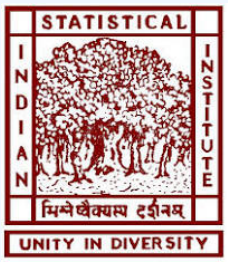
Some selected problems in discrete-valued time series analysis

Subhankar Chattopadhyay



Applied Statistics Unit
Indian Statistical Institute
December 2024

INDIAN STATISTICAL INSTITUTE

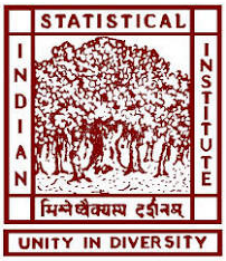
DOCTORAL THESIS

Some selected problems in discrete-valued time series analysis

---

Author:
Subhankar Chattopadhyay

Supervisor:
Prof. Atanu Biswas



A thesis submitted to the Indian Statistical Institute in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Statistics

Applied Statistics Unit

Indian Statistical Institute

December 2024

To Maa and Baba . . .

Abstract

This thesis analyzes some discrete-valued time series problems, which are classified into two types: (i) categorical time series and (ii) count time series. In this thesis, we primarily use two well-known models in the context of discrete-valued time series research: (i) Pegram's operator-based autoregressive (PAR) process, which can be used to analyze both categorical and count data; and (ii) the integer-valued autoregressive (INAR) process, which is used for modelling count time series data. In Chapter 1, we review literature on discrete-valued time series and provide brief descriptions of our research works. Chapter 2 discusses a study on categorical time series. In this chapter, we propose a generalized PAR (GPAR) process that utilizes a generalized kernel to overcome the limitation of the traditional PAR process, which solely provides weights for the same previous category. Chapter 3 consists of a study of time series with truncated counts. In this chapter, we propose a modified PAR (mPAR) process with a modified kernel to model truncated counts in order to avoid the aforementioned drawback of the traditional PAR process. In Chapter 4, we consider the problem of change-point analysis in count time series data using an INAR(1) process with time-varying covariates. We employ the Poisson INAR(1) (PINAR(1)) process with a time-varying smoothing covariate in this study. This model allows us to model both components of active cases at time-point t: (i) survival cases from the previous time-point, and (ii) the number of new cases (innovations) at time-point t. In Chapter 5, we analyze count time series data with zero-inflation and seasonality. To capture both of these features, we propose an INAR(1) process that employs zero-inflated Poisson innovations with seasonality. We investigate the distributional properties and h-step ahead forecasting of all proposed processes. We conduct extensive simulation experiments to explore the usefulness of the proposed processes. Finally, we analyze some real datasets to provide practical illustrations of our proposed methods. In Chapter 6, we summarize our findings and discuss potential future directions for these works.

Acknowledgements

First and foremost, I want to express my gratitude to the Almighty for providing me with the strength to navigate through my incredible PhD experience. I embarked on this journey in 2018. A substantial portion of this journey unfolded during the COVID pandemic, a time that brought forth many obstacles and unknowns. Thus, I wish to extend my sincere appreciation to the outstanding individuals who provided support and guidance during this transformative journey, allowing me to successfully finish my thesis despite numerous unexpected challenges.

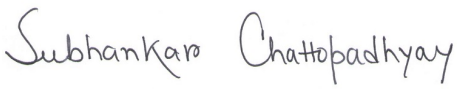
I want to express my gratitude to Prof. Atanu Biswas, my PhD supervisor. His unwavering support, direction, and motivation have been priceless throughout the entire process. Throughout the process of refining my research proposal and submitting my thesis, his consistent support has been crucial in molding my academic development. I greatly appreciated the weekly meetings we conducted, as they not only helped me stay on track academically but also offered me a lot of encouragement. I am deeply thankful for the invaluable impact he has had on my growth.

I have conducted a significant portion of my research in collaboration with esteemed faculty members at the Indian Statistical Institute, Kolkata. I am deeply grateful to Prof. Samarjit Das for his invaluable contributions to the more intricate and technical aspects of my study. The dedication of Dr. Raju Maiti has not only supported me in different aspects of my research but also enhanced my thinking about Statistics. His suggestions for enhancing my research go beyond description. I am deeply thankful to Dr. Raju Maiti for the extensive meetings and phone conversations we had during the entire process, which significantly enhanced my PhD experience and reduced stress levels. I am grateful to Prof. Jing-Shiang Hwang (Institute of Statistical Science, Academia Sinica, Taiwan) and Prof. Bibhas Chakraborty (Duke-NUS Medical School, National University of Singapore) for collaborating with me. I would also like to express my gratitude to Debika (Research Fellow, Indian Institute of Management Udaipur) for dedicating her valuable time to our joint research project during my PhD tenure. In addition to my supervisor and research collaborators, I would like to thank all of my private tutors, as well as the teachers at my school, Narendrapur Ramakrishna Mission Residential College, and the University of Calcutta. I would like to express my gratitude to all the faculty members of the Applied Statistics Unit (ASU), ISI Kolkata, particularly Prof. Tapas Samanta, for allowing me to serve as his teaching assistant for three consecutive academic years. His words always inspired me and motivated me to engage in research. I would like to express my

gratitude to all other ASU faculty members and official staff, as well as other ISI Kolkata units, for creating an outstanding research environment for me. I would also like to thank two anonymous reviewers for their insightful comments and useful recommendations, which helped to improve the thesis.

Throughout this academic pursuit, I am forever grateful to my outstanding friends, whose encouragement has been a consistent source of inspiration. Engaging in collaborative speaking sessions and informal chats with my ISI friends, whether through screens during lockdowns or in person whenever circumstances allowed, proved to be a lifeline during the most challenging times. I am fortunate to have Soumya, Monitirtha, Rahul, Subhajit, Amarnath, Sayantan, Biswadeep, Sayan, Meghna, Mriganka, Saptangshu, Anik, Chirayata, Soutik, Anirban, Arijit, Saurav, Javed-da, Sujay-da, Arijit-da, and Anurag-da as my ISI colleagues. I am pleased to say that we evolved beyond being just lab partners or colleagues; we developed a strong friendship. Aside from them, I must express my gratitude for the unwavering support of Chiranjib, my college companion. He not only shared his expertise in R-coding but also provided constant encouragement during the challenging times of this journey. I would also like to express my gratitude to Arindam, Sudipta, Sourav, and Subhradip, my childhood friends, for their constant support and motivation during this entire journey.

Finally, I would like to extend my sincerest thanks to my family for their unwavering belief in me and their continuous support. Your support was crucial to my achievements. To my Maa and Baba, thank you for everything. I dedicate this PhD thesis to you.



(SUBHANKAR CHATTOPADHYAY)

December 2024

Contents

List of Figures

List of Tables

Some Important Abbreviations

AIC	Akaike Information Criterion
CLS	Conditional Least Squares
GPAR	Generalized Pegram's operator based autogressive (AR)
INAR	Integer-valued AR
MLE	Maximum Likelihood Estimation
mPAR	modified Pegram's operator based AR
MSE	Mean Squared Error
MTDM	Mixture Transition Distribution Model
PAR	Pegram's operator based AR
PGF	Probability Generating Function
PRMSE	Predicted Mean Squared Error
PTP	Percentage of True Prediction
PTPI	PTP within an interval
RMSE	Root Mean Squared Error
ZINAR	INAR with zero-inflated Poisson innovations
ZIP	Zero-inflated Poisson

Some Important Symbols

ϕ	autoregressive parameter for the PAR(1) process
p_i	$P(Y_t = C_i)$ for the PAR(1) process
π_i	$P(\epsilon_t = C_i)$ for the PAR(1) process
Q	transition probability matrix for the MTD(1) process
K	proposed kernel for the GPAR(1) process
θ	weight assigned to proposed kernel for the GPAR(1) process
$M_{(\cdot)}$	normalizing constants for the GPAR(1) process
W	proposed kernel for the mPAR(1) process
η	weight assigned to proposed kernel for the mPAR(1) process
Λ	parameter of right-truncated Poisson distribution
α	autoregressive parameter for the INAR(1) process
β_i^{c1}	model parameters of one change-point study for time-varying components
β_i^{c2}	model parameters of two change-point study for time-varying components
β_i^{s}	model parameters for seasonal components
ρ	zero-inflation parameter for the INAR(1) process with zero-inflation and seasonality

Chapter 1

Introduction

1.1 Preamble

A time series is a sequence of data points $\{Y_t : t = 1, 2, \ldots\}$ measured at successive time intervals. Here, t denotes the time at which Y_t is observed. Some examples of time series include the monthly price of an essential commodity like petrol in India, monthly rainfall data in an area, weekly Air Quality Index (AQI) data in a city, annual crop production, monthly sales figures in a shop, daily stock prices, and so on. A time series generally reflects the fact that observations close together in time are more closely related than those further apart. Time series has a wide range of applications, e.g., signal processing, mathematical finance, economics, weather forecasting, earthquake prediction, biological and medical sciences, and engineering.

In time series analysis, there are primarily two objectives: (i) to analyze the inherent structures within data in order to extract meaningful statistics, and (ii) forecasting, in which we attempt to estimate how the sequence of observations might continue into the future. When dealing with time series data, the order in which the data points are presented is of utmost importance. Rearranging or omitting certain observations can lead to the loss of significant insights. There are largely three types of data in the field of univariate time series, namely (a) continuous data (e.g., daily maximum temperature data of a region, annual profits of a shop, etc.), (b) discrete count data (e.g., monthly dengue data in a region, weekly accidents at a busy flyover in a city, etc.), and (c) discrete categorical data (e.g., sequence of genomes, categories of some air pollutant like PM2.5 particles, etc.). In this thesis, we model and analyze both categorical and count time series data. We collectively refer to the analyses of these data as discrete-valued time series analysis.

If each data point in a time series has one of the finite or countably infinite discrete values, it is considered discrete-valued. A time series of categorical data arises in different fields of science, especially environmental science, e.g., the air quality of a day is recorded as healthy, unhealthy, and hazardous. Similarly, for rainfall data, one may record a day as being dry, low rainfall, moderate rainfall, or heavy rainfall. Similarly, one can be more interested in whether a stock price is up or down over a certain period of time than in the actual value of a stock during the same time period. In all of the above scenarios, the data observed over a time period is categorical in nature. These examples are of the ordinal categories, where the categories are of an ordered nature. The other kind of categorical time series data is called nominal time series data, where the categories cannot be ordered (e.g., a genome structure like a, c, t, g). On the other hand, examples of count time series are monthly crime data for a place, monthly data for a particular disease reported at a hospital, etc. In this field of discrete-valued time series, Weiß, 2018 studied several statistical methods that can be utilized to evaluate datasets of this nature. The findings presented a valuable combination of theoretical frameworks and practical implementations.

Consider an example of daily air quality in Kolkata classified as Good (AQI:0-50), Moderate (AQI:51-100), Poor (AQI: 101-200), Unhealthy (AQI: 201-300), Very unhealthy (AQI: 301-400), and Hazardous (AQI: 401-500) based on the National Air Quality Index during, say, the last three months. In such a case, conventional models like autoregressive and moving average, or ARMA(p, q) process cannot be used as the basic arithmetic operations like addition, subtraction, multiplication, and division cannot be defined with categorical data. For example, Good $\pm$ Moderate does not make any sense. For the same reason, conditional expectation $E(Y_t \mid Y_{t-1})$ and marginal expectation $E(Y_t)$ cannot be defined in a straightforward way. Hence, the autocorrelation function (ACF) and the partial ACF cannot be defined as in the usual continuous-valued time series data. As a result, Box Jenkins' ARMA process (see Box and Jenkins, 1976), which involves addition and multiplication with categories, cannot be applied. In the following sections, we discuss some preliminaries about categorical and count time series.

1.2 Some preliminaries about categorical time series

Categorical time series were initially analyzed using Markov chains with stationary transition probabilities, which preserved the categorical nature of the data. One disadvantage is that higher order Markov chains require estimating a large

number of transition probabilities. Raftery, 1985 addressed this issue by introducing the mixture transition distribution (MTD) model. This model minimizes the number of parameters by expressing the transition probabilities as a linear combination of those from the first order Markov chain. Berchtold and Raftery, 2002 modified this model in their study. Pegram, 1980 investigated a specific type of stationary higher order Markov chains. This method further reduces the number of parameters by representing the transition probabilities as a linear combination of indicator kernels. This model resembles Box and Jenkins' ARMA processes (see Box and Jenkins, 1976) and allows for negative serial correlations. Jacobs and Lewis, 1978a; Jacobs and Lewis, 1978b; Jacobs and Lewis, 1978c; Jacobs and Lewis, 1983 proposed a method to obtain a stationary discrete series with a given marginal probability mass function and given autocorrelation structure. These models are known as discrete ARMA (DARMA) models.

Pegram's operator based autogressive process of order p (PAR(p)) was extended in Biswas and Song, 2009 to encompass more general discrete-valued ARMA processes. These processes are referred to as Pegram's ARMA (PARMA) process. In this study, they examined (i) a count data of the monthly number of claims of short-term disability benefits by the logging workers who had cut injury to the British Columbia Workers' Compensation Board during 1985-1994 (also see Freeland, 1998 and Zhu and Joe, 2006), (ii) another data of hardware failures of the Indian Statistical Institute, Kolkata for 260 working days in 2005, and (iii) an infant sleep status data (see Stoffer et al., 1988). Specifically, this PARMA model is comparable to the new DARMA (NDARMA) process, which was proposed by Jacobs and Lewis, 1983. On the other hand, a number of studies, including Weiß and Göb, 2008, Weiß, 2011b; Weiß, 2013, Biswas and Song, 2009, Biswas and Guha, 2009, and Biswas, Carmen Pardo, and Guha, 2014, were conducted on various measures of serial association for categorical time series. The purpose of these studies was to provide alternative measures for the likes of autocorrelation and partial autocorrelation.

Maiti and Biswas, 2015a studied different methods of forecasting and the forecasting accuracy depending on some forecasting measures in the context of categorical time series data. For illustration, they analyzed the infant sleep status data. Biswas and Song, 2009 and Heagerty and Zeger, 1998 studied the auto-odds ratio function (AORF) as a serial measure of dependence for categorical time series. Maiti and Biswas, 2018 extended the measure to a more general stationary Pegram's autoregressive process and moving average process. Also, there are studies in the field of categorical time series based on regression type models to tackle this kind of data. Some notable works in this regard are due to Kaufmann,

1987, Fahrmeir and Kaufmann, 1987, Fokianos and Kedem, 1998; Fokianos and Kedem, 2003. Weiß, 2018 discussed the analysis of categorical time series data mainly in Chapters 6 and 7, which is the second part of the book.

In Chapter 2, we consider the problem of modelling ordinal categorical time series data. As mentioned above, the Markov Chain model for modelling categorical time series suffers from the problem that it involves a large number of parameters when the order of a Markov chain and/or the number of categories of a time series are large. As an alternative, the MTD model and the PAR process are some useful models that require a relatively small number of parameters. However, they also have their own drawbacks. For example, the PAR process involves an indicator kernel that gives weight to the transition probability only when the same category occurs at previous time-points. Here, we propose a generalized PAR (GPAR) process with a generalized kernel function that gives weight to all the possible cases.

In the following section, we discuss major developments in the field of count time series analysis.

1.3 Some preliminaries about count time series

Over the last few decades, research on count time series data grew in popularity. As previously discussed, count time series data can be found in a variety of scientific disciplines, including social science, econometrics, and epidemiology. In a series of papers, Jacobs and Lewis, 1978a; Jacobs and Lewis, 1978b; Jacobs and Lewis, 1978c; Jacobs and Lewis, 1983 proposed a general class of simple models for discrete-valued time series, which are the discrete ARMA (DARMA) models. Another type of stationary process that has gained significant popularity in the field of count time series is integer-valued processes of order one (INAR(1) process). McKenzie, 1985; McKenzie, 1986 and Al-Osh and Alzaid, 1987 proposed the INAR(1) process based on the binomial thinning operator (see Steutel and Van Harn, 1979). The INAR(1) process with Poisson marginal, also known as the PINAR(1) process (see Al-Osh and Alzaid, 1987, Freeland and McCabe, 2004b; Freeland and McCabe, 2005) is widely used to model count time series data due to its simplicity. To detect positive shifts in the mean of the PINAR(1) process, Weiß, 2011a used an exponentially weighted moving average (EWMA) control chart. Some other popular studies in this regard are due to McKenzie, 2003, Davis, Dunsmuir, and Streett, 2003 and Davis, Dunsmuir, and Streett, 2005. However, the PINAR(1) process, due to its equal marginal mean and marginal variance, has a drawback to model overdispersed data, which can be found if (i)

some extreme observations are present in the data, (ii) a large number of zeros are present in the data, and (iii) both of these cases are present in the data. In the subsequent paragraph, we discuss the models that can handle these types of count time series.

McKenzie, 1986; McKenzie, 1988 proposed INAR(1) processes with geometric and negative binomial marginals for overdispersed count time series data. These models are discrete analogues of the exponential and gamma time series models of Gaver and Lewis, 1980. Al-Osh and Aly, 1992 proposed first order autoregressive time series with negative binomial and geometric marginals, which are discrete analogues of the gamma and exponential processes introduced by Sim, 1990. Zhu and Joe, 2006 investigated a stationary negative binomial time series using the binomial thinning operator. Ristić, Bakouch, and Nastić, 2009 proposed a new geometric INAR(1) (NGINAR(1)) model based on the negative binomial thinning operator. They analyzed monthly sex offence data reported in the 21st police car beat in Pittsburgh from 1990 to 2001. Ristić, Nastić, and Bakouch, 2012 investigated an INAR(1) process with negative binomial marginals (NBINAR(1)) using the negative binomial thinning operator, analyzing monthly burglary counts from 1991 to 2001. Schweer and Weiß, 2014 proposed a compound Poisson INAR(1) model to address overdispersion issues. Maiti, Biswas, and Chakraborty, 2018 developed a new geometric INAR(1) process to model overdispersed count time series data, particularly data with a high number of zeros and ones. Recently, Qian, Li, and Zhu, 2020 proposed a new INAR(1) process with generalized Poisson-inverse Gaussian innovations to solve overdispersion problems.

However, if the overdispersion is caused by a large number of zeros (zero-inflation), such as in data from a rare disease or infection, most of the aforementioned processes, particularly Poisson and negative binomial marginals, are inadequate to deal with the problem. Regression models based on the zero-inflated Poisson distribution are widely used for count data with zero-inflation in independent and identically distributed setups. Several studies, including Lambert, 1992, Böhning et al., 1999, and Angers and Biswas, 2003, demonstrate this. Wang, 2001 and Porter and White, 2012 developed models for handling zero-inflated count time series data sets. These models were applied to daily phone calls reporting faults for a mainframe computer system and daily terrorism attacks in Indonesia, respectively. Jazi, Jones, and Lai, 2012a proposed an INAR(1) model with zero-inflated Poisson innovations (ZINAR(1)), where they analyzed the data of monthly cases of submission of skin-lesions in New Zealand from 2003-2009. Jazi, Jones, and Lai, 2012b studied INAR(1) model with geometric innovations

(INARG(1)) for modelling overdispersed count data. Maiti et al., 2014 proposed a stationary zero-inflated Pegram's operator based integer-valued time series process of order p with Poisson marginal (ZIPPAR(p)) based on the PAR(p) model (see Pegram, 1980), which is very popular in the context of both count and categorical time series data. They examined the daily counts of terrorist attacks in India from 1994 to 2007. Maiti, Biswas, and Das, 2015 investigated a zero-inflated Poisson INAR(1) (ZIPINAR(1)) process to model overdispersion caused by zero-inflation. Weiß, 2018, in the first part of the book (mainly Chapters 2 to 5), examined different approaches for count time series data.

This thesis builds on the PINAR(1) model (see Al-Osh and Alzaid, 1987) and the ZINAR(1) model (see Jazi, Jones, and Lai, 2012a) to create models for count time series data with time-varying properties. Time-varying features can be found in datasets such as COVID-19 and other health data (for example, data on dengue cases in a specific location). In Chapter 4, we propose an INAR(1) model with time-varying Poisson innovations with change-points and apply it to some COVID-19 datasets. We use zero-inflated Poisson innovations with seasonality to create an INAR(1) process for count time series that has both zero-inflation and seasonality properties. The study can be found in Chapter 5. In Chapter 3, we present a modified PAR (mPAR) process for modelling truncated counts.

The following sections present brief overviews of the PAR process, the MTD model, and the INAR process. These processes serve as the foundation upon which the research works presented in this thesis are developed.

1.4 Brief reviews of the PAR, the MTD and the INAR processes

The PAR process

Let $\{Y_t\}$ be a Pegram's operator based AR(p) model such that

$$Y_t = (I(Y_{t-1}), \phi_1) * (I(Y_{t-2}), \phi_2) * \ldots * (I(Y_{t-p}), \phi_p) * (\varepsilon_t, 1 - \phi_1 - \phi_2 - ..... - \phi_p), \tag{1.4.1}$$

which is a mixture of $(p+1)$ discrete distributions, those of $Y_{t-1}, \ldots, Y_{t-p}$ and the innovation term ε_t, where $P(Y_t = i) = P(\varepsilon_t = i) = \pi_i$, $i = 0, 1, \ldots, k$, and it is denoted by $\varepsilon_t \sim D(\pi_0, \ldots, \pi_k)$, i.e., any discrete distribution. The respective

mixing weights are $\phi_1, \ldots, \phi_p$, where

$$\phi_i \in (0,1), \;\; i = 1, \ldots, p, \quad \text{and} \; \sum_{i=1}^{p} \phi_i \in (0,1).$$

For every $t = 0, \pm 1, \pm 2, \ldots$ the conditional probability function takes the form

$$P(Y_t = i | Y_{t-1}, \ldots, Y_{t-p}) = \sum_{j=1}^{p} \phi_j I(Y_{t-j} = i) + (1 - \sum_{j=1}^{p} \phi_j)\pi_i, \tag{1.4.2}$$

where ϕ_i's, $i = 1, \ldots, p$, are chosen such that the polynomial equation

$$1 - \phi_1 z - \ldots - \phi_p z^p = 0$$

has roots lying outside of the unit disc. Here, $I(\cdot)$ is the indicator function.

Biswas and Song, 2009, Biswas and Guha, 2009 and Song et al., 2013 studied various properties of the AR(p) model, such as its autocorrelation structure and also discusses how to estimate its parameters.

The MTD process

The MTD model with $(m+1)$ categories, introduced by Raftery, 1985, ignores the issue of having exponentially increasing number of free numbers in a Markov chain. This process specifies the conditional probability of Y_t given the past as a linear combination of contribution by $Y_{t-1}, Y_{t-2}, \ldots, Y_{t-p}$. Therefore, the MTD($p$) model can be written as

$$P\left(Y_t = i | Y_{t-1} = i_1, \ldots, Y_{t-p} = i_p\right) = \sum_{j=1}^{p} \lambda_j P\left(Y_t = i | Y_{t-j} = i_j\right) = \sum_{j=1}^{p} \lambda_j q_{i_j i}, \tag{1.4.3}$$

where $i, i_1, \ldots, i_p \in \{0, 1, \ldots, m\}$, $q_{i_j i}$'s are the elements of the transition probability matrix (say, Q) of the order $(m+1) \times (m+1)$, and $\lambda_1, \ldots, \lambda_p$ satisfy $\sum_{j=1}^{p} \lambda_j = 1$ ($\lambda_j \geq 0$ for all j), so that the right-hand side of (1.4.3) lies between 0 and 1.

