cyclical component exists as fishing takes place in the evening with light attractions, and increased moonlight diminishes the efficacy of these. Much of the relationship between temperature and CPUE is captured through the yearly cyclical component included in the model, as the movement of fish to protected bays and back to open waters corresponds highly with temperature. However, the inclusion of temperature in the model for dataset II is significant, and this component captures changes in CPUE attributed to temperatures that diverge from the norm. Water level, precipitation and cloud cover covariates are concluded to not significantly influence CPUE values over the period studied. When considering the multivariate models for datasets I and II that assume independent model components, it appears that models of the form described in D1 provide good fits and reasonable diagnostics for basins 2 to 4. Factors influencing CPUE for basin 5 thus seem to also influence CPUE for basins 2,3 and 4. The effect of temperature and other covariates across the basins of Lake Kariba could, however, not be included due to problems experienced with the `dlm` package in R. When the assumption of independent model components is not imposed, the multivariate model fit is seen to improve, and this is due to the fact that movements between the CPUE data for basins 2 to 5 are correlated, and are in fact not independent.

5.3 R `dlm` Package: Summary and Conclusions

This package developed by Petris (2010) provided an easy and flexible means of implementing structural time series models. All model components are individually constructed, with the option of rendering them dynamic. Filtering and smoothing algorithms are developed within the package, and thus easily performed. Additionally, parameter estimates are generated using maximum

likelihood within the package. A great amount of literature is available on this package, including a book on the package itself (Petris et al., 2007), making the implementation process more understandable. Having also considered packages such as `KFAS` (Helske, 2011) and `dlmodeler` (Szymanski, 2012), in addition to the `StructTS` functions of the `stats` package in R (Ripley, 2002), the `dlm` package provided a more flexible means of specifying a model when compared to `StructTS` and `dlmodeler`, and provides more available functions applicable to structural time series models than the `KFAS` package. However, a few drawbacks of the `dlm` package were noted. The available function for implementing a Gibbs sampler cannot include missing values in the data. For this reason, code adapted from Lavine (n.d.) was used to implement a Gibbs sampler for estimating unknown model parameters; this code can be seen in Appendix D. Loglikelihood estimates obtained from the `dlm` package are different to those calculated using the loglikelihood formulae in (3.5.3), the most frequently observed means for calculating loglikelihood. Thus code adapted from the `KFAS` package was utilised to generate this preferred log-likelihood; this adapted code can be seen in Appendix B. The `dlmSum` function is used to create multivariate models. Petris et al. (2007) defines this function to be non-applicable to time varying models. This function returns errors referring to this limitation when explanatory variables are included into the model, whether they are dynamically included or not. Additionally, this function was seen able to include time varying trend components, despite the fact that this function is described not to support time varying models. Lastly, for models containing a regression component, the F_t matrix (using the notation of the `dlm` package), or similarly the Z_t matrix, does not include the explanatory variable in this matrix, but rather separately in a variable called `X`. The Z_t matrix including the explanatory variable is needed for calculating the loglikelihood, the P.E.V and for generating unknown parameter estimates from a Bayesian perspective. Code had to be adapted to redefine Z_t to include explanatory variables. Constructing multivariate models with unknown model parameters, and assuming non-independent models components, proved difficult and could only be done using the log-Cholesky transformation. The specification of multivariate models in this way is not as intuitive as the specification performed for univariate models.

5.4 Future Extensions and Recommendations

In terms of investigating time series models, this thesis considered a detailed look into univariate structural time series models. The multivariate extension was only briefly considered and more extensive multivariate models, perhaps with not the same model form across each of the basins, can be constructed. Another multivariate extension to consider could include constructing a multivariate model in which the cyclical components are exactly the same across the basins of Lake Kariba, i.e. cycles rise and fall at the exact same times. Additionally comprehensive goodness-of-fit and diagnostic measures for multivariate models, which are only considered for univariate models in this thesis, would be recommended for investigation in the future. Another interesting extension would be to build as extensive an independent model considered for basin 5 for basins 4 to 1. In this way, it can be ascertained more accurately whether Kapenta catches are significantly increasing or decreasing over time in the other basins. Cyclical components and explanatory variables can also be investigated to see which phenomena remain consistent in explaining CPUE through all basins, and which impact only certain basins. Lastly, an interesting extension would be to construct models for CPUE for each half of the lake, one for the Zimbabwean side and one for the Zambian side. This may assist in telling if either of the authorities are over-fishing, by seeing significant decreases in CPUE over time by splitting the data as such. Different factors may affect CPUE on either side of the lake, and differing factors between authorities might generate interesting results.