The INAR process

The INAR(1) process based on binomial thinning operator, proposed by McKenzie, 1985 and Al-Osh and Alzaid, 1987, is defined as

$$Y_t = \alpha \circ Y_{t-1} + \varepsilon_t, \tag{1.4.4}$$

where the parameter $\alpha \in (0,1)$, and the innovations ε_t's are independent and identically distributed (i.i.d.) non-negative count random variables, independent of Y_{t-1}. And the binomial thinning operator "$\circ$", introduced by Steutel and Van Harn, 1979, is defined as

$$\alpha \circ Y_{t-1} = \sum_{i=1}^{Y_{t-1}} B_i,$$

where $B_1, B_2, \ldots$ are i.i.d. Bernoulli random variables with success probability α, i.e., $P(B_i = 1) = \alpha = 1 - P(B_i = 0)$. That is, given Y_{t-1}, $\alpha \circ Y_{t-1}$ follows a binomial distribution with mean αY_{t-1} and variance $\alpha(1-\alpha)Y_{t-1}$. The ACF of this INAR(1) process is given by $\rho(h) = \alpha^h$, irrespective of any specific distributional assumption of ε_t.

1.5 Forecasting for discrete-valued time series

The study of h-step ahead forecasting is critical in the context of analyzing time series data. In continuous time series analysis, widely used methods such as recursive predictions using the Durbin-Levinson algorithm and the innovations algorithm are used to obtain h-step ahead conditional mean forecast for the Box-Jenkins ARIMA models (see Brockwell and Davis, 2002). However, these methods cannot be used for discrete-valued time series, because they mainly execute mean forecasts instead of median and mode forecasts. This thesis mainly focuses on mean and mode predictors for count time series (see Freeland and McCabe, 2004a), as well as mode predictors for categorical time series. However, we also study the median predictor for the data analysis in Chapter 2.

The h-step ahead forecasting conditional mean has the desirable property of minimizing the expected predicted mean squared error. To convert the conditional forecasting mean value to an integer, we use the rounding-off operator to find the nearest integer. The mode predictor, on the other hand, is the point in the future time series support space where the density function reaches its maximum.

Depending on these predictors, we employ different forecasting measures for our studies. The details are provided in the respective chapters.

1.6 Organization of the thesis

In light of the aforementioned discussions, we divide the thesis into two main parts: (i) the first part is comprised of Chapter 2, which addresses the issue of categorical time series; and (ii) the second part consists of Chapters 3, 4, and 5,

which provide analyses of count time series data. In the following, we provide brief overviews of the chapters.

1.6.1 An overview of Chapter 2

In this chapter, we consider the problem of modelling categorical time series data with application to air quality data in India. The Markov Chain model is widely used to model categorical time series. However, it suffers from a large number of parameters when the order of a Markov chain and/or the number of categories of a time series are large. As an alternative, the mixture transition distribution (MTD) model, the multinomial logistic regression model, and Pegram's operator based autoregressive (PAR) process are some useful models that require a relatively small number of parameters. However, they also have their own limitations. For example, the PAR process of order p involves an indicator kernel that gives weight to the transition probability only when the same previous category occurs at time $t-1, \ldots, t-p$. Here, a new model, namely the generalized PAR (GPAR) process, is proposed using a generalized kernel function that gives weight (possibly different) to all the possible cases. The proposed model is mainly defined for ordinal categorical time series where categories are ordered in nature, like the air quality of a city observed as Healthy, Unhealthy, and Hazardous. We study the distributional properties and h-step ahead forecasting features of the proposed process, along with the estimation of model parameters. Extensive simulation studies are executed to investigate the utility of the proposed process. Finally, the method is illustrated through the analysis of a real dataset on the air quality of Mumbai city. The contents of this chapter are partially based on Chattopadhyay et al., 2024a.

1.6.2 An overview of Chapter 3

In this chapter, we consider the problem of such count time series, which consist of truncated counts. The PAR model is a well-known method in the field of time series of both counts and categories. However, the PAR model, while handling counts, also suffers from the drawback of having the indicator kernel function that only gives weight to situations in which the same event that occurred at a previous time-point occurs at the current time-point. To address this limitation of the PAR model for count time series data, we propose a modification to the PAR model for modelling truncated counts. The proposed modified PAR (mPAR) process involves a modified kernel in order to take care of the drawback of the PAR model by giving some weight to scenarios in which an event that differs from the

one that occurred at the previous time-point can also occur at the current time-point. The mPAR process is developed to model truncated counts, particularly by using right-truncated Poisson innovations. We examine the distributional properties and the h-step ahead forecasting of our proposed process. To study the usefulness of our proposed process, a detailed comparative study between the mPAR(1) process and the PAR(1) process is executed using some simulation experiments. In the simulation framework, we also investigate the empirical consistency of the estimated model parameters of our proposed process. Finally, for practical illustration, a real dataset is analyzed. The contents of this chapter are partially based on Chattopadhyay et al., 2024b.

1.6.3 An overview of Chapter 4

In this chapter, we look at the problem of change-point analysis for count time series data using an INAR(1) process with time-varying covariates. These types of features can be found in many real-life scenarios, particularly in COVID-19 datasets, where the number of both new and active cases decreases over time before increasing again. To capture those features, we use the Poisson INAR(1) (PINAR(1)) process with a time-varying smoothing covariate. Using this model, we can model both components of active cases at time-point t: (i) the number of non-recovery cases from the previous time-point, and (ii) the number of new cases at time-point t. In addition to forecasting, we look at some theoretical properties of the proposed model. Some simulation studies are carried out to assess the effectiveness of the proposed method. Finally, we examine two COVID-19 datasets and compare our proposed model to another PINAR(1) process with time-varying covariates but no change-point to demonstrate the overall performance of our proposed model. The contents of this chapter are partially based on Chattopadhyay et al., 2021.

1.6.4 An overview of Chapter 5

In this chapter, we study a count time series of weekly dengue cases in Kaohsiung City, Taiwan, during the period 2009–2012, where the problems of a large number of zeros, or zero-inflation and seasonality arise together. To capture both features, we develop an INAR(1) process using zero-inflated Poisson innovations with seasonality. We also incorporate an exogenous variable, namely the weekly maximum temperature, in the innovations to study the effect of temperature on the number of dengue cases, as many studies have found that temperature is a significant factor in spreading dengue infections globally. The proposed model

can capture both the non-recovery cases from the previous time point and the new cases coming at the current time point. The distributional and forecasting properties of the proposed model are derived in this study. Simulation experiments are performed to study the effectiveness of the proposed model. The consistency of the estimated parameters is empirically studied using some simulation studies. The proposed model is compared with some existing INAR models. Finally, we analyze the dataset on weekly dengue cases in Kaohsiung for practical illustration. The contents of this chapter are partially based on Chattopadhyay et al., 2023.

1.6.5 An overview of Chapter 6

The final chapter, i.e., Chapter 6, provides our concluding remarks concerning the research works that are included in the thesis.

Chapter 2

A generalized Pegram's operator based autoregressive process for modelling categorical time series

2.1 Introduction

This chapter focuses on analyzing time series data that is made up of ordinal categories. Chapter 1 explains how categorical time series data can be found in different practical situations, such as environmental science. For example, air quality levels can be categorized as Good, Moderate, Poor, Unhealthy, Very unhealthy, and Hazardous. When dealing with this kind of data, traditional models such as ARMA(p, q) process are not applicable. Weiß, 2018 studied categorical time series in detail mainly in the second part of the book (Chapters 6 and 7).

Most conventional approach to modelling categorical time series is Markov Chain of order, say p, where one directly estimates the transition probabilities $P(Y_t = C_i \mid Y_{t-1}, \ldots, Y_{t-p})$, $i = 0, 1, 2, \ldots, m$, where $C_0, C_1, \ldots, C_m$ are the categories under consideration. However, for a Markov chain of order p with $(m+1)$ categories, the number of parameters to be estimated is $m(m+1)^p$. For example, if $m = 4$ (five categories) and $p = 2$ (second order model), the number of parameters to be estimated is 100, which is quite large.

An alternative approach that substantially reduces the number of parameters of the Markov chain model is called mixture of transition distribution (MTD) model (See Raftery, 1985 and Berchtold and Raftery, 2002). An MTD model of order p can be defined by the following transition probability

$$P(Y_t \mid Y_{t-1}, \ldots, Y_{t-p}) = \sum_{j=1}^{p} \lambda_j P(Y_t \mid Y_{t-j}), \tag{2.1.1}$$

where $\lambda_j \geq 0$ and $\sum_{j=1}^{p} \lambda_j = 1$. For an MTD model of order p with $(m+1)$ states, the total number of parameters to be estimated is $((m+1)m + p - 1)$. If $m = 4$ and $p = 2$, then the number of parameters to be estimated is 21 as compared to 100 in the full Markov Chain model. Although it seems small as compared to a full Markov Chain model, for a second order process with four categories, the number of parameters (21) to be estimated is still not so small and one needs large enough data to estimate all these parameters efficiently. On the other hand, Pegram, 1980 proposed a new Markov model which is a special case of discrete ARMA (DARMA) process proposed by Jacobs and Lewis, 1978a; Jacobs and Lewis, 1978b; Jacobs and Lewis, 1978c. A Pegram's AR process of order p or PAR(p) process is defined by

$$Y_t = (I(Y_{t-1}), \phi_1) * (I(Y_{t-2}), \phi_2) * \ldots * (I(Y_{t-p}), \phi_p) * (\varepsilon_t, 1 - \phi_1 - \phi_2 - ..... - \phi_p), \tag{2.1.2}$$

which is a mixture of $(p+1)$ discrete distributions, those of $Y_{t-1}, \ldots, Y_{t-p}$ and the innovation term ε_t, where $P(Y_t = C_i) = P(\varepsilon_t = C_i) = \pi_i$, $i = 0, 1, \ldots, m$. We denote by $\varepsilon_t \sim D(\pi_0, \ldots, \pi_m)$, where "$D$" stands for any discrete distribution. The respective mixing weights are $\phi_1, \ldots, \phi_p$, where

$$\phi_i \in (0,1), \quad i = 1, \ldots, p, \quad \text{and} \quad \sum_{i=1}^{p} \phi_i \in (0,1).$$

Therefore, the transition probability is defined as

$$P(Y_t = C_i \mid Y_{t-1}, \ldots, Y_{t-p}) = \phi_1 I(Y_{t-1} = C_i) + \ldots + \phi_p I(Y_{t-p} = C_i) + (1 - \phi_1 - \ldots - \phi_p)\pi_i. \tag{2.1.3}$$

Total number of parameters in the full model is $(m + p)$. Thus, if $m = 4$ (total 5 categories) and $p = 2$, then number of parameters to be estimated is 6, as compared to 100 in the full Markov model. The following table provides a better understanding of how many parameters are needed for the PAR, compared to MTD and Markov processes, with order p and $(m+1)$ categories.

TABLE 2.1: Number of parameters for the PAR, MTD and Markov models

$(p, m+1)$	PAR	MTD	Markov model
(1,3)	3	6	6
(1,4)	4	12	12
(1,5)	5	20	20
(2,3)	4	7	18
(2,4)	5	13	48
(2,5)	6	21	100

The main limitation of the PAR(p) process is that it uses the indicator function in its transition probabilities. As a result the model becomes restrictive in nature. The problem with the indicator kernel associated with the PAR(p) process is that for Y_t to take category C_i it gives weight to the case $C_i = C_j$ but no weight to $C_i \neq C_j$ in the transition probability. In fact, it has been observed that the PAR(p) process is especially suitable when a time series has long runs in some categories.

In this chapter, we propose a generalized PAR (GPAR) process that needs just one extra parameter than the usual PAR process. The GPAR process employs a generalized kernel that gives weight to a different category from the previous time-point for appearing at the current time-point. This process is mainly defined for ordinal categorical time series, e.g., air quality of a city classified as Good, Moderate, Poor, Unhealthy, Very unhealthy, and Hazardous, measured in an ordinal scale. We study the distributional properties and h-step ahead forecasting features of the proposed process along with the estimation of model parameters. Extensive simulation experiments are carried out to investigate the applicability of the proposed process. Finally, the method is illustrated using a real dataset on air quality of Mumbai city. This chapter is partially based on Chattopadhyay et al., 2024a.

The rest of this chapter is organized as follows. In Section 2.2, we define the GPAR process of order p and we discuss the method to select the order of the GPAR(p) process using auto-odds ratio function. In section 2.3, we study different distributional properties of the GPAR(1) process. The maximum likelihood estimation method is discussed in Section 2.4. Extensive simulation experiments are carried out in Section 2.5 to examine the usefulness of the proposed GPAR(1) process. The method is illustrated through a real dataset on air quality of Mumbai city in Section 2.6. Finally, some conclusions are drawn in Section 2.7. All the proofs of the theoretical results are provided in Appendix.

2.2 The GPAR(p) process

Note that $\{Y_t\}$ is said to follow a PAR(p) process if its transition probability is written as in equation (2.1.3). As it is noted earlier, the problem with the indicator kernel associated with the PAR(p) process is that it gives weight to transition probability, when $C_i = C_j$ but no weight when $C_i \neq C_j$. Also it does not consider the lag distance between two time-points. To overcome this limitation, we assign some weight to the cases $C_i \neq C_j$, which is given by

$$\begin{aligned} P(Y_t = C_i \mid Y_{t-1} = C_{i_1}, \ldots, Y_{t-p} = C_{i_p}) &= \phi_1 K_1(i_1 \mid i) + \ldots + \phi_p K_p(i_p \mid i) \\ &\quad + (1 - \phi_1 - \phi_2 - .... - \phi_p)\pi_i, \qquad (2.2.1) \\ &\quad \text{where, } i, i_1, \ldots, i_p = 0, 1, \ldots, m. \end{aligned}$$

Here, $K_h(j \mid i) \geq 0$ such that $\sum_{i=0}^{m} K_h(j \mid i) = 1$ for all $j = 0, 1, \ldots, m$ and $h = 1, 2, \ldots, p$ and $\pi_i \geq 0$ for all i and $\sum_{i=0}^{m} \pi_i = 1$. Note that, $p_i = P(Y_t = C_i)$ and $\pi_i = P(\varepsilon_t = C_i)$, and $p_i \neq \pi_i$ for our proposed process, but they are related, which we establish later. We call this process as the generalized Pegram's operator based AR(p) process or GPAR(p) process. Our proposed choice of K_h is

$$K_h(j \mid i) = \theta_{ij}(h) = \frac{\theta^{|i-j|}}{hM_j(h)}, \quad h = 1, 2, \ldots, p; \; i = 0, 1, \ldots, m; \text{ and } \; j = 0, 1, \ldots, m, \qquad (2.2.2)$$

where $M_j(h)$, $j = 0(1)m$, are the normalizing constants, and $\theta \in (0, 1)$. This particular kernel is useful when the time series is ordinal categorical and the categories which are closer are more dependent than the categories far apart. For example, in case of air quality data where air quality is recorded as Good, Moderate, Poor, Unhealthy, Very unhealthy, and Hazardous based on the National Air Quality Index, we often see the days where good air quality is followed by moderate or vice versa, and days with poor air quality followed by unhealthy or vice versa. In such scenarios, it is better to use the above kernel while computing the transition probabilities than the indicator kernel used in the PAR(p) process. Using the above kernel function we can define the following $(m+1) \times (m+1)$

kernel matrix K_h for all $h = 1, 2, \ldots, p$:

$$K_h = \begin{pmatrix} \frac{1}{hM_0(h)} & \frac{\theta}{hM_0(h)} & \cdots & \frac{\theta^m}{hM_0(h)} \\ \frac{\theta}{hM_1(h)} & \frac{1}{hM_1(h)} & \cdots & \frac{\theta^{m-1}}{hM_1(h)} \\ \vdots & & & \\ \frac{\theta^m}{hM_m(h)} & \frac{\theta^{m-1}}{hM_m(h)} & \cdots & \frac{1}{hM_m(h)} \end{pmatrix}$$

such that each row sum is equal to one. Unless defined otherwise 'T' and '$\prime$', refer to transposes of a matrix and a vector, respectively. As one can see this type of kernel is mainly developed for ordinal categorical time series. For simplicity, we assume $K_h = K$ for all h. Then, the above kernel matrix becomes

$$K = \begin{pmatrix} \frac{1}{M_0} & \frac{\theta}{M_0} & \cdots & \frac{\theta^m}{M_0} \\ \frac{\theta}{M_1} & \frac{1}{M_1} & \cdots & \frac{\theta^{m-1}}{M_1} \\ \vdots & & & \\ \frac{\theta^m}{M_m} & \frac{\theta^{m-1}}{M_m} & \cdots & \frac{1}{M_m} \end{pmatrix}$$

where M_j's are such that each row sum is equal to one.

Result 2.1 *Under the assumption that $K_h = K$ for all h, the normalizing constants M_j are given by $M_j = \dfrac{1 - \theta^{j+1} + \theta - \theta^{m-j+1}}{1 - \theta}$, where $j = 0, 1, \ldots, m$.*

The proof of this result is provided in Appendix 2.8.

Note that for the PAR(p) process, the transition probability matrix $K_h = K$ for all $h = 1, 2, \ldots, p$, and is given by

$$\mathbf{K} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & & \\ 0 & 0 & \cdots & 1 \end{pmatrix}.$$

Note 2.1 *Weiß and Swidan, 2024 proposed a new extension of the NDARMA model (see Jacobs and Lewis, 1983) by using some weighting operators which are (i) identity weighting, (ii) reverse weighting, (iii) triangular weighting, and (iv) zero-inflation. These models (WDARMA) can be applied to model nominal time series with negative serial dependencies or to ordinal time series where transitions to neighboring states are more likely*

than sudden large jumps. Therefore, even if there appears to be a close relationship between the WDARMA model and our proposed GPAR model, there are differences in the ideas of developing the weights as we consider a generalized kernel with geometric weights with one more parameter than the PAR model. Moreover, Weiß and Swidan, 2024 incorporated the weights into the NDARMA model to develop the WDARMA model, whereas we incorporate weights into the PAR model to develop the GPAR model. Additionally, the h-step ahead forecasting property of the proposed GPAR(1) model is studied throughout this chapter.

Order selection

Maiti and Biswas, 2018 examined the auto-odds ratio function (AORF) measure (i.e., $r(h)$: the AORF between Y_t and Y_{t-h}) to determine the order of PAR processes. Biswas and Song, 2009 studied this measure for PAR(1) process with binary responses. Maiti and Biswas, 2018, in their article, extended the AORF measure for a more general setup. Their study displayed how the AORF helps to measure the order of PAR processes. They showed that a decreasing pattern of $\log(r(h))$ for different h suggests that an AR of order 1 process would possibly be a good fit for the data. However, in order to select the order more precisely, Weiß and Göb, 2008, Weiß, 2011b; Weiß, 2013 suggested observing the usual PACF, $\rho_p(h)$ based on the estimates $\hat{\kappa}(h)$ for Cohen's κ instead of the $\rho(h)$. This helps us to detect the order of our proposed GPAR process in the data application.

2.3 The GPAR(1) process

In particular, the GPAR(1) process is defined by the following transition probability distribution

$$P(Y_t = C_i \mid Y_{t-1} = C_j) = \phi K(j \mid i) + (1-\phi)\pi_i. \tag{2.3.1}$$

Result 2.2 *Taking the assumption, $P(Y_{t-h} = i) = p_i$ for all h, into consideration, the marginal probability distribution of the GPAR(1) process, defined in equation (2.3.1), is given by*

$$\underset{\sim}{p} = (1-\phi)(I - \phi K^T)^{-1}\underset{\sim}{\pi},$$

where I is an identity matrix with order $(m+1)$, $\underset{\sim}{p} = [p_0, \ldots, p_m]'$, and $\underset{\sim}{\pi} = [\pi_0, \ldots, \pi_m]'$.

The Appendix (Section 2.8) shows the derivation of this result along with the proof of the existence of $\left(I - \phi K^T\right)^{-1}$.

Result 2.3 $\sum_{j=0}^{m} p_j = \underset{\sim}{1}' \underset{\sim}{p} = (1-\phi)\underset{\sim}{1}'(1-\phi K^T)^{-1}\underset{\sim}{\pi} = 1$, *where* $\underset{\sim}{1} = [1,1,\ldots,1]'$.

The proof of this result is shown in the Appendix section (Section 2.8).

Result 2.4 *The h-step ahead forecasting distribution of our proposed GPAR(1) process, i.e., the probability mass function (p.m.f.) of* Y_{t+h} *given* Y_t *is given by*

$$\begin{aligned} p_h(i \mid j) &= P(Y_{t+h} = C_i \mid Y_t = C_j) \\ &= [\phi K_1(j,i) + (1-\phi)\pi_i]\, I(h=1) + \\ &\quad \left[\phi^h K_h(j,i) + (1-\phi)\sum_{n=1}^{h-1} \phi^{h-n}\underset{\sim}{\pi}'\underset{\sim}{K}^{h-n}(i) + (1-\phi)\pi_i\right] I(h \geq 2), \end{aligned} \tag{2.3.2}$$

where $\underset{\sim}{K}^h(i)$ *is the* $(i+1)$*th column of* K^h *and* $K_h(j,i)$ *is the* $(j+1, i+1)$*th element of* K^h.

The proof of this result is shown in the Appendix (Section 2.8).

Note 2.2 *For* $h \geq 1$, *the h-step ahead forecasting distribution of the PAR(1) process is given by*

$$P(Y_{t+h} = C_i \mid Y_t = C_j) = \phi^h I(j=i) + (1-\phi^h)p_i.$$

Note 2.3 *The one-step ahead forecasting distribution of the MTD(1) process is given by*

$$p_1(i) = P(Y_{n+1} = i \mid Y_n = i_1) = q_{i_1 i},$$

which is the $(i_1+1, i+1)$*th element of the transition probability matrix* Q.
The two-step ahead forecasting distribution of the MTD(1) process is given by

$$\begin{aligned} p_2(i) &= P(Y_{n+2} = i \mid Y_n = i_1) \\ &= \sum_{j_0=0}^{m} P(Y_{n+2} = i \mid Y_{n+1} = j_0) \times P(Y_{n+1} = j_0 \mid Y_n = i_1) \\ &= \sum_{j_0=0}^{m} q_{j_0 i} q_{i_1 j_0}, \end{aligned}$$

which is the $(i_1+1, i+1)$*th element of* Q^2.
Similarly, we can find the h-step ahead by extending it to $(i_1+1, i+1)$*-th element of* Q^h.