References

Ansley, C. F. and Kohn, R., 1985. Estimation, Filtering, and Smoothing in State Space Models with Incompletely Specified Initial Conditions. Annals of Statistics, 13, 1286–1316.

Bourdillon, M.F.C., Cheater, A.P. and Murphree, M.W., 1985. Studies of Fishing on Lake Kariba. Mambo Occasional Papers: Socio-Economic Series, 20, 185.

Box, G.E.P. and Jenkins, G.M., 1970. Time Series Analysis, Forecasting and Control. Revised Edition 1976. Holden-Day.

Brockwell, P.J. and Davis, R.A., 1991. Time Series: Theory and Methods. Springer.

Brown, R.G., 2004. Smoothing, Forecasting and Prediction of Discrete Time Series. Courier Dover Publications.

Byrd, R.H., Lu, P., Nocedal, J. and C. Zhu., 1995. A Limited Memory Algorithm for Bound Constrained Optimisation. SIAM Journal on Scientific Computing, 16(5), 1190-1208.

Carter, C.K. and Kohn, R., 1994. On Gibbs Sampling for State Space Models. Biometrika, 81(3), 541-553.

Chatfield, C., 2004. The Analysis of Time Series: An Introduction. 6th Edition. Chapman and Hall/CRC.

Chifamba, P.C., 2000. The Relationship of Temperature and Hydrological Factors to Catch per unit Effort, Condition and Size of the Freshwater Sardine, *Limnothrissa miodon*, (Boulenger), in Lake Kariba. Fisheries Research, 45, 271–281.

Clark, P. K., 1987. The Cyclical Component of U.S. Economic Activity. Quarterly Journal of Economics, 102(4), 797–814.

Commandeur, J.J.F., Koopman, S.J., Ooms, M., 2011. Statistical Software for State Space Methods. Journal of Statistical Software, 41(1), 1–18. URL http://www.jstatsoft.org/v41/ i01/.

De Jong, P., 1988. A Cross–Validation Filter for Time Series Models. Biometrika, 75 (3), 594–600.

De Jong, P. and MacKinnon, M., 1988. Covariances for Smoothed Estimates in State Space Models, Biometrika 75, 601 – 602.

Durbin J., 1987. Statistics and Statistical Science. Journal of the Royal Statistical Society, Series A, 150(3), 177-191.

Durbin, J., 1988. Is a Philosophical Consensus for Statistics Attainable? Journal of Econometrics, 37, 51-61.

Durbin, J. and Koopman, S.J., 2001. Time Series Analysis by State Space Methods. Oxford Statistical Science Series. Oxford University Press.

Fama, E.F. and Gibbons, M.R., 1982. Inflation, Real Returns, and Capital Investment. Journal of Monetary Economics, 9(3), 297–323.

Früwirth-Schnatter, S., 1994. Data Augmentation and Dynamic Linear Models. Journal of Time Series Analysis, 15, 183-202.

Gamerman, D., 1997. Markov Chain Monte Carlo: Stochastic Simulations for Bayesian Inference. Chapman and Hall.

Geweke, J., 1989. Bayesian Inference in Econometric Models using Monte Carlo Integration. Econometrica, 57, 1317-39.

Hamilton, J. D., 1994. State-space models. Engle, R.F. and McFadden, D.L. (Eds.), Hand-book of Econometrics, 4(50), 3039–3080.

Harvey, A.C. and Phillips, G.D.A., 1979. Maximum Likelihood Estimation of Regression Models with Autoregressive-Moving Average Disturbances. Biometrika, 66(1), 49–58.

Harvey, A.C., 1985. Trends and Cycles in Macroeconomic Time Series. Journal of Business & Economic Statistic, 3(3), 216-227.

Harvey, A. C., 1989. Forecasting, Structural Time Series Models and the Kalman Filter. Cambridge University Press.