Note 2.4 *In this study, we consider mode predictors for both simulation study and data analysis. Additionally, we also investigate the median predictor through our data analysis*

(see Section 2.6). The median predictor, which is integer-valued like the mode predictor, has the optimal property of minimizing the expected absolute deviation.

2.4 Parameter Estimation

Maximum likelihood estimation

In the maximum likelihood estimation method, we maximize the log likelihood function with respect to the model parameters $\boldsymbol{\beta}$ such that $(I - \phi K^T)\underset{\sim}{\hat{p}} \geq 0$, to obtain the maximum likelihood estimate of $\boldsymbol{\beta}$, where $\boldsymbol{\beta} = (\phi, \theta)$. Given a dataset $\{Y_1, \ldots, Y_n\}$, the likelihood function can be written as

$$L(\boldsymbol{\beta}) = p(Y_1, \ldots, Y_n) = p(Y_1) \prod_{t=2}^{n} p(Y_t \mid Y_{t-1}). \tag{2.4.1}$$

Here, $P(Y_t = i \mid Y_{t-1})$ is equal to $\phi K(Y_{t-1} \mid i) + (1 - \phi)\hat{\pi}_i$ for the GPAR process, where $(1 - \phi)^{-1}(I - \phi K^T)\underset{\sim}{\hat{p}} = \underset{\sim}{\hat{\pi}}$, and $\phi I(Y_{t-1} = i) + (1 - \phi)\hat{\pi}_i$ for the PAR process, where $\underset{\sim}{\hat{p}} = \underset{\sim}{\hat{\pi}}$.
We carry out the optimization by taking log of the likelihood function ($l(\boldsymbol{\beta}) = \ln L(\boldsymbol{\beta})$) to get the maximum likelihood estimate $\hat{\boldsymbol{\beta}}_{mle}$, which is given by

$$\hat{\boldsymbol{\beta}}_{mle} = \arg\max_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) \text{ subject to } (I - \phi K^T)\underset{\sim}{\hat{p}} \geq 0. \tag{2.4.2}$$

This is a constrained non-linear optimization problem for the GPAR(1) process. Here, the numerical methods are being employed to obtain the ML estimates of the model parameters as there are no closed forms of the ML estimates. We use "solnp()" and "solnl()" functions from "Rsolnp" and "NlcOptim" packages in R software, respectively, for this purpose. For some cases in the simulation study, we obtain "NaNs" (not a number) in R, which we ignore by using the "is.finite()" command.

In the next section, we perform extensive simulation studies to show the empirical consistency of the estimates of the model parameters through the MLE method.

2.5 Simulation Study

2.5.1 Empirical consistency and AIC comparison

In this section, we investigate the empirical consistency of the estimated model parameters of our proposed GPAR(1) process where the parameters are estimated by the MLE method. We carry out the simulation experiments by generating samples from the GPAR(1) process. Here, we consider for $(\phi, \theta) = (0.3, 0.1)$, $(0.3, 0.2)$, $(0.3, 0.3)$, $(0.3, 0.4)$, $(0.5, 0.1)$, $(0.5, 0.2)$, $(0.5, 0.3)$ and $(0.5, 0.4)$, with three, four and five categories, where $\underset{\sim}{\pi} = (0.2, 0.3, 0.5)'$, $(0.1, 0.2, 0.3, 0.4)'$ and $(0.1, 0.2, 0.4, 0.2, 0.1)'$, respectively. Five sample sizes, 200, 500, 1000, 5000 and 10000, are explored here, each with 1000 replications. For the small-sample properties, the experiments are done for sample size 200 whereas sample sizes of 500 and 1000 are used for moderate large-sample properties. Sample sizes of 5000 and 10000 are used for checking the large-sample properties.

The simulation results are given in Tables 2.2 to 2.7. From the simulation results, we can observe that as sample size increases, the MSEs of the estimated model parameters of the proposed GPAR(1) process, i.e., $(\hat{\phi}, \hat{\theta})$, decrease. This property confirms the empirical consistency of the estimated model parameters of the proposed GPAR(1) process.

We also perform a model comparison between the GPAR(1), the PAR(1) and the MTD(1) processes through the Akaike Information Criteria (AIC). The simulation results for average AICs are given in Tables 2.8, 2.9 and 2.10 for $(\phi, \theta) = (0.5, 0.1)$, $(0.5, 0.2)$, $(0.5, 0.3)$ and $(0.5, 0.4)$, with three, four and five categories for which $\underset{\sim}{\pi} = (0.2, 0.3, 0.5)'$, $(0.1, 0.2, 0.3, 0.4)'$ and $(0.1, 0.2, 0.4, 0.2, 0.1)'$, respectively. Here also, we conduct this experiment for sample sizes 200, 500, 1000, 5000 and 10000 with 1000 replications.

For small size 200 with three categories, we can see the differences between average AICs for our proposed GPAR(1) process and the PAR(1) process are marginal. However, as the sample size increases in that setup, it can be observed that the differences between average AICs of our proposed GPAR(1) and the PAR(1) process increase. For the studies regarding four and five categories, the proposed GPAR(1) process performs better than the PAR(1) process for all sample sizes. We can also see that as the value of θ changes from 0.1 to 0.2 or 0.2 to 0.3, the differences between the average AICs of the GPAR(1) process and the PAR(1) process get bigger for every setup. This is expected as we go towards more generalization from the PAR(1) process to the GPAR(1) process. When we

compare with the MTD(1) process, we observe that our proposed GPAR(1) process performs better than the MTD(1) process in this simulation study. However, we can notice that as sample sizes increase, the MTD(1) process outperforms the PAR(1) process and becomes the closest competitor to our proposed GPAR(1) process. This property can mostly be seen from the numerical results for sample sizes 1000, 5000, 10000 with three categories, whereas we can see this happening mostly for 5000 and 10000 sample sizes with four and five categories. The GPAR(1) and PAR(1) processes performing better than the MTD(1) process in most cases with the increase in number of categories is expected since the number of parameters to be estimated for the MTD(1) process increases with respect to the GPAR(1) and the PAR(1) process with the increase in number of categories (see Table 2.1). However, the aforementioned observation that the MTD(1) process outperforms the PAR(1) process for large sample sizes hints at the fact that the MTD(1) process is more generalized than the PAR(1) process for small number of categories. Overall, our proposed GPAR(1) process performs better than the PAR(1) and the MTD(1) processes in terms of the AIC.

TABLE 2.2: Average estimated values of parameters along with the MSEs with true $(\phi, \theta) = (0.3,0.1), (0.3,0.2), (0.3,0.3)$, and $(0.3,0.4)$ for three categories, where $\underset{\sim}{\pi} = (0.2, 0.3, 0.5)'$

(ϕ, θ)	(0.3,0.1)		(0.3,0.2)		(0.3,0.3)		(0.3,0.4)	
n	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)
200	0.3762	0.1709	0.4005	0.2631	0.4262	0.3451	0.4148	0.4037
	(0.0390)	(0.0398)	(0.0604)	(0.0617)	(0.0817)	(0.0827)	(0.0883)	(0.0980)
500	0.3406	0.1396	0.3701	0.2443	0.3927	0.3288	0.4002	0.3868
	(0.0171)	(0.0207)	(0.0383)	(0.0392)	(0.0602)	(0.0579)	(0.0735)	(0.0729)
1000	0.3174	0.1207	0.3314	0.2162	0.3666	0.3249	0.3845	0.4001
	(0.0056)	(0.0098)	(0.0187)	(0.0232)	(0.0409)	(0.0413)	(0.0586)	(0.0566)
5000	0.3042	0.1041	0.3078	0.2052	0.3183	0.3067	0.3439	0.4136
	(0.0009)	(0.0021)	(0.0025)	(0.0051)	(0.0090)	(0.0121)	(0.0232)	(0.0214)
10000	0.3025	0.1028	0.3025	0.2023	0.3075	0.3043	0.3236	0.4076
	(0.0005)	(0.0011)	(0.0010)	(0.0023)	(0.0031)	(0.0054)	(0.0109)	(0.0119)

TABLE 2.3: Average estimated values of parameters along with the MSEs with true $(\phi, \theta) = (0.5,0.1), (0.5,0.2), (0.5,0.3)$, and $(0.5,0.4)$ for three categories, where $\underset{\sim}{\pi} = (0.2, 0.3, 0.5)'$

(ϕ, θ)	(0.5,0.1)		(0.5,0.2)		(0.5,0.3)		(0.5,0.4)	
n	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)
200	0.5239	0.1185	0.5318	0.2127	0.5444	0.3032	0.5350	0.3642
	(0.0184)	(0.0119)	(0.0331)	(0.0241)	(0.0460)	(0.0339)	(0.0637)	(0.0536)
500	0.5138	0.1106	0.5280	0.2117	0.5346	0.3036	0.5543	0.3923
	(0.0085)	(0.0064)	(0.0201)	(0.0139)	(0.0339)	(0.0229)	(0.0533)	(0.0354)
1000	0.5087	0.1074	0.5222	0.2118	0.5337	0.3085	0.5445	0.3962
	(0.0044)	(0.0035)	(0.0118)	(0.0083)	(0.0232)	(0.0146)	(0.0383)	(0.0241)
5000	0.5013	0.1012	0.5021	0.2007	0.5115	0.3048	0.5225	0.4059
	(0.0007)	(0.0006)	(0.0019)	(0.0016)	(0.0052)	(0.0038)	(0.0142)	(0.0076)
10000	0.5011	0.1010	0.5022	0.2014	0.5046	0.3020	0.5166	0.4060
	(0.0004)	(0.0003)	(0.0010)	(0.0008)	(0.0023)	(0.0018)	(0.0074)	(0.0043)

TABLE 2.4: Average estimated values of parameters along with the MSEs with true $(\phi, \theta) = (0.3,0.1), (0.3,0.2), (0.3,0.3)$, and $(0.3,0.4)$ for four categories, where $\underset{\sim}{\pi} = (0.1, 0.2, 0.3, 0.4)'$

(ϕ, θ)	(0.3,0.1)		(0.3,0.2)		(0.3,0.3)		(0.3,0.4)	
n	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)
200	0.3258	0.1332	0.3408	0.2300	0.3644	0.3249	0.3729	0.4102
	(0.0133)	(0.0190)	(0.0253)	(0.0348)	(0.0397)	(0.0532)	(0.0481)	(0.0672)
500	0.3076	0.1102	0.3159	0.2087	0.3453	0.3225	0.3582	0.4027
	(0.0047)	(0.0079)	(0.0104)	(0.0167)	(0.0225)	(0.0303)	(0.0345)	(0.0416)
1000	0.3042	0.1041	0.3116	0.2088	0.3266	0.3164	0.3450	0.4124
	(0.0020)	(0.0037)	(0.0049)	(0.0089)	(0.0126)	(0.0187)	(0.0229)	(0.0281)
5000	0.3004	0.1009	0.3028	0.2030	0.3022	0.2985	0.3146	0.4071
	(0.0004)	(0.0009)	(0.0008)	(0.0018)	(0.0016)	(0.0033)	(0.0047)	(0.0070)
10000	0.3001	0.0998	0.2999	0.1993	0.3008	0.2997	0.3043	0.4006
	(0.0002)	(0.0005)	(0.0004)	(0.0009)	(0.0008)	(0.0017)	(0.0020)	(0.0036)

TABLE 2.5: Average estimated values of parameters along with the MSEs with true $(\phi, \theta) = (0.5,0.1), (0.5,0.2), (0.5,0.3)$, and $(0.5,0.4)$ for four categories, where $\underset{\sim}{\pi} = (0.1, 0.2, 0.3, 0.4)'$

(ϕ, θ)	(0.5,0.1)		(0.5,0.2)		(0.5,0.3)		(0.5,0.4)	
n	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)
200	0.5020	0.1077	0.5037	0.2048	0.5185	0.3019	0.5022	0.3719
	(0.0098)	(0.0065)	(0.0154)	(0.0125)	(0.0250)	(0.0212)	(0.0354)	(0.0328)
500	0.5002	0.1023	0.5095	0.2073	0.5074	0.2985	0.5241	0.4008
	(0.0036)	(0.0026)	(0.0078)	(0.0062)	(0.0142)	(0.0107)	(0.0240)	(0.0172)
1000	0.5003	0.1010	0.5031	0.2010	0.5085	0.3014	0.5177	0.4000
	(0.0018)	(0.0014)	(0.0034)	(0.0028)	(0.0078)	(0.0059)	(0.0156)	(0.0104)
5000	0.4992	0.0993	0.5007	0.2001	0.5022	0.3012	0.5089	0.4031
	(0.0004)	(0.0003)	(0.0006)	(0.0005)	(0.0013)	(0.0011)	(0.0038)	(0.0025)
10000	0.4993	0.0996	0.5009	0.2008	0.5013	0.3003	0.5048	0.4023
	(0.0002)	(0.0001)	(0.0003)	(0.0003)	(0.0007)	(0.0006)	(0.0017)	(0.0012)

TABLE 2.6: Average estimated values of parameters along with the MSEs with true $(\phi, \theta) = (0.3,0.1), (0.3,0.2), (0.3,0.3)$, and $(0.3,0.4)$ for five categories, where $\underset{\sim}{\pi} = (0.1, 0.2, 0.4, 0.2, 0.1)'$

(ϕ, θ)	(0.3,0.1)		(0.3,0.2)		(0.3,0.3)		(0.3,0.4)	
n	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)
200	0.3103	0.1237	0.3223	0.2218	0.3445	0.3246	0.3587	0.4035
	(0.0083)	(0.0135)	(0.0153)	(0.0252)	(0.0266)	(0.0393)	(0.0386)	(0.0574)
500	0.3031	0.1032	0.3036	0.2010	0.3223	0.3141	0.3382	0.4052
	(0.0029)	(0.0048)	(0.0054)	(0.0108)	(0.0119)	(0.0198)	(0.0239)	(0.0319)
1000	0.3018	0.1004	0.3059	0.2062	0.3121	0.3113	0.3192	0.4018
	(0.0014)	(0.0026)	(0.0026)	(0.0056)	(0.0053)	(0.0100)	(0.0120)	(0.0179)
5000	0.3000	0.1004	0.3007	0.2006	0.3006	0.2993	0.3065	0.4061
	(0.0003)	(0.0005)	(0.0005)	(0.0009)	(0.0008)	(0.0017)	(0.0020)	(0.0038)
10000	0.3004	0.1003	0.3006	0.2004	0.3008	0.3009	0.3024	0.4017
	(0.0002)	(0.0003)	(0.0002)	(0.0005)	(0.0004)	(0.0008)	(0.0009)	(0.0017)

TABLE 2.7: Average estimated values of parameters along with the MSEs with true $(\phi, \theta) = (0.5,0.1)$, $(0.5,0.2)$, $(0.5,0.3)$, and $(0.5,0.4)$ for five categories, where $\underset{\sim}{\pi} = (0.1, 0.2, 0.4, 0.2, 0.1)'$

(ϕ, θ)	(0.5,0.1)		(0.5,0.2)		(0.5,0.3)		(0.5,0.4)	
n	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\theta}$ (MSE)
200	0.4936	0.1019	0.4903	0.1971	0.5039	0.3027	0.4993	0.3834
	(0.0061)	(0.0044)	(0.0092)	(0.0086)	(0.0174)	(0.0155)	(0.0251)	(0.0221)
500	0.4969	0.1001	0.4985	0.2006	0.5031	0.3039	0.5061	0.3977
	(0.0026)	(0.0017)	(0.0041)	(0.0035)	(0.0073)	(0.0064)	(0.0138)	(0.0108)
1000	0.4982	0.0985	0.4988	0.1988	0.5024	0.3016	0.5118	0.4038
	(0.0013)	(0.0008)	(0.0020)	(0.0017)	(0.0039)	(0.0033)	(0.0085)	(0.0058)
5000	0.4991	0.0999	0.4997	0.2003	0.5003	0.3001	0.5014	0.3998
	(0.0003)	(0.0002)	(0.0004)	(0.0003)	(0.0007)	(0.0006)	(0.0015)	(0.0012)
10000	0.4994	0.0997	0.4999	0.1998	0.5001	0.3003	0.5006	0.4001
	(0.0001)	(0.0001)	(0.0002)	(0.0002)	(0.0003)	(0.0003)	(0.0008)	(0.0006)

TABLE 2.8: Comparative study between GPAR(1), PAR(1) and MTD(1) methods with respect to AIC, where the data generating process is the proposed GPAR(1) process with true $(\phi, \theta) =$ (0.5,0.1), (0.5,0.2), (0.5,0.3), and (0.5,0.4) for three categories, where $\underset{\sim}{\pi} = (0.2, 0.3, 0.5)'$

(ϕ, θ)	(0.5,0.1)			(0.5,0.2)		
n	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
200	361.6664	361.2340	365.9934	385.2090	385.5268	389.5607
500	895.9916	896.9607	900.3848	954.2602	957.2376	959.1066
1000	1789.7559	1792.9833	1794.4062	1902.9726	1910.3761	1908.8767
5000	8926.3077	8945.5393	8935.7232	9505.3844	9544.5322	9518.2852
10000	17842.2033	17881.9145	17857.2428	19008.6921	19089.1961	19031.3320
(ϕ, θ)	(0.5,0.3)			(0.5,0.4)		
n	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
200	401.1560	401.8041	405.1714	413.3806	413.6493	417.1493
500	996.4152	999.8096	1000.8612	1026.2670	1029.3798	1030.3225
1000	1985.8681	1994.0732	1991.2345	2046.6933	2053.8856	2051.5482
5000	9918.4212	9964.3200	9930.5799	10220.8729	10261.4726	10230.6638
10000	19835.1532	19927.2010	19854.8859	20437.9308	20521.3739	20453.9049

TABLE 2.9: Comparative study between GPAR(1), PAR(1) and MTD(1) methods with respect to AIC, where the data generating process is the proposed GPAR(1) process with true (ϕ, θ) = (0.5,0.1), (0.5,0.2), (0.5,0.3), and (0.5,0.4) for four categories, where $\underset{\sim}{\pi} = (0.1, 0.2, 0.3, 0.4)'$

(ϕ, θ)	(0.5,0.1)			(0.5,0.2)		
n	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
200	442.2904	443.0689	454.8878	471.2953	473.9965	487.0142
500	1096.8140	1100.3736	1114.3573	1169.4054	1178.4645	1187.6482
1000	2188.5219	2196.7766	2207.0322	2331.9806	2350.4915	2353.6238
5000	10919.0098	10963.1610	10946.5975	11639.2092	11736.4921	11678.4017
10000	21835.2725	21925.3226	21874.9511	23276.4477	23473.0126	23337.0120
(ϕ, θ)	(0.5,0.3)			(0.5,0.4)		
n	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
200	493.3006	496.9468	509.881	510.1101	513.0433	527.2056
500	1224.1625	1234.4884	1242.868	1268.6550	1278.6857	1287.0087
1000	2442.4252	2465.0191	2463.531	2529.8263	2551.2077	2550.3022
5000	12195.7682	12314.3649	12237.612	12621.5059	12734.7821	12658.7611
10000	24375.6274	24614.4782	24441.650	25245.9892	25472.4967	25303.6516

TABLE 2.10: Comparative study between GPAR(1), PAR(1) and MTD(1) methods with respect to AIC, where the data generating process is the proposed GPAR(1) process with true (ϕ, θ) = (0.5,0.1), (0.5,0.2), (0.5,0.3), and (0.5,0.4) for five categories, where $\underset{\sim}{\pi} = (0.1, 0.2, 0.4, 0.2, 0.1)'$

(ϕ, θ)	(0.5,0.1)			(0.5,0.2)		
n	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
200	509.749	511.472	529.1306	544.6729	549.2802	567.0207
500	1261.314	1267.372	1295.6795	1348.6071	1362.9740	1382.8802
1000	2514.852	2527.806	2553.2521	2690.6927	2720.3636	2730.2520
5000	12557.676	12627.651	12609.6019	13434.2975	13588.0865	13492.5516
10000	25098.641	25239.612	25170.5919	26847.0419	27155.5779	26929.2092
(ϕ, θ)	(0.5,0.3)			(0.5,0.4)		
n	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
200	569.1382	575.6517	592.5448	587.4382	593.6598	613.8962
500	1413.1674	1431.3946	1448.6500	1460.8420	1478.1965	1496.0483
1000	2815.8204	2853.5494	2854.7115	2911.2130	2948.7143	2948.6553
5000	14056.3345	14250.3060	14109.4285	14529.2535	14720.4589	14577.8621
10000	28111.6791	28499.8295	28186.6599	29056.1556	29439.3677	29119.1684

2.5.2 Forecasting

This simulation experiment is carried out to observe the h-step ahead forecasting performances for varying h of the proposed GPAR(1) process, the PAR(1) process and the MTD(1) process. Here, we consider a forecasting measure, namely the h-step ahead percentage of true prediction (PTP(h)) (see Appendix 2.8), based on the mode predictors. In this study, we take different sets of (ϕ, θ) with three, four and five categories, where $\underset{\sim}{\pi} = (0.5, 0.1, 0.4)'$, $(0.5, 0.25, 0.15, 0.1)'$ and $(0.4, 0.1, 0.05, 0.15, 0.3)'$, respectively. Each time, we generate a data of sample size 500 through our proposed GPAR(1) process. A training set of size 400 is considered to fit three models considered for this study and a test set of size 100 is used to find the PTP(h) for $h = 1, 2, 3, 4$. This procedure is repeated for 1000 times.

The simulation results are given in Tables 2.11, 2.12 and 2.13. From the results, we can see that the GPAR process mostly performs better than the PAR and MTD processes. The number of times the GPAR(1) process performs better than other two processes for each three and four categories is greater than the number of times regarding five categories. One thing is that we obtain some PTP values for which differences between these three processes are marginal. The reason would possibly be because of the mode predictor that we consider for this study. The mode predictor is the point in the future time series support space where the density function reaches its maximum.However, overall, we can say that our proposed GPAR(1) process works better than two other processes in this simulation study.

TABLE 2.11: PTP(h) values for varying h, where the data generating process is the proposed GPAR(1) process with true (ϕ, θ) = (0.6,0.2), (0.6,0.3), (0.7,0.2), (0.7,0.3), (0.75,0.25) and (0.8,0.3) for three categories, where $\underset{\sim}{\pi} = (0.5, 0.1, 0.4)'$

(ϕ, θ)	(0.6,0.2)			(0.6,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	62.34	62.34	62.22	56.90	56.98	56.44
2	48.64	48.00	48.58	45.09	44.42	45.03
3	44.13	43.74	43.97	41.47	41.28	41.41
4	42.93	43.05	42.91	41.02	41.01	41.02
(ϕ, θ)	(0.7,0.2)			(0.7,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	66.35	66.35	66.34	59.64	59.64	59.52
2	50.36	50.31	50.23	45.70	44.74	45.66
3	44.75	43.60	44.60	41.31	40.60	41.29
4	41.78	41.23	41.76	39.67	39.59	39.50
(ϕ, θ)	(0.75,0.25)			(0.8,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	64.41	64.41	64.41	62.26	62.26	62.21
2	48.38	48.46	48.25	46.56	46.56	46.37
3	42.52	41.29	42.40	40.61	39.36	40.67
4	39.65	39.09	39.50	37.88	37.04	38.01

TABLE 2.12: PTP(h) values for varying h, where the data generating process is the proposed GPAR(1) process with true (ϕ, θ) = (0.6,0.2), (0.6,0.3), (0.7,0.2), (0.75,0.25), (0.8,0.25) and (0.8,0.3) for four categories, where $\underset{\sim}{\pi} = (0.5, 0.25, 0.15, 0.1)'$

(ϕ, θ)	(0.6,0.2)			(0.6,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	58.28	58.25	58.09	51.89	51.76	51.32
2	42.80	42.70	42.64	40.29	40.19	39.89
3	42.97	42.89	42.86	40.61	40.60	40.52
4	43.09	43.08	43.07	40.58	40.60	40.55
(ϕ, θ)	(0.7,0.2)			(0.75,0.25)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	62.23	62.23	62.22	60.06	60.06	60.01
2	44.79	43.94	44.40	42.88	41.84	42.24
3	40.56	40.41	40.38	37.47	37.42	37.18
4	40.78	40.82	40.71	37.69	37.77	37.42
(ϕ, θ)	(0.8,0.25)			(0.8,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	61.56	61.56	61.56	57.75	57.75	57.68
2	44.15	43.37	43.55	40.82	39.63	40.18
3	36.28	35.61	35.99	34.27	34.06	34.36
4	35.01	35.11	34.82	33.58	33.77	33.51

TABLE 2.13: PTP(h) values for varying h, where the data generating process is the proposed GPAR(1) process with true (ϕ, θ) = (0.6,0.2), (0.6,0.3), (0.7,0.25), (0.7,0.3), (0.75,0.25) and (0.8,0.3) for five categories, where $\underset{\sim}{\pi} = (0.4, 0.1, 0.05, 0.15, 0.3)'$

(ϕ, θ)	(0.6,0.2)			(0.6,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	55.73	55.73	55.32	48.52	48.52	47.90
2	38.47	37.91	38.00	34.38	33.46	34.11
3	34.29	33.65	33.99	31.39	31.10	31.34
4	33.05	32.80	32.81	30.71	30.57	30.72
(ϕ, θ)	(0.7,0.25)			(0.7,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	56.15	56.15	55.92	52.13	52.13	51.79
2	37.93	37.84	37.53	34.80	34.21	34.58
3	32.12	31.06	31.86	30.49	29.11	30.34
4	30.08	29.48	30.01	28.76	28.02	28.71
(ϕ, θ)	(0.75,0.25)			(0.8,0.3)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	53.42	53.42	53.30	55.52	55.52	55.42
2	35.69	35.36	35.33	37.56	37.60	36.68
3	30.62	29.06	30.25	30.16	28.12	29.85
4	28.33	27.27	28.19	27.31	25.37	27.17

2.5.3 Simulation experiment when the data generating process is the MTD(1) process

We also generate samples from the MTD(1) model with four categories to perform a model comparison study between the GPAR(1), the PAR(1) and the MTD(1) processes through AIC, where the transition probability matrix Q is taken as

$$\begin{pmatrix} 0.80 & 0.10 & 0.05 & 0.05 \\ 0.15 & 0.65 & 0.15 & 0.05 \\ 0.05 & 0.25 & 0.55 & 0.15 \\ 0.05 & 0.10 & 0.30 & 0.55 \end{pmatrix}.$$

Here also, we conduct this experiment for sample sizes 200, 500, 1000, 5000 and 10000 with 1000 replications in each case. The simulation results for average AICs are given in Table 2.15 in Appendix (see Section 2.8). From the results, we can see the MTD(1) process has lower AIC values than the GPAR(1) process for sample sizes 1000, 5000 and 10000. The GPAR(1) process has lower AIC values than the PAR(1) process for all the cases.

For studying the predictive performances, we generate data of sample size 500

through our proposed MTD(1) process for three categories with transition probability matrix as $\begin{pmatrix} 0.70 & 0.30 & 0.00 \\ 0.30 & 0.40 & 0.30 \\ 0.20 & 0.50 & 0.30 \end{pmatrix}$ and four categories $\begin{pmatrix} 0.80 & 0.10 & 0.05 & 0.05 \\ 0.15 & 0.65 & 0.15 & 0.05 \\ 0.05 & 0.25 & 0.55 & 0.15 \\ 0.05 & 0.10 & 0.30 & 0.55 \end{pmatrix}$. A training set of size 400 is considered to fit three models considered for this study and a test set of size 100 is used to find the PTP(h) for $h = 1, 2, 3, 4$. This procedure is repeated for 1000 times. The numerical results for this study is given in Table 2.16 in Appendix (see Section 2.8). For three categories, the MTD(1) process has larger PTP(h) values than the GPAR(1) process and the PAR(1) process for $h = 1, 2$ and all h, respectively, and similar PTP(h) values for $h = 3, 4$ as the GPAR(1) process. For four categories, we can see that for $h = 1, 2$, the MTD(1) process has little lower PTP(h) values than other two models. However, for $h = 3, 4$, the MTD(1) process performs better than other two models.

2.6 Data analysis

In this study, we consider the Air Quality Index data of Mumbai, which was recorded in 2021. The data is available at Mumbai AQI Data, n.d. in AQI-IN version (see AQI IN 1, n.d. and AQI IN 2, n.d.). Usually, there are six categories for measuring AQI, which are "Good", "Moderate", "Poor", "Unhealthy", "Severe" and "Hazardous". The details can be found at the aforementioned url addresses. This dataset has a missing value which is for January 24. We fill that data-point by the category that appears most in the month of January, which is "Unhealthy". In this study, we combine these six categories to four. Those four categories are: category 0 for AQI from 0 to 100, category 1 for AQI from 101 to 200, category 2 for AQI from 201 to 300, and category 3 for AQI with more than 300. The snapshot of the data is given in Figure 2.1. Therefore, the number of parameters to be estimated for the GPAR(1), the PAR(1) and the MTD(1) processes are 5, 4 and 12, respectively. The proportions of four categories, i.e., category 0, category 1, category 2 and category 3, in the data, are 0.4712, 0.3041, 0.1562 and 0.0685, respectively. This indicates that "Good" and "Moderate" categories are recorded in Mumbai mostly in 2021.

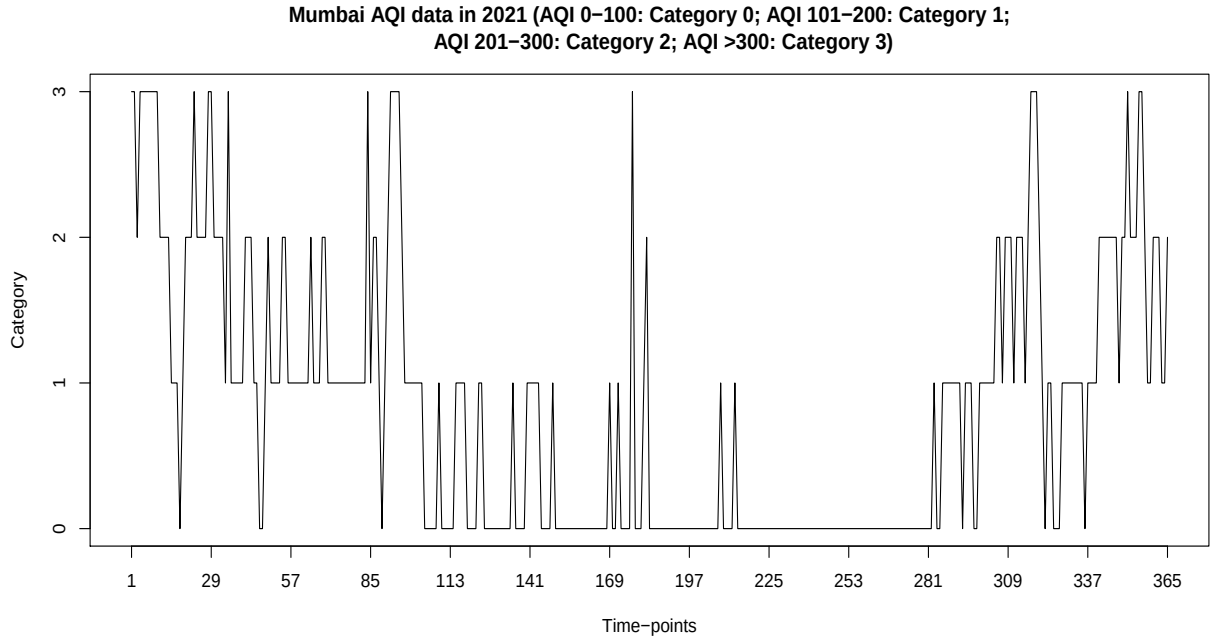


FIGURE 2.1: The AQI data of Mumbai recorded in 2021

Here, we first calculate the auto-odds ratio function (AORF), i.e., $r(h)$ that denotes the AORF between Y_t and Y_{t-h}. Maiti and Biswas, 2018 extensively showed how the AORF helps to measure the order in their study (see Section 2.2). They showed that a decreasing pattern of $\log(r(h))$ suggests that an AR of order 1 process would possibly be a good fit for the data. The $\log(r(h))$ for various lagged values for our data are computed and presented in Figure 2.2. We can see that there is a decreasing pattern in the AORF values as h increases. Moreover, as mentioned earlier, to measure the order more precisely, we calculate Cohen's κ estimates based estimates of PACFs $\rho_p(h)$ (see Weiß and Göb, 2008, Weiß, 2011b; Weiß, 2013) for some lagged values. If we can ignore the values for $h > 1$, which are all less than 0.17 (see Figure 2.3), we can say that an AR(1) process would possibly be a decent fit for our data.

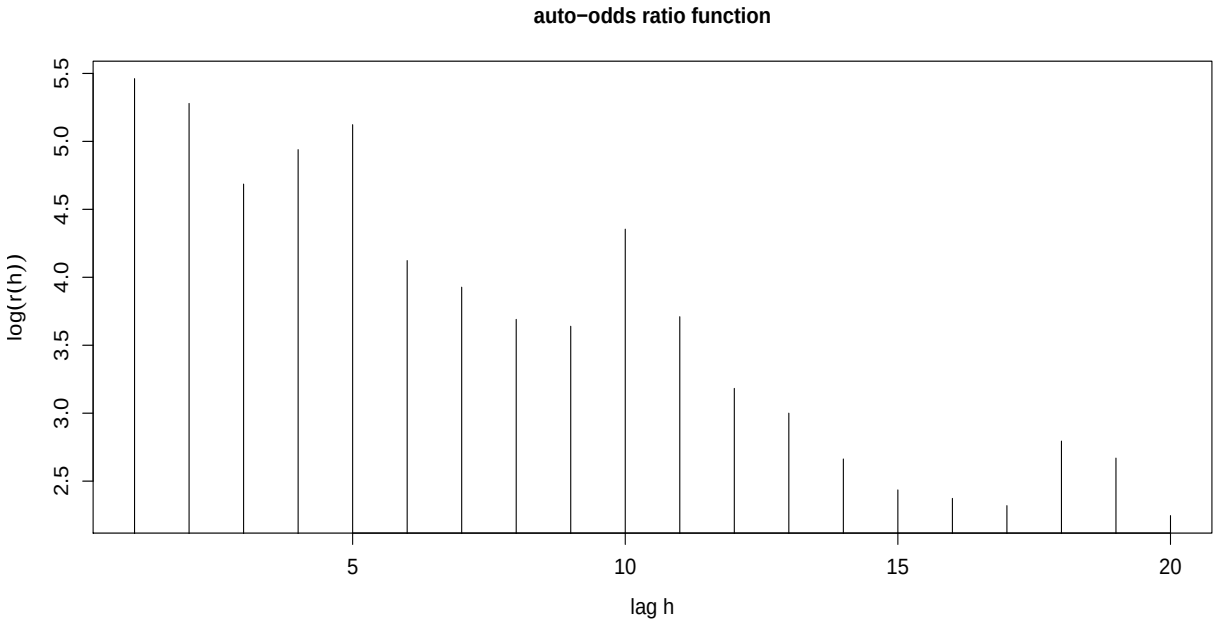


FIGURE 2.2: The AORF of Mumbai AQI data in 2021

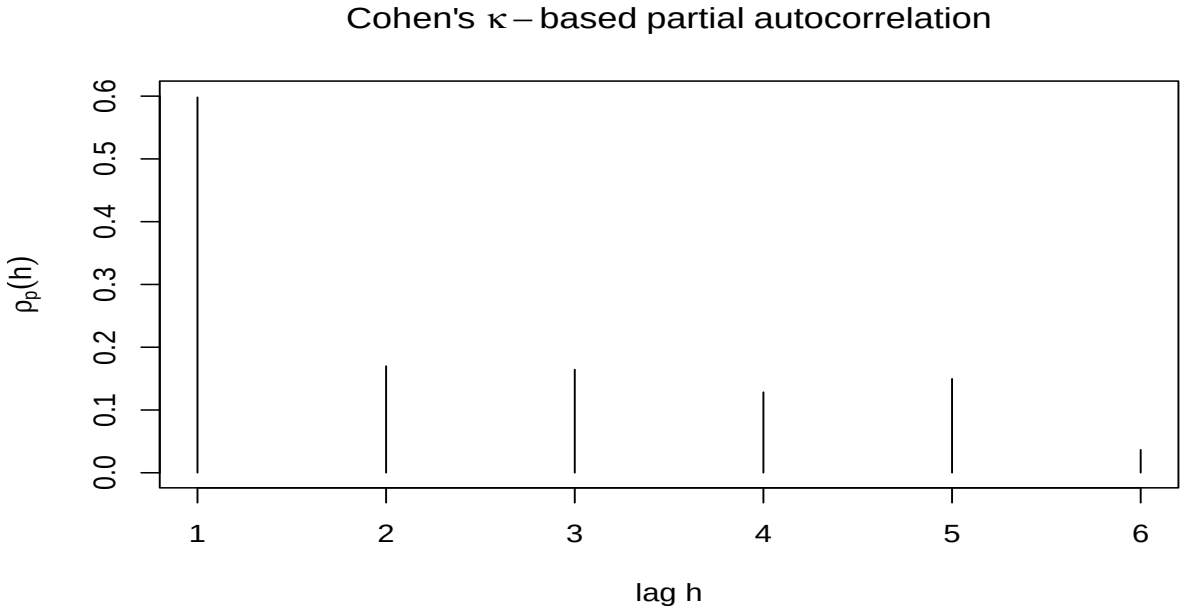


FIGURE 2.3: The plot of Cohen's κ-based estimates of PACFs of Mumbai AQI data in 2021

We fit the GPAR(1), the PAR(1), the MTD(1) processes to this dataset. The estimated model parameters for the GPAR(1) process are $(\hat{\phi}, \hat{\theta}) = (0.8347, 0.1548)$. The estimated model parameter for the PAR(1) process is $\hat{\phi} = 0.5895$. The sample

proportions are $(0.4712, 0.3041, 0.1562, 0.0685)'$. Moreover, the transition probability matrix Q for the MTD(1) process is estimated as

$$\begin{pmatrix} 0.872 & 0.122 & 0 & 0.006 \\ 0.180 & 0.649 & 0.153 & 0.018 \\ 0.018 & 0.286 & 0.571 & 0.125 \\ 0.040 & 0.080 & 0.320 & 0.560 \end{pmatrix}.$$

The AIC value of the MTD(1) process is lowest (which is 530.2993), and the proposed GPAR(1) process has lower AIC (which is 568.1683) value than that of the PAR(1) process (which is 619.2919). The lowest AIC value for the MTD(1) process is somewhat expected since it is more generalized process than other two processes, especially when the number of categories is four. The hint of this property is recorded in our simulation study.

Additionally, we also study the forecasting performances of the processes with respect to the PTP measure. We divide the data into parts: (i) first 315 days for the training set and (ii) last 50 days for the test set. The forecasting results are given in Table 2.14. As we can see, our proposed GPAR(1) process performs better for mode predictor. There is no unique better performing model between the PAR(1) process and the MTD(1) process in terms of the PTP measure with respect to the mode predictor. For the median predictor, the MTD(1) process has a higher PTP value for $h = 2$ compared to the GPAR(1) process. And there is no unique better performing model between the PAR(1) process and the GPAR(1) process, and similarly between the PAR(1) process and the MTD(1) process.

TABLE 2.14: Forecasting results for Mumbai 2021 AQI data analysis

Measures	PTP[mode](h)			PTP[median](h)		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	58.00	58.00	58.00	58.00	58.00	58.00
2	42.86	42.86	32.65	42.86	34.69	46.94
3	43.75	22.92	33.33	33.33	35.42	33.33

2.7 Conclusions

In this chapter, we have proposed a generalized Pegram's operator based autoregressive process of order 1 (GPAR(1) process), mainly for ordinal categories. The generalization allows us to overcome the limitation of the indicator kernel involved in the PAR process. This limitation is that it assigns weights only to scenarios in the transition probabilities, where the same category appears at both

the current and previous time-points, but it does not assign any weight to situations where a different category can occur at the current time-point compared to the category at the previous time-point. The proposed GPAR process requires fewer parameters to be estimated compared to the MTD process, particularly when dealing with a larger number of categories. The reason for this is that the GPAR process incorporates only one additional parameter in comparison to the PAR process. Overall, the proposed GPAR process offers advantages over both the PAR process and the MTD process, when it comes to modelling categorical time series data that has a finite number of ordinal categories. In this chapter, we have examined various theoretical findings concerning the proposed GPAR(1) process, which encompasses the h-step ahead forecasting distribution.

We have conducted comprehensive simulation experiments that have demonstrated the empirical consistency of the estimated model parameters using the MLE method. Furthermore, we have examined the efficacy of our suggested procedure in relation to the AIC and the PTP[mode](h) measure through different simulation studies. For practical demonstration of our proposed procedure, we have examined the Air Quality Index (AQI) data of Mumbai in 2021. In the data analysis, the GPAR(1) process has performed better than the PAR(1) process in terms of both the AIC and the PTP[mode](h) measure. Nevertheless, the MTD(1) process has exhibited a lower AIC compared to the GPAR(1) process. The reason for this could be that the MTD(1) process is more generalized than the other two processes, when there are fewer categories. We have observed a subtle indication of this phenomenon in the simulation study regarding model comparison based on the AIC. However, during the simulation studies, it has been observed that the GPAR(1) process has mostly given better performances than the MTD(1) process in terms of both AIC and PTP[mode](h) measures. Furthermore, the GPAR(1) process has exhibited superior performance compared to the MTD(1) process in terms of PTP[mode](h), especially for $h = 2$ and 3, in the data analysis. However, for the median predictor, the MTD(1) process has a higher PTP value for only $h = 2$ compared to the GPAR(1) process. And there is no unique better performing model between the PAR(1) process and the GPAR(1) process, and similarly between the PAR(1) process and the MTD(1) process. In summary, we expect that our method will be a useful approach for analyzing categorical time series data in the future.

2.8 Appendix

For simplicity, we assume $i \equiv C_i$ in the derivations, i.e., $i = m_0$ for C_{m_0} (m_0-th category).

Appendix A.

Here, we have

$$\frac{1}{M_j}\sum_{i=0}^{m}\theta^{|i-j|} = 1 \implies \frac{1}{M_j}\left[\sum_{i=0}^{j}\theta^{j-i} + \sum_{i=j+1}^{m}\theta^{i-j}\right] = 1. \tag{2.8.1}$$

Now, we can write $\sum_{i=0}^{j}\theta^{j-i} = \dfrac{1-\theta^{j+1}}{1-\theta}$, and $\sum_{i=j+1}^{m}\theta^{i-j} = \dfrac{\theta-\theta^{m-j+1}}{1-\theta}$. So, from equation (2.8.1), we get

$$\frac{1}{M_j}\frac{1-\theta^{j+1}+\theta-\theta^{m-j+1}}{1-\theta} = 1 \implies M_j = \frac{1-\theta}{1-\theta^{j+1}+\theta-\theta^{m-j+1}}.$$

There, we have $K(j \mid i) = \dfrac{(1-\theta)\theta^{|i-j|}}{1-\theta^{j+1}+\theta-\theta^{m-j+1}}$, where $i = 0, 1, \ldots, m$ and $j = 0, 1, \ldots, m$.

Appendix B.

For GPAR(1) process, we have

$$P(Y_t = i \mid Y_{t-1}) = \phi K(Y_{t-1} \mid i) + (1-\phi)\pi_i.$$

Now, with the the assumption that $P(Y_{t-h} = i) = p_i$, we can find

$$\begin{aligned}\sum_{j=0}^{m} K(Y_{t-1} = j \mid i)\, P(Y_{t-1} = j) &= \frac{\theta^{|i|}}{M_0}P(Y_{t-1}=0) + \frac{\theta^{|i-1|}}{M_1}P(Y_{t-1}=1) + \ldots + \\ &\qquad \frac{\theta^{|i-m|}}{M_m}P(Y_{t-1}=m) \\ &= \left(\frac{\theta^{|i|}}{M_0}, \frac{\theta^{|i-1|}}{M_1}, \ldots, \frac{\theta^{|i-m|}}{M_m}\right)\begin{pmatrix} p_0 \\ \vdots \\ p_m \end{pmatrix}\end{aligned}$$

So, we can write

$$\begin{aligned}\begin{pmatrix} p_0 \\ \vdots \\ p_m \end{pmatrix} &= \phi\,(K(\cdot \mid 0), \ldots, K(\cdot \mid m)) \begin{pmatrix} p_0 \\ \vdots \\ p_m \end{pmatrix} + (1-\phi)\underset{\sim}{\pi} \\ \Rightarrow \quad \underset{\sim}{p} &= \phi K^T \underset{\sim}{p} + (1-\phi)\underset{\sim}{\pi} \\ \Rightarrow \quad (I - \phi K^T)\underset{\sim}{p} &= (1-\phi)\underset{\sim}{\pi}.\end{aligned}$$

Here, $K\underset{\sim}{1} = \underset{\sim}{1}$, since each row of K sums to 1, where $\underset{\sim}{1} = [1, 1, \ldots, 1]'$. So, this proves that 1 is an eigenvalue of K.
Now, suppose, there exists λ and non-zero $\underset{\sim}{x}$ such that $K\underset{\sim}{x} = \lambda\underset{\sim}{x}$ (i.e., λ is an eigenvalue of K). So, we have

$$K_{i0}x_0 + \ldots + K_{im}x_m = \lambda x_i, \tag{2.8.2}$$

by comparing the i-th row of the both sides for $i = 0, 1, \ldots, m$.
Let $\mid x_g \mid = max\,(\mid x_0 \mid, \ldots, \mid x_m \mid)$, i.e., x_g is the entry $\underset{\sim}{x}$ that has the maximal absolute value. Note that, $\mid x_g \mid > 0$ since, otherwise, we have $\underset{\sim}{x} = 0$ that contradicts that an eigenvalue is non-zero.
From equation (2.8.2), with $i = m$, we have

$\mid \lambda \mid\mid x_g \mid = \mid K_{g0}x_0 + \ldots + K_{gm}x_m \mid$
$\leq K_{g0} \mid x_0 \mid + \ldots + K_{gm} \mid x_m \mid$ (by the triangle inequality and $K_{ij} > 0$)
$\leq K_{g0} \mid x_g \mid + \ldots + K_{gm} \mid x_g \mid = (K_{g0} + \ldots + K_{gm}) \mid x_g \mid$ (since $\mid x_g \mid$ is maximum).
Hence, $\mid \lambda \mid\mid x_g \mid < \mid x_g \mid \Rightarrow \mid \lambda \mid \leq 1$(since $\mid x_g \mid > 0$).
Now, let us assume, K has eigenvalues ${\lambda_0}^K, \ldots, {\lambda_m}^K$, where $\mid {\lambda_j}^K \mid \leq 1$ for $j = 0, 1, \ldots, m$.
Hence, $(I - \phi K)$ has the eigenvalues $(1 - \phi{\lambda_0}^K), \ldots, (1 - \phi{\lambda_m}^K)$.
Hence, $(I - \phi K)$ will be singular if at least one eigenvalue is 0, i.e., $1 - \phi{\lambda_j}^K = 0$, which implies that ${\lambda_j}^K = \dfrac{1}{\phi}$. But, we know $\phi \in (0, 1)$, and that means ${\lambda_j}^K > 1$, which is a contradiction.
So, $(I - \phi K)$ is non-singular $\implies$ $(I - \phi K)$ is invertible $\implies$ $(I - \phi K)^T = (I - \phi K^T)$ is invertible.