Harvey, A. C. and Shephard, N., 1993. Structural Time Series Models. pp.261-302. In Maddala, G.S., Rao, C.R. and Vinod, H.D. (Eds.), Handbook of Statistics, 11. Elsevier Science Publishers B. V.

Harvey, A.C., 1997. Trends, Cycles and Autoregressions. The Economic Journal, 107(440), 192-201.

Harvey, A. C., Koopman, S. J. and Penzer, J. M., 1998. Messy Time Series: A Unified Approach. Advances in Econometrics, 13, 103-143.

Harvey, A.C., 2000. Trend Analysis. University of Cambridge, Faculty of Economics and Politics, Manuscript.

Harvey, C., 2003. Time Series Models. 2nd Edition. Harvester Wheatsheaf.

Helske, J., 2010. KFAS: Kalman filter and Smoothers for Exponential Family State Space Models. R package version 0.5.1, URL http://CRAN.R-project.org/package=KFAS.

Horky, P., Slavik, O., Bartos, L., Kolariva, J. and Randak, T., 2006. Folia Zoologica, 55(4), 411-417.

Jackson, P.B.N., 1961. Ichthyology, the fish of middle Zambezi. Kariba Studies, 1, 1-36.

Jalles, J.T., 2009. Structural Time Series Models and the Kalman Filter: A Concise Review. FEUNL Working Paper No. 541. Available at: http://dx.doi.org/10.2139/ssrn.1496864 [Accessed 17 October 2012].

Karange, L.P. and Harding, E., 1995. On the Relationship between Hydrology and Fisheries in man-made Lake Kariba, Central Africa. Fisheries Research, 22, 205–226.

Kohn, R. and Ansley, C.F., 1989. A Fast Algorithm for Signal Extraction, Influence and Cross- Validation in State Space Models. Biometrika, 76(1), 65–79.

Kolding, J., Musando, B., and Songore, N., 2003. Inshore Fisheries and Fish Population Changes in Lake Kariba. In: Jul-Larsen, E., Kolding, J., Nielsen, J.R.,Overa, R. and van Zwieten, P.A.M. (eds.), Management, Co-management or no Management? Major dilemmas in Southern African freshwater fisheries. Part 2: Case studies, FAO Fisheries Technical Paper, 426(2), 67-99.

Koopman, S.J., 1997. Exact Initial Kalman Filtering and Smoothing for Nonstationary Time Series Models. Journal of the American Statistical Association, 92(440), 1630–1638.

Koopman, S.J., Shephard, N. and Doornik, J.A., 1999. Statistical Algorithms for Models in State Space using SsfPack 2.2. Econometrics Journal, 2(1), 113–166.

Lavine, M., n.d. State Space Markov Models, ST 797M. Available at: http://www.math.umass.edu/~lavine/courses/797/stat797.html [Accessed 16 October 2012].

Litterman, R., 1986. Forecasting with Bayesian Vector Autoregressions - Five Years of Experience. Journal of Business and Economic Statistics, January, 4(1), 25-38.

Madamombe, L., 2002. The Economic Development of the Kapenta Fishery Lake Kariba (Zimbabwe/Zambia). [pdf] Norwegian College of Fishery Science, University of Tromso. Available at: http://munin.uit.no/bitstream/handle/10037/336/thesis.pdf?sequence=1 [Accessed 15 October 2012].

Magadza, C.H.D., 1980. The Distribution of Zooplankton in the Sanyati Bay, Lake Kariba; a Multivariate Analysis. Hydrobiologia, 70, 57-67.

Magadza, **C.H.D**., 1996. Climate Change: Some Likely Multiple Impacts in Southern Africa. In: Downing,T.E.(ed.), Climate Change and World Food Security. Springer-Verlag, Dordrecht, 449–483.

Marshall, B.E., 1982. The Influence of River Flow on Pelagic Sardine Catches in Lake Kariba. Journal of Fish Biology, 20, 465–470.

Marshall, B.E., 1988. Seasonal and Annual Variations in the Abundance of Pelagic Sardines with Special Reference to the Effects of Droughts. Archives of Hydrobiology, 112, 399–409.