Appendix C.

Here, A is a non-singular matrix of order n, and $B^{n\times n}$ and $C^{n\times n}$ be such matrices that $(A + BC)$ is non-singular. Let us assume that $A + BC = P$.

Now, $A + BC = P \implies A^{-1}(A + BC) = A^{-1}P \implies I_n + A^{-1}BC = A^{-1}P \implies C[I_n + A^{-1}BC] = CA^{-1}P \implies C + CA^{-1}BC = CA^{-1}P \implies [I_n + CA^{-1}B]C = CA^{-1}P \implies C = [I_n + CA^{-1}B]^{-1}CA^{-1}P \implies BC = B[I_n + CA^{-1}B]^{-1}CA^{-1}P \implies A + BC = A + B[I_n + CA^{-1}B]^{-1}CA^{-1}P \implies P = A + B[I_n + CA^{-1}B]^{-1}CA^{-1}P \implies A = [I_n - B(I_n + CA^{-1}B)^{-1}CA^{-1}]P \implies P^{-1} = A^{-1}[I_n - B(I_n + CA^{-1}B)^{-1}CA^{-1}]$
$\implies (A + BC)^{-1} = A^{-1} - A^{-1}B(I_n + CA^{-1}B)^{-1}CA^{-1}.$
Putting $A = I_{m+1}(= I), C = (-\phi), B = K^T$ in above equation, we have
$(I - \phi K^T)^{-1} = I + IK^T(I - \phi K^T)^{-1}\phi I \implies (I - \phi K^T)^{-1} = I + \phi K^T(I - \phi K^T)^{-1} \implies \underset{\sim}{1}'(I - \phi K^T)^{-1} = \underset{\sim}{1}' + \phi\underset{\sim}{1}'K^T(I - \phi K^T)^{-1} \implies \underset{\sim}{1}'(I - \phi K^T)^{-1} = \underset{\sim}{1}' + \phi\underset{\sim}{1}'(I - \phi K^T)^{-1} \implies \underset{\sim}{1}'(I - \phi K^T)^{-1}\underset{\sim}{\pi} = \underset{\sim}{1}'\underset{\sim}{\pi} + \phi\underset{\sim}{1}'(I - \phi K^T)^{-1}\underset{\sim}{\pi} \implies (1 - \phi)\underset{\sim}{1}'(I - \phi K^T)^{-1}\underset{\sim}{\pi} = \underset{\sim}{1}'\underset{\sim}{\pi} = 1.$

Appendix D.

We have $p_h(i \mid j) = P(Y_{t+h} = i \mid Y_t = j)$. In this derivation, $\underset{\sim}{K^h}(i)$ is the $(i+1)$th column of K^h and $K_h(j, i)$ is the $(j+1, i+1)$th element of K^h.
Now, for $h = 1$ (i.e., $p_1(i \mid j)$), we can write

$$\begin{aligned} p_1(i \mid j) &= P(Y_{t+1} = i \mid Y_t = j) \\ &= \phi K(j \mid i) + (1 - \phi)\pi_i \\ &= \phi\frac{\theta^{|i-j|}}{M_j} + (1 - \phi)\pi_i \\ &= \phi K[j + 1, i + 1] + (1 - \phi)\pi_i \\ &= \phi K_1(j, i) + (1 - \phi)\pi_i. \end{aligned}$$

For $h = 2$ (i.e., $p_2(i \mid j)$), we have

$$\begin{aligned}
p_2(i \mid j) &= P(Y_{t+2} = i \mid Y_t = j) \\
&= \sum_{i_1=0}^{m} P(Y_{t+2} = i \mid Y_{t+1} = i_1) \times P(Y_{t+1} = i_1 \mid Y_t = j) \\
&= \sum_{i_1=0}^{m} \left[\phi \frac{\theta^{|i-i_1|}}{M_{i_1}} + (1-\phi)\pi_i\right] \times \left[\phi \frac{\theta^{|i_1-j|}}{M_j} + (1-\phi)\pi_{i_1}\right] \\
&= \phi^2 \sum_{i_1=0}^{m} \frac{\theta^{|i-i_1|+|i_1-j|}}{M_j M_{i_1}} + \phi(1-\phi) \sum_{i_1=0}^{m} \frac{\theta^{|i-i_1|}}{M_{i_1}} \pi_{i_1} + \phi(1-\phi)\pi_i \sum_{i_1=0}^{m} \frac{\theta^{|i_1-j|}}{M_j} + \\
&\qquad (1-\phi)^2 \pi_i \sum_{i_1=0}^{m} \pi_{i_1} \\
&= \phi^2 K^2[j+1, i+1] + \phi(1-\phi)\underset{\sim}{\pi}' K[\,, i+1] + \phi(1-\phi)\pi_i + (1-\phi)^2\pi_i \\
&= \phi^2 K^2[j+1, i+1] + \phi(1-\phi)\underset{\sim}{\pi}' K[\,, i+1] + (1-\phi)\pi_i \\
&= \phi^2 K_2(j, i) + (1-\phi) \sum_{n=1}^{2-1} \phi^{2-n} \underset{\sim}{\pi}' \underset{\sim}{K}^{2-n}(i) + (1-\phi)\pi_i.
\end{aligned}$$

Again, for $h = 3$ (i.e., $p_3(i \mid j)$), we have

$$\begin{aligned}
p_3(i \mid j) &= P(Y_{t+3} = i \mid Y_t = j) \\
&= \sum_{i_2=0}^{m} P(Y_{t+3} = i \mid Y_{t+2} = i_2) \times P(Y_{t+2} = i_2 \mid Y_t = j) \\
&= \sum_{i_2=0}^{m} \left[\phi \frac{\theta^{|i-i_2|}}{M_{i_2}} + (1-\phi)\pi_i\right] \times \\
&\qquad \left[\phi^2 \sum_{i_1=0}^{m} \frac{\theta^{|i_2-i_1|+|i_1-j|}}{M_j M_{i_1}} + \phi(1-\phi) \sum_{i_1=0}^{m} \frac{\theta^{|i_2-i_1|}}{M_{i_1}} \pi_{i_1} + (1-\phi)\pi_{i_2}\right] \\
&= \phi^3 \sum_{i_2=0}^{m} \sum_{i_1=0}^{m} \frac{\theta^{|i-i_2|+|i_2-i_1|+|i_1-j|}}{M_j M_{i_1} M_{i_2}} + \phi^2(1-\phi) \sum_{i_2=0}^{m} \sum_{i_1=0}^{m} \frac{\theta^{|i-i_2|+|i_2-i_1|}}{M_{i_1} M_{i_2}} \pi_{i_1} + \\
&\qquad \phi(1-\phi) \sum_{i_2=0}^{m} \frac{\theta^{|i-i_2|}}{M_{i_2}} \pi_{i_2} + \phi^2(1-\phi)\pi_i \sum_{i_2=0}^{m} \sum_{i_1=0}^{m} \frac{\theta^{|i_2-i_1|+|i_1-j|}}{M_j M_{i_1}} + \\
&\qquad \phi(1-\phi)^2 \pi_i \sum_{i_2=0}^{m} \sum_{i_1=0}^{m} \frac{\theta^{|i_2-i_1|}}{M_{i_1}} \pi_{i_1} + (1-\phi)^2 \pi_i \sum_{i_2=0}^{m} \pi_{i_2} \\
&= \phi^3 K^2[j+1, i+1] + \phi^2(1-\phi)\underset{\sim}{\pi}' K^2[\,, i+1] + \phi(1-\phi)\underset{\sim}{\pi}' K[\,, i+1] + \\
&\qquad (1-\phi)(\phi^2 + \phi - \phi^2 + 1 - \phi)\pi_i \\
&= \phi^3 K_3(j, i) + (1-\phi) \sum_{n=1}^{3-1} \phi^{3-n} \underset{\sim}{\pi}' \underset{\sim}{K}^{3-n}(i) + (1-\phi)\pi_i.
\end{aligned}$$

Hence, assuming

$$
\begin{aligned}
p_n(i \mid j) &= \phi^n K_n(j,i) + (1-\phi)\sum_{l=1}^{n-1}\phi^{n-l}\underset{\sim}{\pi}'\underset{\sim}{K}^{n-l}(i) + (1-\phi)\pi_i \\
&= \phi^n \sum_{i_{n-1}=0}^{m}\cdots\sum_{i_1=0}^{m}\frac{\theta^{|i-i_{n-1}|+|i_{n-1}-i_{n-2}|+\ldots+|i_1-j|}}{M_j\ldots M_{i_{n-1}}} + \\
&\quad \phi^{n-1}(1-\phi)\sum_{i_{n-1}=0}^{m}\cdots\sum_{i_1=0}^{m}\frac{\theta^{|i-i_{n-1}|+|i_{n-1}-i_{n-2}|+\ldots+|i_2-i_1|}}{M_{i_1}\ldots M_{i_{n-1}}}\pi_{i_1} + \\
&\quad \phi^{n-2}(1-\phi)\sum_{i_{n-1}=0}^{m}\cdots\sum_{i_2=0}^{m}\frac{\theta^{|i-i_{n-1}|+|i_{n-1}-i_{n-2}|+\ldots+|i_3-i_2|}}{M_{i_2}\ldots M_{i_{n-1}}}\pi_{i_2} + \ldots + \\
&\quad \phi(1-\phi)\sum_{i_{n-1}=0}^{m}\frac{\theta^{|i-i_{n-1}|}}{M_{i_{n-1}}}\pi_{i_{n-1}} + (1-\phi)\pi_i,
\end{aligned}
$$

We show that

$$
p_{n+1}(i \mid j) = \phi^{n+1}K_{n+1}(j,i) + (1-\phi)\sum_{l=1}^{n}\phi^{n+1-l}\underset{\sim}{\pi}'\underset{\sim}{K}^{n+1-l}(i) + (1-\phi)\pi_i.
$$

$$
\begin{aligned}
p_{n+1}(i \mid j) &= P(Y_{t+n+1} = i \mid Y_t = j) \\
&= \sum_{i_n=0}^{m} P(Y_{t+n+1} = i \mid Y_{t+n} = i_n) \times P(Y_{t+n} = i_n \mid Y_t = j) \\
&= \sum_{i_n=0}^{m} \left[\phi \frac{\theta^{|i-i_n|}}{M_{i_n}} + (1-\phi)\pi_i \right] \times \left[\phi^n \sum_{i_{n-1}=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i_n - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_1 - j|}}{M_j \dots M_{i_{n-1}}} + \right. \\
&\quad \phi^{n-1}(1-\phi) \sum_{i_{n-1}=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i_n - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_2 - i_1|}}{M_{i_1} \dots M_{i_{n-1}}} \pi_{i_1} + \\
&\quad \phi^{n-2}(1-\phi) \sum_{i_{n-1}=0}^{m} \dots \sum_{i_2=0}^{m} \frac{\theta^{|i_n - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_3 - i_2|}}{M_{i_2} \dots M_{i_{n-1}}} \pi_{i_2} + \dots + \\
&\quad \left. \phi(1-\phi) \sum_{i_{n-1}=0}^{m} \frac{\theta^{|i_n - i_{n-1}|}}{M_{i_{n-1}}} \pi_{i_{n-1}} + (1-\phi)\pi_{i_n} \right] \\
&= \phi^{n+1} \sum_{i_n=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_1 - j|}}{M_j \dots M_{i_n}} + \\
&\quad \phi^n(1-\phi) \sum_{i_n=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_2 - i_1|}}{M_{i_1} \dots M_{i_n}} \pi_{i_1} + \dots + \\
&\quad \phi^2(1-\phi) \sum_{i_n=0}^{m} \sum_{i_{n-1}=0}^{m} \frac{\theta^{|i - i_n| + |i_n - i_{n-1}|}}{M_{i_n - 1} M_{i_n}} \pi_{i_{n-1}} + \phi(1-\phi) \sum_{i_n=0}^{m} \frac{\theta^{|i-i_n|}}{M_{i_n}} \pi_{i_n} + \\
&\quad (1-\phi)\pi_{i_n} + \phi^n(1-\phi)\pi_i \sum_{i_n=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_1 - j|}}{M_j \dots M_{i_{n-1}}} + \\
&\quad \phi^{n-1}(1-\phi)^2 \pi_i \sum_{i_n=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_2 - i_1|}}{M_{i_1} \dots M_{i_{n-1}}} \pi_{i_1} + \dots + \\
&\quad \phi(1-\phi)^2 \pi_i \sum_{i_n=0}^{m} \sum_{i_{n-1}=0}^{m} \frac{\theta^{|i - i_n| + |i_n - i_{n-1}|}}{M_{i_{n-1}}} \pi_{i_{n-1}} + (1-\phi)^2 \pi_i \sum_{i_n=0}^{m} \pi_{i_n} \\
&= \phi^{n+1} \sum_{i_n=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_1 - j|}}{M_j \dots M_{i_n}} + \\
&\quad \phi^n(1-\phi) \sum_{i_n=0}^{m} \dots \sum_{i_1=0}^{m} \frac{\theta^{|i - i_{n-1}| + |i_{n-1} - i_{n-2}| + \dots + |i_2 - i_1|}}{M_{i_1} \dots M_{i_n}} \pi_{i_1} + \dots + \\
&\quad \phi^2(1-\phi) \sum_{i_n=0}^{m} \sum_{i_{n-1}=0}^{m} \frac{\theta^{|i - i_n| + |i_n - i_{n-1}|}}{M_{i_n - 1} M_{i_n}} \pi_{i_{n-1}} + \phi(1-\phi) \sum_{i_n=0}^{m} \frac{\theta^{|i-i_n|}}{M_{i_n}} \pi_{i_n} + \\
&\quad (1-\phi)\pi_{i_n} + \pi_i \left[\phi^n(1-\phi) + (1-\phi)^2 \left(\phi^{n-1} + \dots + \phi + 1 \right) \right].
\end{aligned}
$$

From the above expression, we get the desired result which is

$$p_{n+1}(i \mid j) = \phi^{n+1} K_{n+1}(j,i) + (1-\phi) \sum_{l=1}^{n} \phi^{n+1-l} \underset{\sim}{\pi}' \underset{\sim}{K}^{n+1-l}(i) + (1-\phi)\pi_i.$$

Appendix E.

TABLE 2.15: Comparative study between GPAR(1), PAR(1) and MTD(1) methods with respect to AIC, where the data generating process is the MTD(1) process with four catgories

n	GPAR(1)	PAR(1)	MTD(1)
200	394.7962	407.4833	396.0537
500	978.8443	1013.383	979.9345
1000	1953.833	2025.099	1941.769
5000	9750.622	10113.03	9633.437
10000	19486.76	20221.65	19231.34

TABLE 2.16: PTP(h) values for varying h, where the data generating process is the MTD(1) process with three and four categories

No. of categories	Three			Four		
h	GPAR(1)	PAR(1)	MTD(1)	GPAR(1)	PAR(1)	MTD(1)
1	52.76	51.62	55.30	65.96	65.96	65.79
2	47.91	47.39	48.42	49.81	49.81	49.46
3	47.07	46.84	47.06	40.52	39.61	41.61
4	46.76	46.70	46.75	34.82	34.19	37.45

Appendix F.

The mathematical form of PTP(h) (i.e., h-step ahead percentage of true prediction) is given by

$$PTP(h) = \frac{\sum_{t=r}^{r+s-h} I\left(Y_{t+h} = \hat{Y}_{t+h}\right)}{s-h+1} \times 100\%, \tag{2.8.3}$$

where total sample size is $(r+s)$ out of which training set consists of first r observations (to estimate the parameters of the model) and test set contains remaining s observations (based on which some descriptive measures of forecasting accuracy are studied using the predicted values $\hat{Y}_t$'s). In our study, $\hat{Y}_t$'s are mode (for both simulation study and data analysis) and median (for data analysis) predictors.

Chapter 3

A modified Pegram's operator based autoregressive process for truncated counts

3.1 Introduction

In the previous chapter (Chapter 2), we study the Pegram's operator-based autoregressive (PAR) process for categorical time series data. However, as discussed earlier, this method is also useful for modelling count time series(see Biswas and Song, 2009 and Chapters 2 to 5 of Weiß, 2018). Different studies based on the Pegram operator and thinning operators have also been conducted recently. Maiti et al., 2014 studied a zero-inflated Pegram's operator based integer-valued time series process with zero-inflated Poisson marginals. In Khoo, Ong, and Biswas, 2017, a new INAR(1) model based on Pegram and thinning operators was introduced. Shirozhan, Mohammadpour, and Bakouch, 2019 presented a new geometric INAR(1) model that combines Pegram and generalized binomial thinning operators.

However, for a count time series data, the PAR process has also the disadvantage that it uses the indicator kernel, which makes this model a bit restrictive. It only assigns weight to those scenarios in the transition probabilities, in which the same event that transpired at the previous time-point also occurs at the current time-point, i.e., when the same event happens at both time-points. Other scenarios are disregarded by the indicator kernel. Consequently, in the context of count time series, we suggest a new modified PAR process of order 1 (mPAR(1)) that employs a modified kernel. The new kernel assigns a certain amount of weight

to scenarios in which an event that differs from the one that occurred at the previous time-point can also occur at the current time-point. We study count time series data in this chapter, more precisely truncated count data. In order to model truncated counts, the proposed mPAR(1) process is developed using the right-truncated Poisson distribution. We discuss the distributional properties of our proposed method throughout the chapter. The h-step ahead forecasting distribution is also studied. Extensive simulation experiments are carried out to assess the utility of our proposed method. In both the simulation study and the data application, we compare our proposed mPAR(1) process to the PAR(1) process. We analyze a real dataset as a practical demonstration of our proposed method. This chapter is partially based on Chattopadhyay et al., 2024b.

The chapter is structured in the following manner. In the subsequent section (Section 3.2), we present an in-depth analysis of the modified kernel and the mathematical characteristics of our proposed methodology. The mathematical properties encompass a comprehensive examination of the distributional properties and the forecasting distribution for h-step ahead. Following Section 3.3 delves into the discussion of the model parameter estimation method. Extensive simulation studies are conducted in Section 3.4 to assess the empirical consistency of the estimated model parameters. This section also includes a comprehensive study comparing the mPAR(1) process and the PAR(1) process. Following this, in Section 3.5, a real dataset is examined to provide a practical demonstration of the proposed process. In the concluding section (Section 3.6), some concluding remarks are made regarding our study. All of the proofs are provided in the Appendix (refer to Section 3.7).

3.2 The mPAR(p) process

3.2.1 The modified kernel

In this section, we discuss the construction of the proposed modified Pegram's AR process of order p (mPAR(p) process) from the idea of simple Pegram's AR process of order p (PAR(p) process). Then, in the subsequent section, we discuss about mPAR(1) process, which is the main focus in our study.

The problem with the indicator kernel associated with the PAR process in the context of count time series is that it gives some weight when i-th count is equal to j-th count but when $i \neq j$, it does not contribute any weight in the conditional distribution setup. To remove this drawback, we assign some kind of weight even

when $i \neq j$. Here is our proposed kernel function

$$W(j \mid i) = \begin{cases} \eta & \text{if } i = j \\ (1-\eta)/k & \text{if } i \neq j \end{cases} \tag{3.2.1}$$

where $\eta \in (0,1)$ and k indicates that the count time series contains counts $0, 1, \ldots, k$. The value of k can be determined by $max(Y_1, \ldots, Y_n)$ from a given count time series data. The main motivation for this particular kernel is to try new and simpler weights than those used in Chapter 2 to model discrete-valued time series data. For this study, the data is a right-truncated count time series.

By incorporating the kernel (equation (3.2.1)) into the conditional probability distribution setup of the PAR(p) process (see equation (1.4.2)), we get

$$P(Y_t = i \mid Y_{t-1}, \ldots, Y_{t-p}) = \sum_{j=1}^{p} \phi_j W(Y_{t-j} \mid i) + (1 - \sum_{j=1}^{p} \phi_j)\pi_i. \tag{3.2.2}$$

Here, for our proposed model, $p_i = P(Y_t = i)$ and $\pi_i = P(\varepsilon_t = i)$ are not same, whereas $p_i = \pi_i$ for the PAR(p) process. However, for our proposed process, p_i and π_i are somewhat related, which we establish in the subsequent section.

3.2.2 Mathematical properties

Result 3.1 *Taking expectation in both sides of equation (3.2.2), we get the marginal distribution of Y_t as follows*

$$P(Y_t = i) = \frac{\frac{1-\eta}{k}\sum_{i=1}^{p}\phi_i}{1 - \left(\eta - \frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i} + \frac{1 - \sum_{i=1}^{p}\phi_i}{1 - \left(\eta - \frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i}\pi_i. \tag{3.2.3}$$

The derivation of this result is presented in Appendix (see Section 3.7).
The form of the conditional expectation of the process is given as

$$\begin{aligned} E(Y_t \mid Y_{t-1}, \ldots, Y_{t-p}) &= \left(\eta - \frac{1-\eta}{k}\right)(\phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p}) + \\ &\quad \frac{(k+1)(1-\eta)}{2}\sum_{i=1}^{p}\phi_i + \left(1 - \sum_{i=1}^{p}\phi_i\right)E(\varepsilon_t). \end{aligned}$$

The derivation of this result is presented in Appendix (see Section 3.7).

3.2.3 The mPAR(1) process

Conditional and marginal properties

In this section, we study some theoretical results regarding the proposed mPAR process of order 1 (mPAR(1)), which is the focal point of this chapter.

If we put $p = 1$ in equation (3.2.3), then we get the marginal distribution form of Y_t which is given by

$$P(Y_t = i) = \frac{\frac{1-\eta}{k}\phi}{1-\left(\eta - \frac{1-\eta}{k}\right)\phi} + \frac{1-\phi}{1-\left(\eta - \frac{1-\eta}{k}\right)\phi}\pi_i.$$

Note 3.1 *For our proposed mPAR(1) process, the choice of k has to be finite, and for that reason, we propose ε_t to follow a right-truncated Poisson distribution (truncation on the values beyond k, i.e., it exhibits counts from $0, 1, 2, \ldots, k$) with parameter Λ. That is, for our study, we propose*

$$p_x = P(\varepsilon_t = x) = \frac{f(x)I(x \leq k)}{F(k)};\ x = 0, 1, \ldots, k,$$

where $f(x) = f(X = x) = \frac{e^{-\Lambda}\Lambda^x}{x!}$ for $x = 0, 1, 2, \ldots$, i.e., X follows Poisson distribution with parameter Λ, and $F(k) = \sum_{x=0}^{k} f(X = x)$.
We can derive the mean and the variance of the right-truncated Poisson distribution as

$$\mu = \Lambda\frac{F(k-1)}{F(k)},$$

and

$$\sigma^2 = \Lambda^2\frac{F(k-2)}{F(k)} + \Lambda\frac{F(k-1)}{F(k)} - \Lambda^2\left(\frac{F(k-2)}{F(k)}\right)^2.$$

We use the notations μ and σ^2 in our derivations rather than the explicit forms for simplicity. This choice is made for both the proposed mPAR(1) process and the usual PAR(1) process.

Here, we have $P(Y_t = i \mid Y_{t-1}) = \phi W(Y_{t-1} \mid i) + (1-\phi)\pi_i$, and so the conditional expectation and variance are given by

$$E\left(Y_t \mid Y_{t-1}\right) = \phi(\eta - \frac{1-\eta}{k})Y_{t-1} + \frac{(1-\eta)(k+1)}{2}\phi + (1-\phi)\mu, \tag{3.2.4}$$

and $V\left(Y_t \mid Y_{t-1}\right) = E\left({Y_t}^2 \mid Y_{t-1}\right) - E^2\left(Y_t \mid Y_{t-1}\right)$, where

$$E\left({Y_t}^2 \mid Y_{t-1}\right) = \phi(\eta - \frac{1-\eta}{k}){Y_{t-1}}^2 + \frac{(k+1)(2k+1)(1-\eta)}{6}\phi + (1-\phi)(\sigma^2+\mu^2).$$

The derivation of this result is presented in Appendix (see Section 3.7).
The marginal expectation can be derived as

$$\mu_y = E(Y_t) = \frac{\frac{(1-\eta)(k+1)}{2}\phi + (1-\phi)\mu}{1 - \left(\eta - \frac{1-\eta}{k}\right)\phi},$$

by taking expectation on $E\left(Y_t \mid Y_{t-1}\right)$, and the variance can be found as ${\sigma_y}^2 = V(Y_t) = E({Y_t}^2) - E^2(Y_t)$, where

$$E({Y_t}^2) = \frac{\frac{(1-\eta)(k+1)(2k+1)}{6}\phi + (1-\phi)(\mu^2+\sigma^2)}{1 - \left(\eta - \frac{1-\eta}{k}\right)\phi}.$$

Autocorrelation structure

The autocovariance function $\gamma_y(h)(= cov(Y_t, Y_{t+h}) = E(Y_tY_{t+h}) - E(Y_t)E(Y_{t+h}) = E(Y_tY_{t+h}) - {\mu_y}^2)$ of the above process can be found as

$$\gamma_y(h) = \psi^h({\mu_y}^2 + {\sigma_y}^2) + \frac{1-\psi^h}{1-\psi}\left((1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}\right)\mu_y - {\mu_y}^2,$$

where $\psi = \left(\eta - \frac{1-\eta}{k}\right)\phi$. So the function $\gamma_y(h)$ only depends on h.
From the above result, we can write the autocorrealtion function (ACF) as

$$\rho_y(h) = \psi^h,$$

which decays exponentially to 0 as $h \to \infty$. The derivations of these results are presented in Appendix (see Section 3.7).

h-step ahead forecasting distribution

Result 3.2 *The h-step ahead forecasting distribution of our proposed mPAR(1) process, i.e., the probability mass function (p.m.f.) of Y_{t+h} given Y_t is derived as*

$$p_h(i \mid j) = \phi\psi^{h-1}W(j \mid i) + \frac{1-\psi^h}{1-\psi}(1-\phi)\pi_i + \frac{1-\psi^{h-1}}{1-\psi}\frac{1-\eta}{k}\phi,$$

where $p_h(i \mid j) = P(Y_{t+h} = i \mid Y_t = j)$.

Using the result, the forecasting mean can be obtained as

$$E(Y_{t+h} \mid Y_t) = \psi^h Y_t + \frac{1-\psi^h}{1-\psi}\left[(1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}\right],$$

and the conditional variance ($V(Y_{t+h} \mid Y_t) = E({Y_{t+h}}^2 \mid Y_t) - E^2(Y_{t+h} \mid Y_t)$) can be derived from

$$E({Y_{t+h}}^2 \mid Y_t) = \psi^h {Y_t}^2 + \frac{1-\psi^h}{1-\psi}\left[(1-\phi)(\mu^2+\sigma^2) + \frac{(1-\eta)(k+1)(2k+1)}{6}\phi\right].$$

As $h \to \infty$, $E(Y_{t+h} \mid Y_t)$ coverges to μ_y (marginal mean of our proposed process), and $V(Y_{t+h} \mid Y_t)$ coverges to ${\sigma_y}^2$ (marginal variance). This derivation is given in Appendix (see Section 3.7).

Note 3.2 *Here, we mainly consider the rounded mean and mode predictors for both the simulation study and data analysis to investigate different forecasting measures. However, we wish to explore the median predictor too in a future version of this study where a detailed study about only forecasting can be performed.*

3.3 Parameter Estimation

Maximum likelihood estimation (MLE)

In the maximum likelihood estimation method, we maximize the log-likelihood function with respect to the model parameters $\boldsymbol{\beta}^*$ to obtain the maximum likelihood estimate of $\boldsymbol{\beta}^*$; here, $\boldsymbol{\beta}^*$ is vetor-valued, i.e., $\boldsymbol{\beta}^*$ is equal to (ϕ, Λ, η) and (ϕ, Λ) for the mPAR(1) process and the PAR(1) process, respectively. Given a dataset $\{Y_1, \ldots, Y_n\}$, the likelihood function is given by

$$L(\boldsymbol{\beta}^*) = p(Y_1, \ldots, Y_n) = p(Y_1)\prod_{t=2}^{n} p(Y_t \mid Y_{t-1}),$$

where $P(Y_t = i \mid Y_{t-1})$ is equal to $\phi W(Y_{t-1} \mid i) + (1-\phi)\pi_i$ and $\phi I(Y_{t-1} = i) + (1-\phi)\pi_i$ for the mPAR(1) process and the PAR(1) process, respectively. Here, π_i is actually $\pi_i^{\varepsilon,\Lambda}$, i.e., the probability of a right-truncated Poisson distribution with parameter Λ.

We take the log of this function ($l(\boldsymbol{\beta}^*) = \ln L(\boldsymbol{\beta}^*)$) to get the maximum likelihood estimate $\hat{\boldsymbol{\beta}^*}_{mle}$, which is given by

$$\hat{\boldsymbol{\beta}^*}_{mle} = \arg\max_{\boldsymbol{\beta}^*} l(\boldsymbol{\beta}^*).$$

Here, numerical methods are employed to obtain the ML estimates of the model parameters as there are no closed forms of the ML estimators. We use "optim" function and "extraDistr" (for "tpois" functions) package in R software for our calculations.

In the subsequent section, we perform an extensive simulation study to establish the empirical consistency of the estimators of the model parameters.

3.4 Simulation Study

3.4.1 Empirical consistency and AIC comparison

In this section, through some simulation experiments, we check the empirical consistency of the estimated model parameters for our proposed mPAR(1) process through MLE method by studying the average estimated values of the model parameters and their mean squared errors (MSEs) for different sets of the model parameters with varying sample sizes. We carry out this simulation study for (ϕ, Λ, η) as (i) $(0.4, 3, 0.5)$, (ii) $(0.4, 3, 0.6)$, (iii) $(0.4, 3, 0.7)$, (iv) $(0.6, 3, 0.5)$, (v) $(0.6, 3, 0.6)$, and (vi) $(0.6, 3, 0.7)$, with right-truncation occurring beyond 6 and 7. Four sample sizes of 100, 200, 500, 1000 and 5000 are explored with 1000 replications. In this study, the data generating process is the proposed mPAR(1) model, and the estimation process is the MLE method, discussed in Section 3.3. Sample sizes of 100 and 200 are used to observe the small sample properties whereas sample sizes of 5000 are used for large sample properties. Sample sizes of 500 and 1000 are explored for having an idea about moderate sample properties. The results are reported in Tables 3.1 and 3.2. From the results, it can be observed that as sample size increases, MSEs of the estimated model parameters of our proposed mPAR(1) process decrease, and this ensures the empirical consistency of the estimated model parameters.

We also conduct a comparative study between the mPAR(1) process and the PAR(1) process with respect to AIC with the setup same as above. Here, we expect gradual improvements in the performances of the PAR(1) process compared to proposed mPAR(1) process for higher values of η as the mPAR(1) process approaches more towards to the PAR(1) process for higher values of η. The results

are given in Tables 3.3 and 3.4 for sample sizes 200, 500, 1000 and 5000. From the tables, it is clearly seen that the proposed mPAR(1) process performs better than the PAR(1) process in terms of their average AICs. However, as the values of η increase, we observe that the performances of the PAR(1) process also improve gradually, which is expected as we go towards the usual PAR(1) process with increasing value of η. Overall, our proposed mPAR(1) has performed better than the PAR(1) process in the simulation study for model comparison.

TABLE 3.1: Average estimated values of parameters along with the MSEs for right-truncation beyond 6

(ϕ, Λ, η)	(0.4,3,0.5)			(0.4,3,0.6)			(0.4,3,0.7)		
n	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)
100	0.4041	2.9849	0.5471	0.3935	3.0009	0.6574	0.4084	2.9862	0.7414
	(0.0257)	(0.1440)	(0.0428)	(0.0231)	(0.1275)	(0.0430)	(0.0203)	(0.1340)	(0.0352)
200	0.3928	2.9865	0.5360	0.3987	3.0017	0.6321	0.3972	3.0105	0.7335
	(0.0150)	(0.0660)	(0.0232)	(0.0131)	(0.0586)	(0.0234)	(0.0109)	(0.0561)	(0.0234)
500	0.3959	3.0036	0.5160	0.3989	3.0022	0.6171	0.3962	2.9988	0.7209
	(0.0065)	(0.0240)	(0.0078)	(0.0061)	(0.0231)	(0.0106)	(0.0056)	(0.0226)	(0.0124)
1000	0.4010	2.9993	0.5048	0.4000	3.0020	0.6057	0.3981	2.9972	0.7105
	(0.0034)	(0.0118)	(0.0033)	(0.0030)	(0.0107)	(0.0043)	(0.0029)	(0.0105)	(0.0059)
5000	0.4004	3.0000	0.5001	0.3980	2.9989	0.6027	0.4001	2.9977	0.6999
	(0.0006)	(0.0025)	(0.0006)	(0.0006)	(0.0021)	(0.0008)	(0.0006)	(0.0020)	(0.0010)
(ϕ, Λ, η)	(0.6,3,0.5)			(0.6,3,0.6)			(0.6,3,0.7)		
n	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)
100	0.5778	3.0782	0.5484	0.5698	2.9970	0.6502	0.5882	2.9976	0.7387
	(0.0301)	(0.6048)	(0.0266)	(0.0252)	(0.4295)	(0.0267)	(0.0209)	(0.3656)	(0.0224)
200	0.5835	3.0112	0.5241	0.5905	2.9782	0.6231	0.5887	3.0119	0.7301
	(0.0153)	(0.1461)	(0.0103)	(0.0142)	(0.1385)	(0.0121)	(0.0128)	(0.1162)	(0.0138)
500	0.6007	3.0045	0.5043	0.5999	2.9976	0.6063	0.5977	2.9948	0.7090
	(0.0063)	(0.0520)	(0.0030)	(0.0057)	(0.0448)	(0.0039)	(0.0051)	(0.0420)	(0.0052)
1000	0.5980	2.9978	0.5025	0.5991	3.0041	0.6046	0.6000	2.9958	0.7040
	(0.0031)	(0.0263)	(0.0015)	(0.0030)	(0.0244)	(0.0019)	(0.0025)	(0.0203)	(0.0022)
5000	0.6003	2.9979	0.5005	0.5989	3.0020	0.6015	0.5997	2.9993	0.7003
	(0.0007)	(0.0049)	(0.0003)	(0.0006)	(0.0046)	(0.0004)	(0.0005)	(0.0040)	(0.0004)

TABLE 3.2: Average estimated values of parameters along with the MSEs for right-truncation beyond 7

(ϕ, Λ, η)	(0.4,3,0.5)			(0.4,3,0.6)			(0.4,3,0.7)		
n	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)
100	0.3812	2.9926	0.5544	0.3925	2.9692	0.6456	0.3934	2.9977	0.7391
	(0.0194)	(0.1208)	(0.0386)	(0.0174)	(0.1150)	(0.0335)	(0.0157)	(0.0992)	(0.0308)
200	0.3958	2.9956	0.5251	0.3933	2.9940	0.6319	0.3989	2.9987	0.7192
	(0.0107)	(0.0532)	(0.0179)	(0.0099)	(0.0497)	(0.0187)	(0.0091)	(0.0487)	(0.0182)
500	0.3971	3.0099	0.5057	0.3961	2.9989	0.6135	0.3984	3.0020	0.7115
	(0.0041)	(0.0222)	(0.0052)	(0.0045)	(0.0206)	(0.0076)	(0.0035)	(0.0187)	(0.0073)
1000	0.3976	3.0031	0.5041	0.3972	3.0033	0.6071	0.3984	3.0010	0.7072
	(0.0023)	(0.0112)	(0.0025)	(0.0021)	(0.0111)	(0.0032)	(0.0017)	(0.0101)	(0.0035)
5000	0.4001	2.9985	0.5004	0.4001	2.9998	0.6003	0.4002	3.0005	0.6991
	(0.0005)	(0.0022)	(0.0005)	(0.0004)	(0.0021)	(0.0005)	(0.0004)	(0.0019)	(0.0007)
(ϕ, Λ, η)	(0.6,3,0.5)			(0.6,3,0.6)			(0.6,3,0.7)		
n	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)	$\hat{\phi}$ (MSE)	$\hat{\Lambda}$ (MSE)	$\hat{\eta}$ (MSE)
100	0.5799	2.9890	0.5387	0.5797	3.0066	0.6338	0.5864	2.9754	0.7266
	(0.0221)	(0.3008)	(0.0198)	(0.0179)	(0.2764)	(0.0190)	(0.0153)	(0.2382)	(0.0166)
200	0.5856	2.9955	0.5163	0.5903	3.0046	0.6167	0.5914	2.9879	0.7195
	(0.0104)	(0.1220)	(0.0068)	(0.0105)	(0.1108)	(0.0090)	(0.0087)	(0.0964)	(0.0095)
500	0.5974	3.0055	0.5063	0.5958	3.0011	0.6068	0.5962	2.9994	0.7070
	(0.0042)	(0.0516)	(0.0023)	(0.0038)	(0.0466)	(0.0030)	(0.0036)	(0.0392)	(0.0034)
1000	0.5968	3.0004	0.5029	0.5975	3.0076	0.6034	0.5973	2.9988	0.7045
	(0.0023)	(0.0243)	(0.0012)	(0.0020)	(0.0210)	(0.0014)	(0.0017)	(0.0197)	(0.0017)
5000	0.6001	3.0006	0.5007	0.5994	3.0003	0.6012	0.5990	3.0004	0.7006
	(0.0004)	(0.0049)	(0.0002)	(0.0004)	(0.0045)	(0.0003)	(0.0003)	(0.0040)	(0.0003)

TABLE 3.3: Comparative study between the mPAR(1) and the PAR(1) methods with respect to AIC where the data generating process is the proposed mPAR(1) process for right-truncation beyond 6

(ϕ, Λ, η)	(0.4,3,0.5)		(0.4,3,0.6)		(0.4,3,0.7)	
n	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)
200	729.1011	732.3282	711.2206	713.1494	689.3871	689.9925
500	1817.0913	1826.7577	1771.6088	1778.0936	1718.3784	1721.7675
1000	3632.5535	3653.6644	3541.6496	3555.7290	3433.0129	3440.9254
5000	18146.3931	18254.8448	17688.6621	17760.9847	17161.0207	17205.6125
(ϕ, Λ, η)	(0.6,3,0.5)		(0.6,3,0.6)		(0.6,3,0.7)	
n	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)
200	710.9769	719.5673	677.1540	683.3068	635.4105	638.6975
500	1774.4587	1799.1599	1688.8548	1706.1375	1586.8335	1597.2912
1000	3545.6446	3595.1204	3372.3289	3407.3078	3169.3065	3191.3255
5000	17707.0785	17959.1824	16851.8451	17029.8821	15846.1955	15960.4827

TABLE 3.4: Comparative study between the mPAR(1) and the PAR(1) methods with respect to AIC where the data generating process is the proposed mPAR(1) process for right-truncation beyond 7

(ϕ, Λ, η)	(0.4,3,0.5)		(0.4,3,0.6)		(0.4,3,0.7)	
n	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)
200	764.4629	769.9011	742.7114	745.9555	721.0277	722.8466
500	1908.4880	1923.5491	1853.7471	1863.8318	1792.8522	1798.6552
1000	3810.9611	3842.0297	3704.4441	3725.5283	3580.5978	3593.2003
5000	19035.5016	19196.6695	18511.7574	18623.4227	17909.5281	17978.9751
(ϕ, Λ, η)	(0.6,3,0.5)		(0.6,3,0.6)		(0.6,3,0.7)	
n	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)
200	748.3859	761.1242	711.0079	720.1849	665.3266	670.7413
500	1865.0132	1899.5166	1770.9864	1795.5491	1660.1418	1675.6964
1000	3729.5635	3799.5460	3539.4389	3589.8105	3314.3373	3346.5559
5000	18627.7919	18984.2905	17676.4865	17933.4943	16568.3237	16735.4297

3.4.2 Forecasting

This simulation experiment is performed to study the h-step ahead forecasting performances for varying h of the proposed mPAR(1) process compared to the

PAR(1) process. For comparison, we consider four measures of forecasting criteria, namely (i) the h-step ahead predicted root mean squared error (PRMSE(h)) (based on rounded off mean predictors), (ii) PTP[mean](h), (iii) PTP[mode](h), and (iv) the h-step ahead PTP within an interval for mode predictor (PTPI[mode](h)) (where $\delta = 1$). The mathematical forms of these measures are also provided in Appendix 3.7. In order to perform this study, we simulate datasets from the proposed mPAR(1) process for $(\phi, \Lambda, \eta) = (0.4, 1.5, 0.5)$, $(0.4, 1.5, 0.6)$, $(0.4, 1.5, 0.7)$, $(0.5, 1.5, 0.5)$, $(0.5, 1.5, 0.6)$, $(0.5, 1.5, 0.7)$, $(0.6, 1.5, 0.5)$, $(0.6, 1.5, 0.6)$, and $(0.6, 1.5, 0.7)$, with right-truncation occurring beyond 6 and 7. That is, we take these sets of parameters every time for each truncation. Each time, we generate a data of sample size 500 of which a training set of size 400 is used to fit the models considered for this study and a test set of size 100 is considered to find the forecasting measures for $h = 1, 2, 3, 4$. This procedure is repeated for 1000 times. The mean (by rounding off to the next integer) and mode are considered for prediction throughout this study.

We report the h-step ahead forecasting performances for two models for $h = 1, 2, 3, 4$ in Tables 3.5 and 3.6. From the tables, it is observed that our proposed process mostly performs better than the PAR(1) process. In terms of PTP[mode](h) and PTPI[mode](h) measures, our proposed method largely outperforms the PAR process. However, If we look at that the performances of the PAR(1) process compared to the mPAR(1) process with respect to four measures, we can say that the performance of PAR(1) process more or less improves gradually with increased values of η, as seen in the simulation study regarding model comparison with respect to AIC. Overall, we can state that our proposed mPAR(1) method generally performs better compared to the PAR(1) method in the simulation study for forecasting.