Mergner, S., 2009. Applications of State Space Models in Finance. Universitätsverlag Göttingen. Available at: http://webdoc.sub.gwdg.de/univerlag/2009/mergner.pdf [Accessed: 27 November 2012].

Meyler, A., Kenny, G., and Quinn, T., 1998. Forecasting Irish inflation using ARIMA models. Technical Paper 3/RT/98. Central Bank of Ireland: Economic Analysis, Research, and Publication Department.

Mtada, O.S.M., 1987. The influence of thermal stratification on pelagic fish yields in Lake Kariba, Zambia/Zimbabwe. Journal of Fish Biology, 30(2), 127–133.

Ndebele-Murisa, M.R., Mashonjowa, E. and Hill, T., 2011. The Implications of

a Changing Climate on the Kapenta Fish Stocks of Lake Kariba, Zimbabwe. Transactions of the Royal Society of South Africa, 66(2), 105-119.

Petris, G., Petrone, S. and Campagnoli, P., 2007. Dynamic Linear Models with R. Springer.

Petris, G., and Petrone, S., 2011. State Space Models in R. Journal of Statistical Software, 41(4), 1-24.

Petris, G., 2009. dlm: an R Package for Bayesian Analysis of Dynamic Linear Models. [pdf] University of Arkansas, Fayetteville AR. Available at: http://cran.r-project.org/web/packages/dlm/vignettes/dlm.pdf [Accessed 16 October 2012].

Petris, G., 2010. dlm: Bayesian and Likelihood Analysis of Dynamic Linear Models. R package version 1.1-1, URL http://CRAN.R-project.org/package=dlm.

Pfaff, B., 2006. Analysis of Integrated and Cointegrated Time Series with R. Springer-Verlag, New York. URL http://CRAN.R-project.org/package=urca.

Pinheiro, J.C., and Bates, D.M., 1996. Unconstrained Parameterizations for Variance-Covariance Matrices. Statistics and Computing, 6, 289–296.

Ripley, B.D., 1987. Stochastic Simulation. Wiley.

Ripley, B.D., 2002. Time Series in R 1.5.0. R News, 2(2), 2–7. URL http://CRAN. R-project.org/doc/Rnews/.

Rosenberg, B., 1973. Random Coefficients Models: the Analysis of a Cross-Section of Time Series by Stochastically Convergent Parameter Regression. Annals of Economic and Social Measurement, 2(4), 399–428.

Shephard, N., 1994. Partial Non-Gaussian State Space Models. Biometrika, 81, 115-131.

Stock, J. H. and Watson, M.W., 1991. A Probability Model of the Coincident Economic Indicators. In: Lahiri, K. and Moore, G.H, (Eds.), Leading Economic Indicators: New Approaches and Forecasting Records, 63–89. Cambridge University Press.

Stockton, D., and Glassman, J., 1987. An Evaluation of the Forecast Performance of Alternative Models of Inflation. Review of Economics and Statistics, 69(1), 108-117.

Szymanski, Cyrille., 2012. Dlmodeler: Generalised Dynamic Linear Models. R package version 1.2.1, URL http://CRAN.R-project.org/package=dlmodeler

West, M. and Harrison, J., 1997. Bayesian Forecasting and Dynamic Models. 2nd edition. Springer-Verlag.

Appendix A

```
## Building a model for weekly log CPUE data for basin 5 of
##lake Kariba, with a local linear trend, a cyclical component
##with period=29.53/7 and 2 harmonics, a cyclical component
##with period=365/7 and 3 harmonics and a regression term of
##weekly temperature data. Unknown observation and Evolution
##variances are estimated using maximum likelihood
#_____________________________________________________________

buildit <- function (par)
 { mod <- dlmModPoly(1)+dlmModTrig(tau=(29.53/7),q=2)
          +dlmModTrig(tau=(365/7), q=3)
          +dlmModReg(weeklyTemp, addInt=FALSE)
  V(mod) <- exp(par[1])
  diag(W(mod))[1] <- exp(par[2])
  return(mod)
 }

fit2 <- dlmMLE (data, par = c(1,1), buildit )
dlmY <- buildit(fit2$par)
V(dlmY)                  #Observation Variance
W(dlmY)                  #Matrix with Evolution Variance

filt <- dlmFilter(data, dlmY)          #Kalman Filter
smooth<-dlmSmooth(data, dlmY)          #Smoother
```