TABLE 3.5: Different forecasting measures for varying h where the data generating process is the proposed mPAR(1) process for different sets of (ϕ, Λ, η) with right-truncation beyond 6

	mPAR(1)				PAR(1)			
h	1	2	3	4	1	2	3	4
Measures	(ϕ, Λ, η)=(0.4,1.5,0.5)							
PRMSE(h)	1.6204	1.6183	1.6184	1.6182	1.6197	1.6183	1.6184	1.6182
PTP[mean](h)	22.02	22.21	22.23	22.23	22.00	22.21	22.23	22.23
PTP[mode](h)	32.92	27.87	27.94	27.93	32.72	26.41	26.18	26.12
PTPI[mode](h)	63.17	70.06	70.22	70.22	63.10	67.76	68.16	68.18
	(ϕ, Λ, η)=(0.4,1.5,0.6)							
PRMSE(h)	1.5687	1.5787	1.5782	1.5780	1.5694	1.5787	1.5782	1.5780
PTP[mean](h)	23.01	22.69	22.71	22.71	22.90	22.69	22.71	22.71
PTP[mode](h)	37.24	28.85	28.95	28.97	37.17	28.41	28.15	28.23
PTPI[mode](h)	64.48	71.42	71.94	71.95	64.73	70.04	70.94	71.10
	(ϕ, Λ, η)=(0.4,1.5,0.7)							
PRMSE(h)	1.4796	1.5190	1.5177	1.5171	1.4797	1.5189	1.5175	1.5171
PTP[mean](h)	25.03	23.16	23.16	23.19	24.81	23.16	23.15	23.19
PTP[mode](h)	41.75	30.07	30.08	30.09	41.75	30.11	29.88	29.99
PTPI[mode](h)	67.58	72.12	74.08	74.12	67.64	72.27	73.59	73.99
	(ϕ, Λ, η)=(0.5,1.5,0.5)							
PRMSE(h)	1.6922	1.7065	1.7068	1.7069	1.6922	1.7066	1.7068	1.7069
PTP[mean](h)	20.36	21.08	21.09	21.11	20.30	21.08	21.09	21.11
PTP[mode](h)	35.06	26.29	26.49	26.48	34.87	23.94	22.50	22.18
PTPI[mode](h)	59.84	65.97	66.96	66.97	60.20	62.81	62.21	61.89
	(ϕ, Λ, η)=(0.5,1.5,0.6)							
PRMSE(h)	1.6282	1.6575	1.6567	1.6572	1.6282	1.6577	1.6567	1.6572
PTP[mean](h)	21.24	21.47	21.47	21.45	21.20	21.47	21.47	21.45
PTP[mode](h)	40.56	27.67	27.50	27.52	40.54	26.16	24.97	24.33
PTPI[mode](h)	63.80	65.41	68.58	68.59	63.82	64.72	65.31	65.05
	(ϕ, Λ, η)=(0.5,1.5,0.7)							
PRMSE(h)	1.5380	1.5970	1.5956	1.5960	1.5374	1.5969	1.5956	1.5960
PTP[mean](h)	23.35	22.31	22.31	22.31	23.05	22.27	22.31	22.31
PTP[mode](h)	45.70	30.11	28.43	28.53	45.70	29.10	27.30	27.02
PTPI[mode](h)	68.08	65.54	70.57	70.79	68.08	66.74	68.75	69.27
	(ϕ, Λ, η)=(0.6,1.5,0.5)							
PRMSE(h)	1.7524	1.7996	1.7977	1.7975	1.7524	1.7995	1.7977	1.7976
PTP[mean](h)	19.31	19.83	20.05	20.06	19.36	19.76	20.04	20.05
PTP[mode](h)	37.90	25.09	24.72	24.78	37.85	22.66	20.31	20.19
PTPI[mode](h)	59.93	60.11	63.19	63.27	59.96	58.20	58.20	58.09
	(ϕ, Λ, η)=(0.6,1.5,0.6)							
PRMSE(h)	1.6860	1.7508	1.7502	1.7504	1.6866	1.7510	1.7504	1.7504
PTP[mean](h)	20.48	20.38	20.79	20.78	20.59	20.35	20.76	20.78
PTP[mode](h)	44.09	27.72	25.51	25.55	44.09	25.47	22.62	21.44
PTPI[mode](h)	64.69	60.21	64.61	65.22	64.69	59.41	60.94	60.36
	(ϕ, Λ, η)=(0.6,1.5,0.7)							
PRMSE(h)	1.5860	1.6877	1.6895	1.6889	1.5859	1.6879	1.6895	1.6889
PTP[mean](h)	22.09	21.20	21.66	21.70	22.01	21.10	21.66	21.70
PTP[mode](h)	50.38	31.13	26.91	26.76	50.38	30.47	25.18	24.17
PTPI[mode](h)	69.36	60.91	65.54	67.46	69.36	60.77	63.68	63.83

TABLE 3.6: Different forecasting measures for varying h where the data generating process is the proposed mPAR(1) process for different sets of (ϕ, Λ, η) with right-truncation beyond 7

	mPAR(1)				PAR(1)			
h	1	2	3	4	1	2	3	4
Measures	(ϕ, Λ, η)=(0.4,1.5,0.5)							
PRMSE(h)	1.8118	1.8226	1.8227	1.8222	1.8120	1.8227	1.8227	1.8222
PTP[mean](h)	20.72	21.47	21.48	21.48	20.65	21.47	21.48	21.48
PTP[mode](h)	32.98	27.57	27.59	27.60	32.44	24.43	22.90	22.81
PTPI[mode](h)	62.02	68.98	69.07	69.07	61.05	64.24	63.24	63.11
	(ϕ, Λ, η)=(0.4,1.5,0.6)							
PRMSE(h)	1.7318	1.7445	1.7442	1.7435	1.7322	1.7445	1.7442	1.7435
PTP[mean](h)	21.63	22.25	22.24	22.24	21.56	22.25	22.24	22.24
PTP[mode](h)	36.85	28.62	28.67	28.68	36.85	27.15	26.21	26.04
PTPI[mode](h)	63.12	70.51	71.05	71.07	62.99	67.57	67.69	67.57
	(ϕ, Λ, η)=(0.4,1.5,0.7)							
PRMSE(h)	1.6296	1.6640	1.6635	1.6634	1.6299	1.6640	1.6635	1.6634
PTP[mean](h)	23.46	22.96	22.97	22.97	23.34	22.96	22.97	22.97
PTP[mode](h)	41.16	29.53	29.43	29.43	41.15	29.15	28.19	28.29
PTPI[mode](h)	66.18	70.85	72.75	72.74	66.18	70.23	71.12	71.44
	(ϕ, Λ, η)=(0.5,1.5,0.5)							
PRMSE(h)	1.9117	1.9539	1.9509	1.9505	1.9125	1.9548	1.9514	1.9511
PTP[mean](h)	19.28	19.96	20.18	20.20	19.34	19.81	20.11	20.16
PTP[mode](h)	34.94	25.89	26.07	26.09	34.75	22.42	20.38	20.31
PTPI[mode](h)	57.93	63.93	64.99	65.02	58.12	59.72	59.08	59.00
	(ϕ, Λ, η)=(0.5,1.5,0.6)							
PRMSE(h)	1.8314	1.8806	1.8805	1.8798	1.8312	1.8818	1.8805	1.8798
PTP[mean](h)	20.17	20.72	20.95	20.96	20.30	20.65	20.94	20.95
PTP[mode](h)	40.52	27.60	27.03	27.05	40.52	24.89	22.47	21.68
PTPI[mode](h)	62.52	64.17	67.33	67.37	62.53	61.17	61.82	61.26
	(ϕ, Λ, η)=(0.5,1.5,0.7)							
PRMSE(h)	1.7144	1.7875	1.7871	1.7876	1.7150	1.7883	1.7870	1.7876
PTP[mean](h)	21.58	21.42	21.77	21.77	21.54	21.35	21.77	21.77
PTP[mode](h)	45.57	29.91	27.96	28.02	45.57	28.15	25.85	24.70
PTPI[mode](h)	66.83	64.63	69.71	69.91	66.83	63.25	65.83	65.58
	(ϕ, Λ, η)=(0.6,1.5,0.5)							
PRMSE(h)	2.0036	2.0892	2.1029	2.1045	2.0043	2.0910	2.1043	2.1062
PTP[mean](h)	19.58	17.27	17.44	17.35	19.68	16.94	16.88	16.86
PTP[mode](h)	37.43	24.34	23.65	23.71	37.44	21.72	19.36	19.38
PTPI[mode](h)	57.65	57.56	60.94	61.04	57.67	54.65	55.27	55.31
	(ϕ, Λ, η)=(0.6,1.5,0.6)							
PRMSE(h)	1.9157	2.0091	2.0243	2.0259	1.9149	2.0096	2.0278	2.0294
PTP[mean](h)	20.88	18.67	19.08	19.20	21.15	18.51	18.81	18.91
PTP[mode](h)	43.65	26.86	24.77	24.85	43.65	24.11	20.57	19.79
PTPI[mode](h)	62.76	58.20	62.51	63.01	62.76	56.31	57.77	57.37
	(ϕ, Λ, η)=(0.6,1.5,0.7)							
PRMSE(h)	1.7814	1.8989	1.9163	1.9154	1.7813	1.8997	1.9183	1.9161
PTP[mean](h)	22.13	19.87	20.48	20.65	22.53	19.80	20.36	20.60
PTP[mode](h)	49.94	30.67	26.35	26.35	49.94	29.64	23.68	21.78
PTPI[mode](h)	67.85	59.50	63.81	65.87	67.85	58.73	60.65	60.89

3.5 Data Analysis

For the practical illustration of our proposed method, we consider the real dataset of monthly number of U.S. cases of poliomyelitis for 1970 to 1983, which is used in Zeger, 1988. This data consists of 168 observations (see Zeger, 1988). This dataset contains four extreme observations in the context of our study. We replace those four extreme values: 7 (May, 1979), 8 (June, 1979), 9 (July, 1970) and 14 (November, 1972), which are more than 6, with the next highest value of this dataset, i.e., with 6, to carry out this data analysis. Therefore, the value of k, determined by $max(Y_1, \ldots, Y_{168})$, is 6. The reason behind taking the value of k as 6 is that this dataset is a zero-inflated data, and our kernel is not appropriate for count time series with higher values, especially for a zero-inflated data. Figure 3.1 displays the data plot along with its autocorrelation function (ACF) and Partial ACF (PACF) plots. From the plots, we can say that the AR(1) process is a good fit for this dataset.

After fitting our proposed mPAR(1) process (estimated model parameters: $(\hat{\phi}, \hat{\Lambda}, \hat{\eta}) = (0.2065, 0.8246, 0.2539)$ and the PAR(1) process (estimated model parameters: $(\hat{\phi}, \hat{\Lambda}) = (0.0444, 1.2093)$), we find out that our proposed mPAR(1) method produces the AIC value of 509.9682, whereas the AIC value for the PAR(1) is obtained as 538.5404. So this implies our proposed mPAR(1) process performs better than the PAR(1) process.

To examine the predictive performances of these two processes, we divide this dataset into two parts − a training set of 156 observations to fit the models and a test set of 12 observations to find all three forecasting measures. It is observed from the obtained values of the forecasting measures, no difference between two models for the PTP[mean] measures (based on rounded off mean predictors) can be found, but the PTP[mode] values of our proposed method are mostly higher than those of the PAR process. For $h = 1, 2, 3$, we have PTP[mode] values for the proposed mPAR(1) process and the PAR(1) process as (41.67, 36.36, 40.00) and (33.33, 36.36, 30.00), respectively. And for the PRMSE measure (based on expected mean values), the PAR(1) process has lower value than the mPAR(1) process for $h = 1$, but for $h = 2, 3$, the mPAR(1) process performs better than the PAR(1) process. These are displayed in Table 3.7. Therefore, we can say that our proposed mPAR(1) process provides better performances than the PAR(1) process for forecasting. Overall, our proposed mPAR(1) method gives better performance compared to the PAR(1) process in this data analysis.

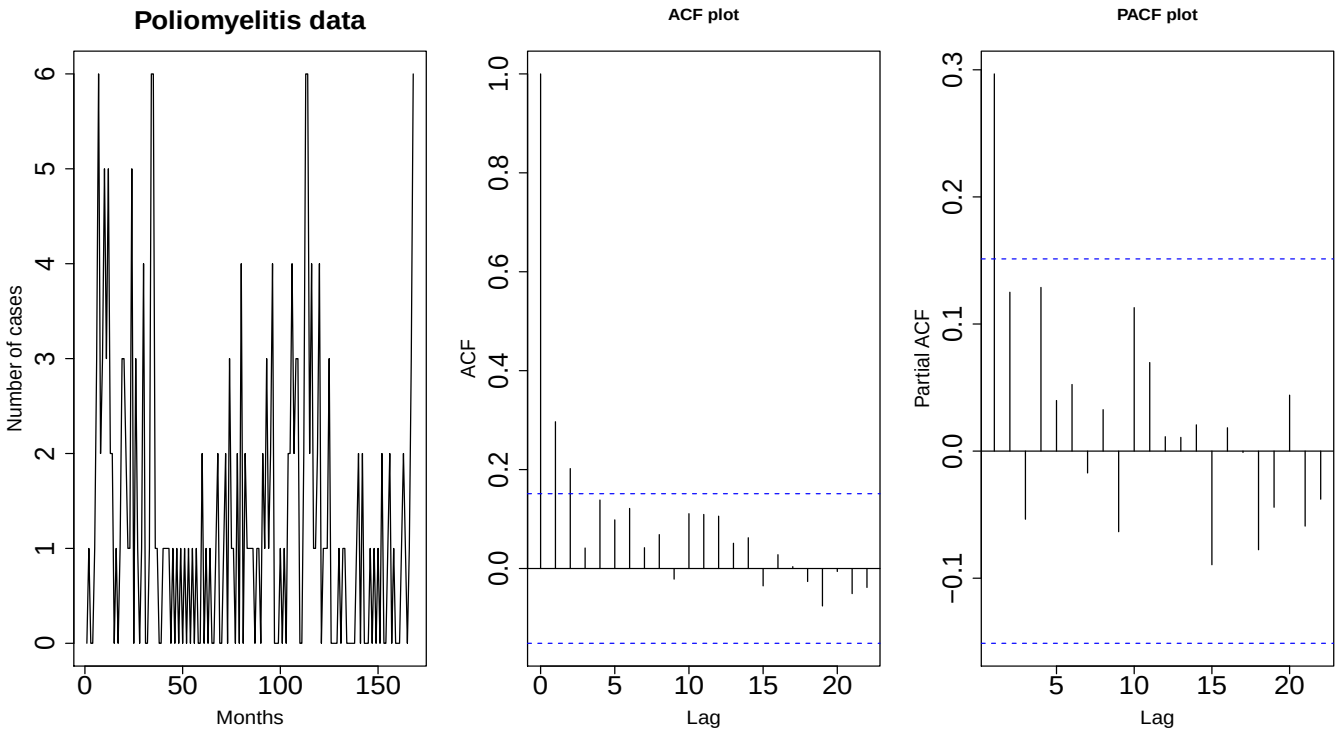


FIGURE 3.1: Monthly number of U.S. cases of poliomyelitis for 1970 to 1983

TABLE 3.7: Forecasting results for the data

Measures	PRMSE(h)		PTP[mean](h)		PTP[mode](h)	
h	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)	mPAR(1)	PAR(1)
1	1.6714	1.6636	33.33	33.33	41.67	33.33
2	1.7257	1.7278	36.36	36.36	36.36	36.36
3	1.8084	1.8107	30.00	30.00	40.00	30.00

3.6 Conclusions

In this chapter, we have proposed a modified PAR(1) (mPAR(1)) process for modelling truncated count data. The modification has been employed to tackle the drawback, which exists for the PAR process regarding the use of indicator kernel. In the context of count time series data, the indicator kernel only gives significance to $i = j$ cases but no weight to $i \neq j$ cases. In our proposed mPAR process, we have considered a modified kernel which gives some weights to $i \neq j$ cases too. We have discussed the mathematical properties like the distributional features and the h-step ahead forecasting distribution of the proposed mPAR process in this chapter.

Extensive simulation studies have been performed to show the empirical consistency of the estimated model parameters through the MLE method. Model comparison between our proposed mPAR(1) process and the PAR(1) process has also been examined through some simulation experiments. Some detailed simulation studies have been executed to make a comparison of the predictive performances of two methods with respect to some forecasting measures. Our proposed mPAR(1) process has mostly performed better than the PAR(1) process in those studies. In the data example of the monthly number of U.S. cases of poliomyelitis for 1970 to 1983, it has also been found that the proposed mPAR process has given better performance than the PAR process. Overall, we anticipate that the proposed mPAR process will serve as a fruitful method in addressing count time series data in the future.

3.7 Appendix

Appendix A. Proofs regarding the marginal distribution of Y_t of the mPAR(p) process

$$\begin{aligned} P(Y_t = i) &= \sum_{i_1} \cdots \sum_{i_p} P_{Y_t, Y_{t-1}, \ldots, Y_{t-p}} \left(Y_t = i, Y_{t-1} = i_1, \ldots, Y_{t-p} = i_p\right) \\ &= \sum_{i_1} \cdots \sum_{i_p} P_{Y_{t-1}, \ldots, Y_{t-p}} \left(i_1, \ldots, i_p\right) P\left(Y_t = i \mid i_1, \ldots, i_p\right). \end{aligned}$$

Let us define

$$\begin{aligned} p_i &= P(Y_t = i) \\ &= \sum_{i_1} \cdots \sum_{i_p} \left[\phi_1 W\left(Y_{t-1} = i_1 \mid i\right) + \ldots + \phi_p W\left(Y_{t-p} = i_p \mid i\right)\right] P_{Y_{t-1}, \ldots, Y_{t-p}}\left(i_1, \ldots, i_p\right) \\ &\qquad + \left(1 - \sum_{i=1}^{p} \phi_i\right) \pi_i. \end{aligned}$$

Now,

$$\begin{aligned}
&\sum_{i_1}\cdots\sum_{i_p} W\left(Y_{t-1}=i_1 \mid i\right) P_{Y_{t-1},\ldots,Y_{t-p}}\left(i_1,\ldots,i_p\right) \\
&\qquad = \sum_{i_1} W\left(Y_{t-1}=i_1 \mid i\right) P_{Y_{t-1}}\left(i_1\right) \\
&\qquad = \eta P\left(Y_{t-1}=i\right) + \frac{1-\eta}{k}\left(\sum_{i_1} P\left(Y_{t-1}=i_1\right) - P\left(Y_{t-1}=i\right)\right) \\
&\qquad = \eta p_i + \frac{1-\eta}{k}\left(S-p_i\right), \text{ where } S=\sum_i P\left(Y_j=i\right) \\
&\qquad = S\frac{1-\eta}{k} + \left(\eta - \frac{1-\eta}{k}\right) p_i.
\end{aligned}$$

We have

$$\begin{aligned}
p_i &= P(Y_t=i) \\
&= S\frac{1-\eta}{k}\sum_{i=1}^{p}\phi_i + \left(\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i\right) p_i + \left(1-\sum_{i=1}^{p}\phi_i\right)\pi_i.
\end{aligned}$$

Therefore,

$$p_i = \frac{S\frac{1-\eta}{k}\sum_{i=1}^{p}\phi_i}{1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i} + \frac{1-\sum_{i=1}^{p}\phi_i}{1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i}\pi_i.$$

So, we can write

$$\begin{aligned}
S &= \sum_i P(Y_t = i) \\
&\Rightarrow S = \frac{\sum_{i=0}^{k}\left(S\frac{1-\eta}{k}\sum_{j=1}^{p}\phi_j\right) + \left(1-\sum_{j=1}^{p}\phi_j\right)\sum_{i=0}^{k}\pi_i}{1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{j=1}^{p}\phi_j} \\
&\Rightarrow S - S\left[\left(\eta-\frac{1-\eta}{k}\right)\sum_{j=1}^{p}\phi_j\right] \\
&\qquad = S\frac{k+1}{k}(1-\eta)\sum_{j=1}^{p}\phi_j + \left(1-\sum_{j=1}^{p}\phi_j\right) \\
&\Rightarrow S\left[1-\left(\frac{(k+1)\eta-1+k+1-(k+1)\eta}{k}\right)1-\sum_{j=1}^{p}\phi_j\right] \\
&\qquad = 1-\sum_{j=1}^{p}\phi_j \\
&\Rightarrow S\left[1-\sum_{j=1}^{p}\phi_j\right] = 1-\sum_{j=1}^{p}\phi_j \\
&\Rightarrow S = 1, \qquad \text{since} \sum_{j=1}^{p}\phi_j \in (0,1).
\end{aligned}$$

Now, we are to show that $P(Y_t = i) > 0$. We know that $p_i = P(Y_t = i) = \frac{\pi_i + \left(\frac{1-\eta}{k} - \pi_i\right)\sum_{i=1}^{p}\phi_i}{1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i}$, where $\sum_{i=1}^{p}\phi_i \in (0,1)$.

Since $\sum_{i=1}^{p}\phi_i \in (0,1)$, we can say that $\pi_i + \left(\frac{1-\eta}{k} - \pi_i\right)\sum_{i=1}^{p}\phi_i$ takes value between π_i and $\frac{1-\eta}{k}$, and both are greater than 0. Again, $1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i$ takes value between 1 and $(1-\eta)+\frac{1-\eta}{k} = \frac{k+1}{k}(1-\eta)$, and both are greater than 0 too. So, $P(Y_t = i) > 0$.

So, we have $P(Y_t = i) = \frac{\frac{1-\eta}{k}\sum_{i=1}^{p}\phi_i}{1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i} + \frac{1-\sum_{i=1}^{p}\phi_i}{1-\left(\eta-\frac{1-\eta}{k}\right)\sum_{i=1}^{p}\phi_i}\pi_i$, and $P(Y_t = i) > 0$, and $\sum_{i=0}^{k}P(Y_t = i) = 1$.

Appendix B. Proof of the conditional expectation

The conditional expectation of the process is given by

$$\begin{aligned} E\left(Y_t \mid Y_{t-1}, \ldots, Y_{t-p}\right) &= \sum_{i=0}^{k} P\left(Y_t = i \mid Y_{t-1}, \ldots, Y_{t-p}\right) \\ &= \sum_{i=0}^{k} i\left[\phi_1 W\left(Y_{t-1} \mid i\right) + \ldots + \phi_p W\left(Y_{t-p} \mid i\right) + \left(1 - \sum_{i=1}^{p} \phi_i\right) \pi_i\right] \end{aligned}$$

Given Y_{t-j} (where $j = 1, \ldots, p$), we can write

$$\begin{aligned} \sum_{i=0}^{k} iW\left(Y_{t-j} \mid i\right) &= \eta Y_{t-j} + \sum_{\substack{i=o \\ (\neq Y_{t-j})}}^{k} i\frac{1-\eta}{k} \\ &= \eta Y_{t-j} + \frac{1-\eta}{k}\left[\sum_{i=0}^{k} i - Y_{t-j}\right] \\ &= \eta Y_{t-j} + \frac{1-\eta}{k}\frac{k(k+1)}{2} - \frac{1-\eta}{k} Y_{t-j} \\ &= \eta Y_{t-j} + \frac{(1-\eta)(k+1)}{2} - \frac{1-\eta}{k} Y_{t-j} \end{aligned}$$

So, we can write

$$\begin{aligned} E(Y_t \mid Y_{t-1}, \ldots, Y_{t-p}) = &\left(\eta - \frac{1-\eta}{k}\right)\left(\phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p}\right) + \frac{(k+1)(1-\eta)}{2}\sum_{i=1}^{p}\phi_i + \\ &\left(1 - \sum_{i=1}^{p}\phi_i\right) E(\varepsilon_t). \end{aligned}$$

For $\eta = 1$, $P\left(Y_t = i\right)$ is equal to π_i, i.e., the marginal distribution of Y_t is $D\left(\pi_0, \pi_1, \ldots, \pi_k\right)$, and equation (3.7.1) becomes $E(Y_t \mid Y_{t-1}, \ldots, Y_{t-p}) = \left(\phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p}\right) + \left(1 - \sum_{i=1}^{p}\phi_i\right) E(\varepsilon_t)$. These two results are true for the indicator kernel which is the case for $\eta = 1$ in our study.