Appendix B

```
#Model Checks
#_____________________________________________________________
#Values to be Specified
#_____________________________________________________________

d <- 1                   # no. of time varying parameters in
                           state vector
s <- 2                   # no. of non zero parameters
m <- 12                  # length of matrix 'W' diagonal
q <- 11                  # No. of variables that are not non-
                           constant(m minus no. of regressors)
```

```r
##(1) Log-Likelihood
#_____________________________________________________________

{if (length(dim(X(dlmY))) == 0)
Zt=dlmY$FF
else {
cols<-matrix(rep(dlmY$FF[,1:q], length(data)),length(data),q,
TRUE)
newX<-cbind(cols, X(dlmY))
newX1<-unmatrix(newX, TRUE)
Zt<-array(newX1, dim=c(1,m,length(data)))}}

LaraLL<-function (yt, model, raw.result = FALSE, logLik =
TRUE, filter = TRUE)      ## (Derived from likelihood function
                               in KFAS package)
{
    if (!require("KFAS"))
        stop("required package could not be found: KFAS")
    res <- KFAS::kf(yt = yt, Zt = Zt, Tt = dlmY$GG, Rt =
    diag(1,m,m), Ht = dlmY$V, Qt = dlmY$W, a1 = t(dlmY$m0), P1
    = diag(0,m,m),P1inf = diag(1,m,m), optcal = c(FALSE,
    FALSE,   FALSE, FALSE))
    if (raw.result)
        raw.res <- res
    else raw.res <- NA
        return(list(backend = "KFAS", at = res$at, Pt =
         res$Pt, logLik = res$lik, raw.result = raw.res))
}

LL<-LaraLL(data, dlmY)
LL$logLik

LL2<-dlmLL(data, dlmY)
LL2

##(2) AIC
##_____________________________________________________________

npar<- m+d
Aic1<-(-2*LL$logLik)+(2*npar)
Aic1

Aic2<-(-2*(-LL2))+(2*npar)
Aic2

##(3) PREDICTION ERROR VARIANCE
#_____________________________________________________________
```

```
Rt <- with(filt, dlmSvd2var(U.R, D.R))
Ft <- numeric()

{if (length(dim(X(dlmY))) == 0)
 for (i in 1:length(data))
 { Ft[i] <- Zt %*% Rt[[i]] %*% t(Zt) + V(dlmY)
 }
 else {for (i in 1:length(data))
 { Ft[i] <- Zt[ , ,i] %*% Rt[[i]] %*% as.vector(t(Zt[ , ,i]))
            + V(dlmY)
}}}

plot(Ft, type = "b")
EV<-Ft[length(data)]  #prediction error variance

##(4) R-squared
#_____________________________________________________________

y.delta.dev <- diff(data, na.rm=TRUE) - mean(diff(data),
na.rm=TRUE)

den <- sum(y.delta.dev^2, na.rm = TRUE)

num <- (length(data) - d) *  EV

R2 <- 1 - (num / den)

##(5) Heteroscedasticity. Note 'H' was not made use of in this
##thesis, but was graphically tested instead
#_____________________________________________________________

T <- length(data)
p <- round((T - d) / 3)
k <- d

res<-residuals(filt, type=c("raw"), sd=F)
vt<-res/(Ft^0.5)

num <- sum(vt[(T - p + 1) : T]^2, na.rm=TRUE)
den <- sum(vt[(k + 1) : (p + k)]^2, na.rm=TRUE)

H <- num / den
H

plot(vt)
```

```
##(6) Boxtest. Note, only the autocorrelation and not the
##boxtest was utilised.
#_____________________________________________________________

acf(vt, lag.max=40, na.action=na.pass)
v<-vt[-c(1:d)]
Box.test(v, lag = 10, type = "Ljung", fitdf = s - 1)
```

Appendix C

```
##Confidence Interval Of Slope Presence
#_____________________________________________________________
U.S <- smooth$U.S
D.S <- smooth$D.S

v <- dlmSvd2var(U.S, D.S)

vars <- c()

  for (i in 1:(length(data) + 1))
   { vars[i] <- v[[i]][2, 2]
   }

  pl <- smooth$s[,2] + qnorm(0.05, sd = sqrt(vars))
  pu <- smooth$s[,2] + qnorm(0.95, sd = sqrt(vars))

  ymin <- min(pl)
  ymax <- max(pu)

jpeg("Graph45.jpeg", width=5, height=5, units="in", res=500)
plot(Date2,smooth$s[-1, 2], col = 2, ylim = c(ymin, ymax), lty
= 4, type = "l",ylab = "slope")

lines(Date2,pl[-1], lty = 2)
lines(Date2,pu[-1], lty = 2)

abline(h = 0, lty = 3, col = "grey")
dev.off()


### Confidence Interval for changes in trend level component
#_____________________________________________________________
```

```
U.C <- filt$U.C
D.C <- filt$D.C

U.R <- filt$U.R
D.R <- filt$D.R

U.S <- smooth$U.S
D.S <- smooth$D.S

n.obs <- length(smooth$s[-1, 1])
dlmWfp <- dlmY


Cov.filter <- dlmSvd2var(U.C, D.C)

Cov.smooth <- dlmSvd2var(U.S, D.S)

Cov.one.ahead <- dlmSvd2var(U.R, D.R)

n <- n.obs

T <- dlmWfp$G
Cov.diff <- NULL
Cov.diff[[n]] <- Cov.smooth[[n+1]]

for (i in (n-1):1)

 { c.diff <- Cov.filter[[i+1]] %*%
             t(T) %*%
             solve(Cov.one.ahead[[i+1]]) %*%
             Cov.diff[[i+1]]

   Cov.diff[[i]] <- c.diff
 }


x2 <- smooth$s[-1, 1]
trend.diffs <- x2 - x2[n.obs]

v <- c()

for(i in 1:n.obs)
   { v[i] <- Cov.smooth[[n.obs + 1]][1, 1] + Cov.smooth[[i +
             1]][1, 1] - 2 * Cov.diff[[i]][1, 1]
   }
```

```
trend.diffs <- ts(trend.diffs, start = c(1986, 1), frequency =
1)
  pl <- trend.diffs + qnorm(0.05, sd = sqrt(v))
  pu <- trend.diffs + qnorm(0.95, sd = sqrt(v))

ymin <- min(pl)
ymax <- max(pu)

jpeg("Graph46.jpeg", width=5, height=5, units="in", res=500)
plot(Date2, trend.diffs, ylim = c(ymin, ymax), ylab =
"previous - current level", type='l')

abline(h = 0, lty = 3)

  lines(Date2,pl, lty = 2, col='red')
  lines(Date2,pu, lty = 2, col='red')

  title(ylab = "year", outer = TRUE)
  title(xlab = "Time", cex = 1.5, outer = TRUE)

dev.off()
```

Appendix D

```
## Baysian Analysis: Example seen from Lavine (n.d)
#_____________________________________________________________

mod    <- dlmY
data   <- data
n.data <- length(data)       # number of time points
MC     <- 10000              # number of MCMC iterations
q      <- 11                  # same as in appendix B
keep   <- 1:MC
n.keep <- length(keep)      # number of iterations to save

# PARAMETERS
# Theta:      state vector
# Tau:        observation precision
# Winvpoly:   evolution precision

# PRIORS

tau.a       <- 0; tau.b       <- 0
winvpoly1.a  <- 0; winvpoly1.b  <- 0
```

```
# STORAGE FOR MCMC OUTPUT

Theta.Gibbs     <- array ( NA, dim = c ( n.keep, n.data+1,
                   length(dlmY$m0) ) )
Tau.Gibbs        <- rep ( NA, n.keep )
Winvpoly1.Gibbs  <- rep ( NA, n.keep )

{if (length(dim(X(dlmY))) == 0)
Zt=dlmY$FF
else {
cols<-matrix(rep(dlmY$FF[,1:q], length(data)),length(data),q,
TRUE)
Zt<-cbind(cols, X(dlmY))
}}

# THE SAMPLER

for ( i in 1:MC ) {
     print ( paste ( "beginning loop", i ) )
     # sample theta
     filt       <- dlmFilter ( data, dlmY)
     theta      <- dlmBSample ( filt )
     theta.pred <- theta[-n.data+1,] %*% t(mod$GG)
     theta.res  <- theta[-1,] - theta.pred

# sample tau
     {if (length(dim(X(dlmY))) == 0)
     fit<- theta[-1,]%*%t(mod$FF)
     else{
     fit<-vector(length=length(data))
     for ( k in 1:length(data) ) {
     fit[k]   <- theta[-1,][k,]%*%(Zt[k,])}
     }}
     rss    <- sum ( (data-fit)^2, na.rm=TRUE )
     tau    <- rgamma ( 1, shape=tau.a+n.data/2,
                           rate=tau.b+rss/2
                     )
     mod$V <- 1/tau

# sample winvpoly1
     rss         <- sum ( theta.res[,1]^2 , na.rm=TRUE)
     winvpoly1   <- rgamma ( 1, shape = winvpoly1.a + n.data/2,
                                rate  = winvpoly1.b + rss/2
                           )

     if ( !is.na ( j <- match ( i, keep ) ) ) {
      Theta.Gibbs[j,,]   <- theta
```

```
      Tau.Gibbs[j]       <- tau
      Winvpoly1.Gibbs[j]  <- winvpoly1
  }
    print ( paste ( "ending loop", i ) )
}
```

Appendix E(1)

```
## Multivariate model assuming independent model #components
across basins 2 to 5
#_____________________________________________________________

build.multi1 <- function ( par ) {
mod <- (          (dlmModPoly ( order=1, dV=exp(par[1]),
dW=c(exp(par[2]))) +dlmModTrig(tau=(29.53/7), q=2, dV=0,
dW=0)+dlmModTrig(tau=(365/7), q=3, dV=0, dW=0))
                 %+% (dlmModPoly ( order=1, dV=exp(par[1]),
dW=c(exp(par[2])))+dlmModTrig(tau=(29.53/7), q=2, dV=0,
dW=0)+dlmModTrig(tau=(365/7), q=3, dV=0, dW=0))
                 %+% (dlmModPoly ( order=1, dV=exp(par[1]),
dW=c(exp(par[2])))+dlmModTrig(tau=(29.53/7), q=2, dV=0,
dW=0)+dlmModTrig(tau=(365/7), q=3, dV=0, dW=0))
                  %+% (dlmModPoly ( order=1, dV=exp(par[1]),
dW=c(exp(par[2])))+dlmModTrig(tau=(29.53/7), q=2, dV=0,
dW=0)+dlmModTrig(tau=(365/7), q=3, dV=0, dW=0))
        )
   return ( mod )
}

fit2 <- dlmMLE (matrixdata, par = c(1,1), build.multi1 )
dlmY <- build.multi1(fit2$par) #matrix data contains the log
                               of CPUE for each of the four
                               basins of lake Karibs, in a
                               length(data)x4 matrix
V(dlmY)
W(dlmY)

filt <- dlmFilter(matrixdata, dlmY)
smooth<-dlmSmooth(matrixdata, dlmY)
```

Appendix E(2)

```
## Multivariate model assuming non-independent model
#components across basins 2 to 5
#Note Log-Cholesky reparameterisation of the W matrix
#_____________________________________________________________

uni<-dlmModPoly(1)+dlmModTrig(tau=(29.53/7),q=2)+
dlmModTrig(tau=(365/7), q=3)
mod<-uni%+%uni%+%uni%+%uni
FF(mod)<-FF(uni)%x%diag(4)
GG(mod)<-GG(uni)%x%diag(4)
W(mod)[]<-0


 buildit <- function(par) {
    U <- matrix(0, nrow = 4, ncol = 4)
    U[upper.tri(U)] <- par[1 : 6]
    diag(U) <- exp(0.5* par[7:10])
    W(mod)[1:4, 1:4] <-crossprod(U)
    diag(W(mod))[5:44] <- 0
    diag(V(mod)) <- exp(0.5* par[11 : 14])
    mod
}

fit2 <- dlmMLE(matrixdata, rep(0, 14), buildit, control =
list(maxit = 500))
dlmY <- buildit(fit2$par)
filt <- dlmFilter(matrixdata, dlmY)
smooth<-dlmSmooth(matrixdata, dlmY)
```