Appendix C. Mean and variance of right-truncated Poisson distribution beyond k

Here, the mean (μ) can be derived in the following way:

$$\begin{aligned}
\mu &= E(\varepsilon_t) \\
&= E(X \mid X \leq k) \\
&= \frac{1}{F(k)} \sum_{x=0}^{k} x \frac{e^{-\Lambda}\Lambda^x}{x!}, \text{ where } F(k) = \textstyle\sum_{x=0}^{k} f(x) \\
&= \frac{\Lambda}{F(k)} \sum_{z=0}^{k-1} \frac{e^{-\Lambda}\Lambda^z}{z!}, \text{ where } z = x-1 \\
&= \Lambda \frac{F(k-1)}{F(k)}.
\end{aligned}$$

Now, we can write

$$\begin{aligned}
E(X(X-1) \mid X \leq k) &= \frac{1}{F(k)} \sum_{x=0}^{k} x(x-1) \frac{e^{-\Lambda}\Lambda^x}{x!} \\
&= \frac{\Lambda^2}{F(k)} \sum_{z=0}^{k-2} \frac{e^{-\Lambda}\Lambda^z}{z!}, \text{ where } z = x-2 \\
&= \Lambda^2 \frac{F(k-2)}{F(k)}.
\end{aligned}$$

So, the variance (σ^2) can be expressed as

$$\begin{aligned}
&V(X \mid X \leq k) \\
&\quad = E(X(X-1) \mid X \leq k) + E(X \mid X \leq k) - E(X \mid X \leq k)^2 \\
&\quad = \Lambda^2 \frac{F(k-2)}{F(k)} + \Lambda \frac{F(k-1)}{F(k)} - \Lambda^2 \left(\frac{F(k-2)}{F(k)} \right)^2.
\end{aligned}$$

Appendix D. Conditional expectation and variance of the mPAR(1) process

Here, we can find the form of $E(Y_t \mid Y_{t-1})$ (mentioned in equation (3.2.4)) from the result that shows $\sum_{i=0}^{k} iW(Y_{t-1} \mid i) = \left(\eta - \frac{1-\eta}{k}\right) Y_{t-1} + \frac{(1-\eta)(k+1)}{2}$ (also

shown in Section 3.7), and the form of $E\left({Y_t}^2 \mid Y_{t-1}\right)$ can be found from the following result

$$\begin{aligned}\sum_{i=0}^{k} i^2 W\left(Y_{t-1} \mid i\right) &= \eta {Y_{t-1}}^2 + \sum_{\substack{i=o \\ (\neq Y_{t-j})}}^{k} i^2 \frac{1-\eta}{k} \\ &= \eta {Y_{t-1}}^2 + \frac{(k+1)(2k+1)(1-\eta)}{6} - \frac{1-\eta}{k} {Y_{t-1}}^2 \\ &= \left(\eta - \frac{1-\eta}{k}\right) {Y_{t-1}}^2 + \frac{(k+1)(2k+1)(1-\eta)}{6}.\end{aligned}$$

Appendix E. h-step ahead forecasting distribution, i.e. the distribution of Y_{n+h} given Y_n

Here, let us write the h-step ahead forecasting distribution as

$$p_h\left(i \mid i_1\right) = P\left(Y_{t+h} = i \mid Y_t = i_1\right).$$

Now, for $h = 1$ (1-step ahead), we have

$$p_1\left(i \mid i_1\right) = P\left(Y_{t+1} = i \mid Y_t = i_1\right) = \phi W\left(i_1 \mid i\right) + (1-\phi)\pi_i.$$

For $h = 2$ (2-step ahead), we can write

$$\begin{aligned}p_2\left(i \mid i_1\right) &= P\left(Y_{t+2} = i \mid Y_t = i_1\right) \\ &= \sum_{j=0}^{k} P\left(Y_{t+2} = i \mid Y_{t+1} = j\right) P\left(Y_{t+1} = j \mid Y_t = i_1\right) \\ &= \sum_{j=0}^{k} \left[\phi W(j \mid i) + (1-\phi)\pi_i\right] \left[\phi W(i_1 \mid j) + (1-\phi)\pi_i\right] \\ &= \sum_{j=0}^{k} \left[\phi^2 W(j \mid i) W(i_1 \mid j) + \phi(1-\phi)\pi_i W(i_1 \mid j)\right] + \\ &\qquad \sum_{j=0}^{k} \left[\phi(1-\phi)\pi_j W(j \mid i) + (1-\phi)^2 \pi_i \pi_j\right] \\ &= \phi^2 \left(\eta - \frac{1-\eta}{k}\right) W(i_1 \mid i) + \phi^2 \frac{1-\eta}{k} + (\phi - \phi^2)\pi_i + \phi\left(\eta - \frac{1-\eta}{k}\right)\pi_i + \\ &\quad \phi\frac{1-\eta}{k} - \phi^2\left(\eta - \frac{1-\eta}{k}\right)\pi_i - \phi^2\frac{1-\eta}{k} + (1 - 2\phi + \phi^2)\pi_i \qquad (3.7.1) \\ &= \phi^2 \left(\eta - \frac{1-\eta}{k}\right) W(i_1 \mid i) + \left(1 + \left(\eta - \frac{1-\eta}{k}\right)\phi\right)(1-\phi)\pi_i + \frac{1-\eta}{k}\phi.\end{aligned}$$

[Equation (3.7.1) can be written from the following results

$$\begin{aligned}\sum_{j=0}^{k} W(j \mid i)W(i_1 \mid j) &= \eta W(i_1 \mid i) + \frac{1-\eta}{k} \sum_{\substack{j=o \\ (\neq i)}}^{k} W(i_1 \mid j) \\ &= \eta W(i_1 \mid i) + \frac{1-\eta}{k} \left\{ \sum_{j=0}^{k} W(i_1 \mid j) - W(i_1 \mid i) \right\} \\ &= \left(\eta - \frac{1-\eta}{k}\right) W(i_1 \mid i) + \frac{1-\eta}{k},\end{aligned}$$

$$\sum_{j=0}^{k} \pi_i W(i_1 \mid j) = \pi_i,$$

,

$$\begin{aligned}\sum_{j=0}^{k} \pi_j W(i_1 \mid j) &= \eta \pi_i + \sum_{\substack{j=o \\ (\neq i)}}^{k} \frac{1-\eta}{k} \pi_j \\ &= \left(\eta - \frac{1-\eta}{k}\right) \pi_i + \frac{1-\eta}{k},\end{aligned}$$

and

$$\sum_{j=0}^{k} \pi_i \pi_j = \pi_i. \Big]$$

Let

$$\begin{aligned}p_n(i \mid i_1) &= \phi^n \left(\eta - \frac{1-\eta}{k}\right)^{n-1} W(i_1 \mid i) + \\ &\quad \left(1 + \left(\eta - \frac{1-\eta}{k}\right)\phi + \ldots + \left(\eta - \frac{1-\eta}{k}\right)^{n-1} \phi^{n-1}\right) \\ &\quad \times (1-\phi)\pi_i + \left(1 + \left(\eta - \frac{1-\eta}{k}\right)\phi + \ldots + \left(\eta - \frac{1-\eta}{k}\right)^{n-2} \phi^{n-2}\right) \frac{1-\eta}{k}\phi.\end{aligned}$$

Let us write $\psi = \left(\eta - \frac{1-\eta}{k}\right)\phi$, and then we have $p_n(i \mid i_1) = \phi\psi^{n-1}W(i_1 \mid i) + \frac{1-\psi^n}{1-\psi}(1-\phi)\pi_i + \frac{1-\psi^{n-1}}{1-\psi}\frac{1-\eta}{k}\phi$.

Now we have to prove for $(n+1)$th step ahead setup, $p_{n+1}(i \mid i_1) = \phi\psi^n W(i_1 \mid i) + \frac{1-\psi^{n+1}}{1-\psi}(1-\phi)\pi_i + \frac{1-\psi^n}{1-\psi}\frac{1-\eta}{k}\phi$. We can write $p_{n+1}(i \mid i_1)$ as

$$\begin{aligned}
p_{n+1}(i \mid i_1) &= P(Y_{t+n+1} = i \mid Y_t = i_1) \\
&= \sum_{j=0}^{k} P(Y_{t+n+1} = i \mid Y_{t+1} = j)\, P(Y_{t+1} = j \mid Y_t = i_1) \\
&= \sum_{j=0}^{k} \left[\phi\psi^{n-1} W(j \mid i) + \frac{1-\psi^n}{1-\psi}(1-\phi)\pi_i + \frac{1-\psi^{n-1}}{1-\psi}\frac{1-\eta}{k}\phi\right] \\
&\qquad \times \left[\phi W(i_1 \mid j) + (1-\phi)\pi_j\right] \\
&= \sum_{j=0}^{k} \left[\phi^2\psi^{n-1} W(j \mid i) W(i_1 \mid i) + \phi(1-\phi)\frac{1-\psi^n}{1-\psi}\pi_i W(i_1 \mid j)\right] + \\
&\qquad \sum_{j=0}^{k} \left[\frac{1-\psi^{n-1}}{1-\psi}\phi^2\frac{1-\eta}{k} W(i_1 \mid j)\right] \\
&+ \sum_{j=0}^{k} \left[(\phi-\phi^2)\psi^{n-1}\pi_j W(j \mid i) + \frac{1-\psi^n}{1-\psi}(1-2\phi+\phi^2)\pi_i\pi_j + \frac{1-\psi^{n-1}}{1-\psi}\phi(1-\phi)\frac{1-\eta}{k}\pi_j\right] \\
&= \phi\psi^n W(i_1 \mid i) + \frac{1-\psi^{n+1}}{1-\psi}(1-\phi)\pi_i + \frac{1-\psi^n}{1-\psi}\frac{1-\eta}{k}\phi. \\
&\qquad \text{(after some simplifications)}
\end{aligned}$$

So we have

$$p_h(i \mid Y_t) = P(Y_{t+h} = i \mid Y_t) = \phi\psi^{h-1} W(Y_t \mid i) + \frac{1-\psi^h}{1-\psi}(1-\phi)\pi_i + \frac{1-\psi^{h-1}}{1-\psi}\frac{1-\eta}{k}\phi.$$

Now we try to find the mathematical forms of $E(Y_{t+h} \mid Y_t)$ and $V(Y_{t+h} \mid Y_t)$.
We can write

$$\begin{aligned}
E(Y_{t+h} \mid Y_t) &= \phi\psi^{h-1}\sum_{i=0}^{k} i W(Y_t \mid i) + \frac{1-\psi^h}{1-\psi}(1-\phi)\mu + \frac{1-\psi^{h-1}}{1-\psi}\frac{(1-\eta)(k+1)}{2}\phi \\
&= \phi\psi^{h-1}\left[\left(\eta - \frac{1-\eta}{k}\right) Y_t + \frac{(1-\eta)(k+1)}{2}\right] + \frac{1-\psi^h}{1-\psi}(1-\phi)\mu \\
&\qquad + \frac{1-\psi^{h-1}}{1-\psi}\frac{(1-\eta)(k+1)}{2}\phi \\
&= \psi^h Y_t + \phi\psi^{h-1}\frac{(1-\eta)(k+1)}{2} + \frac{1-\psi^h}{1-\psi}(1-\phi)\mu \\
&\qquad + \frac{1-\psi^{h-1}}{1-\psi}\frac{(1-\eta)(k+1)}{2}\phi \\
&= \psi^h Y_t + \frac{1-\psi^h}{1-\psi}\left[(1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}\right].
\end{aligned}$$

Again

$$\begin{aligned} E({Y_{t+h}}^2 \mid Y_t) &= \phi\psi^{h-1}\sum_{i=0}^{k} i^2 W(Y_t \mid i) + \frac{1-\psi^h}{1-\psi}(1-\phi)(\mu^2+\sigma^2) + \\ &\qquad \frac{1-\psi^{h-1}}{1-\psi}\frac{(1-\eta)(k+1)(2k+1)}{6}\phi \\ &= \phi\psi^{h-1}\left[\left(\eta - \frac{1-\eta}{k}\right){Y_t}^2 + \frac{(1-\eta)(k+1)(2k+1)}{6}\right] + \\ &\qquad \frac{1-\psi^h}{1-\psi}(1-\phi)(\mu^2+\sigma^2) + \frac{1-\psi^{h-1}}{1-\psi}\frac{(1-\eta)(k+1)(2k+1)}{6}\phi \\ &= \psi^h {Y_t}^2 + \frac{1-\psi^h}{1-\psi}\left[(1-\phi)(\mu^2+\sigma^2) + \frac{(1-\eta)(k+1)(2k+1)}{6}\phi\right]. \end{aligned}$$

So we can write $V(Y_{t+h} \mid Y_t) = E({Y_{t+h}}^2 \mid Y_t) - E^2(Y_{t+h} \mid Y_t)$. Now, to prove that h-step ahead conditional mean and variance converge to marginal mean and variance, we need $\lim_{h\to\infty}\psi^h = 0$, i.e., the condition $\psi \in (-1,1)$. So to have this condition $\psi \in (-1,1)$ (or, $|\psi| < 1$) in place, we need

(i) $\left(\eta - \frac{1-\eta}{k}\right)\phi < 1 \Rightarrow \eta < \frac{1+\frac{k}{\phi}}{k+1}$, which is always true since $\phi < 1 \Rightarrow \frac{k}{\phi} > k \Rightarrow \frac{1+\frac{k}{\phi}}{k+1} > 1$, and

(ii) $\left(\eta - \frac{1-\eta}{k}\right)\phi > -1 \Rightarrow \eta > \frac{1-\frac{k}{\phi}}{k+1}$, which is always true since $k > 1 \Rightarrow \frac{k}{\phi} > \frac{1}{\phi} > 1 \Rightarrow 1 - \frac{k}{\phi} < 0$, i.e., $\eta >$ a negative quantity (true always), which is $\frac{1-\frac{k}{\phi}}{k+1}$.

Therefore, $\psi \in (-1,1)$ and we can write

$$\lim_{h\to\infty} E(Y_{t+h} \mid Y_t) = \frac{\frac{(1-\eta)(k+1)}{2}\phi + (1-\phi)\mu}{1-\left(\eta - \frac{1-\eta}{k}\right)\phi} = \mu_y,$$

and

$$\lim_{h\to\infty} V(Y_{t+h} \mid Y_t) = \frac{\frac{(1-\eta)(k+1)(2k+1)}{6}\phi + (1-\phi)(\mu^2+\sigma^2)}{1-\left(\eta - \frac{1-\eta}{k}\right)\phi} - E^2(Y_t) = {\sigma_y}^2.$$

Now for the autocovariance structure, we have

$$\begin{aligned} E(Y_{t+h}Y_t) &= EY_tE(Y_{t+h} \mid Y_t) \\ &= EY_t\left[\psi^h Y_t + \frac{1-\psi^h}{1-\psi}\left((1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}\right)\right] \\ &= \psi^h({\mu_y}^2 + {\sigma_y}^2) + \frac{1-\psi^h}{1-\psi}\left((1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}\right)\mu_y, \end{aligned}$$

which depends only on h.
Now for the autocorrelation structure, we have

$$\begin{aligned} \rho_y(h) &= cor(Y_t, Y_{t+h}) \\ &= \frac{\left[\psi^h({\mu_y}^2 + {\sigma_y}^2) + \frac{1-\psi^h}{1-\psi}\left((1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}\right)\mu_y - {\mu_y}^2\right]}{{\sigma_y}^2} \\ &= \frac{\left[\psi^h({\mu_y}^2 + {\sigma_y}^2) + \left(1-\psi^h\right)\left(\frac{(1-\phi)\mu + \phi\frac{(1-\eta)(k+1)}{2}}{1-\left(\eta - \frac{1-\eta}{k}\right)\phi}\right)\mu_y - {\mu_y}^2\right]}{{\sigma_y}^2} \\ &= \frac{\left[\psi^h({\mu_y}^2 + {\sigma_y}^2) + \left(1-\psi^h\right)(\mu_y)\,\mu_y - {\mu_y}^2\right]}{{\sigma_y}^2} \quad \text{(from the expression of } \mu_y) \\ &= \frac{\left[\psi^h{\mu_y}^2 + \psi^h{\sigma_y}^2 + {\mu_y}^2 - \psi^h{\mu_y}^2 - {\mu_y}^2\right]}{{\sigma_y}^2} \\ &= \psi^h, \end{aligned}$$

and so

$$\lim_{h\to\infty} \rho_y(h) = 0.$$

Appendix F

In Table 3.8, we present some more numerical results for the simulation study regarding forecasting. We also execute this study for 1000 replications, using a sample size of 500 observations (the first 400 for the training set and the last 100 for the test set). Here, the truncation is beyond 8. From the results, we can see that the mPAR(1) process relatively performs better than the PAR(1) process, especially in terms of the mode predictor.

TABLE 3.8: Different forecasting measures for varying h where the data generating process is the proposed mPAR(1) process for different sets of (ϕ, Λ, η) with right-truncation beyond 8

	mPAR(1)				PAR(1)			
h	1	2	3	4	1	2	3	4
Measures	(ϕ, Λ, η)=(0.4,2.5,0.5)							
PRMSE(h)	1.9973	2.0038	2.0041	2.0039	1.9975	2.0038	2.0041	2.0039
PTP[mean](h)	18.61	18.66	18.66	18.65	18.56	18.66	18.66	18.65
PTP[mode](h)	29.83	21.63	21.58	21.57	29.52	21.24	20.82	20.90
PTPI[mode](h)	53.17	57.42	58.44	58.43	53.87	56.56	56.91	56.94
	(ϕ, Λ, η)=(0.4,2.5,0.6)							
PRMSE(h)	1.9205	1.9488	1.9488	1.9497	1.9210	1.9486	1.9488	1.9497
PTP[mean](h)	19.69	19.32	19.29	19.28	19.61	19.31	19.29	19.28
PTP[mode](h)	34.08	22.97	22.46	22.48	34.02	22.78	21.96	22.07
PTPI[mode](h)	56.13	57.59	59.87	59.90	56.26	58.12	59.05	59.12
	(ϕ, Λ, η)=(0.5,2.5,0.5)							
PRMSE(h)	2.0821	2.1119	2.1106	2.1099	2.0819	2.1120	2.1105	2.1099
PTP[mean](h)	17.19	17.61	17.61	17.62	17.18	17.61	17.61	17.62
PTP[mode](h)	32.90	21.01	20.40	20.46	32.82	19.49	18.47	18.30
PTPI[mode](h)	53.35	53.04	54.84	54.92	53.38	51.69	52.15	52.00
	(ϕ, Λ, η)=(0.5,2.5,0.6)							
PRMSE(h)	2.0042	2.0621	2.0616	2.0628	2.0045	2.0614	2.0616	2.0628
PTP[mean](h)	18.19	18.31	18.35	18.34	18.18	18.30	18.35	18.34
PTP[mode](h)	38.23	22.40	20.86	20.87	38.23	21.69	20.12	19.78
PTPI[mode](h)	57.42	53.83	56.39	56.83	57.42	52.91	54.36	54.18
	(ϕ, Λ, η)=(0.6,2.5,0.5)							
PRMSE(h)	2.1578	2.2230	2.2227	2.2231	2.1588	2.2228	2.2232	2.2232
PTP[mean](h)	16.31	16.29	16.51	16.50	16.36	16.32	16.47	16.48
PTP[mode](h)	36.15	20.45	19.09	19.15	36.15	18.94	17.28	16.69
PTPI[mode](h)	54.12	49.04	51.38	51.65	54.12	47.81	48.59	48.31
	(ϕ, Λ, η)=(0.6,2.5,0.6)							
PRMSE(h)	2.0665	2.1686	2.1713	2.1709	2.0671	2.1684	2.1717	2.1711
PTP[mean](h)	17.09	16.71	17.02	17.04	17.07	16.73	17.03	17.04
PTP[mode](h)	41.96	23.06	19.66	19.60	41.96	22.33	18.45	17.58
PTPI[mode](h)	58.91	48.92	52.10	53.47	58.91	49.81	50.42	50.26

Appendix G. Some forecasting measures

The mathematical form of PTP(h) measure is given in Section 2.8 of Chapter 2. The mathematical forms of PRMSE(h) and PTPI(h) are given as follows:

1. The h-step ahead predicted root mean squared error [denoted by PRMSE(h)] is defined as

$$PRMSE(h) = \sqrt{\frac{\sum_{t=r}^{r+s-h} \left(Y_{t+h} - \hat{Y}_{t+h}\right)^2}{s-h+1}}.$$

2. The h-step ahead PTP within an interval [denoted by PTPI(h)] is defined as

$$PTPI(h) = \frac{\sum_{t=r}^{r+s-h} I\left[\left|Y_{t+h} - \hat{Y}_{t+h}\right| \leq \delta\right]}{s-h+1} \times 100\%.$$

Here, total sample size is $(r+s)$, out of which training set consists of first r observations and test set contains remaining s observations. In our study, $\delta = 1$, and $\hat{Y}_t$'s are the mean or mode predictors.

Chapter 4

Change-point analysis through INAR process with application to some COVID-19 data

4.1 Introduction

In this chapter, we use the INAR process (see McKenzie, 1985; McKenzie, 1986, Al-Osh and Alzaid, 1987) to model COVID-19 active cases, an example of count time series data (see first part of Weiß, 2018, i.e., mainly Chapters 2 to 5) with time-varying properties. At time-point t, there are two components in an INAR process: (i) non-recovery cases from the previous time-point (survival part) and (ii) new cases entering the process at time-point t (innovation part). In the existing INAR processes, the innovation terms usually contain no time-varying covariates, i.e., the new cases entering the process are not time-dependent. However, in real-world scenarios, we can see that the rapid change in the number of infected cases renders the innovation terms time-dependent. Aside from the time-varying nature of the innovation terms, we can as well observe some change-points in these datasets. In the the COVID-19 pandemic scenarios, we observed mainly two types of curves for daily new cases reported in different parts of the world: (i) the curve, at first, began to increase exponentially, but after major steps like nationwide lockdowns, implementation of social distancing measures, a massive number of tests, etc. taken by the respective authorities in different countries, the curve started decreasing, and (ii) the curve that came down started to rise again. The curves of daily active cases changed in the same way. As a result, we can clearly identify one change-point (upwards to downwards) for a curve as in Case (i) and two (upwards to downwards and then downwards to upwards)

for a curve as in Case (ii). In this chapter, we attempt to develop a new kind of Poisson INAR (PINAR) process (see Al-Osh and Alzaid, 1987) based on the binomial thinning operator (see Steutel and Van Harn, 1979) for count time series data such as the COVID-19 data, in which we model the innovation terms using some time-varying covariates and smoothing change-point function without changing the survival part.

The PINAR process is widely used to model count time series data due to its simplicity. However, the PINAR process based on the binomial thinning operator cannot handle count time series data with both change-points and time-varying innovation terms. As a result, we introduce a new suitably adjusted PINAR model that can address both of these features found in the COVID-19 datasets. To incorporate the change-points in our proposed PINAR model, the innovation terms are modelled with a smoothing version (see Smooth Maximum, n.d.) of time-varying covariate that includes the change-points. The idea of capturing change-points in the innovations through time-varying smoothing covariate is inspired by Chan and Tong, 1986, Hansen, 2000 and Fong et al., 2017, whose works are mainly based on continuous data. We use this smoothing version of time-varying components in our proposed model to capture the changing curvatures in daily active case data. The effectiveness of the proposed model for both one- and two-change-point studies is later evaluated using some simulation studies and two COVID-19 datasets. We compare our proposed model to another PINAR model with time-varying covariates but no change-point to demonstrate overall performance of the proposed model. The contents of this chapter are partially based on Chattopadhyay et al., 2021.

The remainder of the chapter is arranged as follows. Section 4.2 discusses two real COVID-19 datasets. Section 4.3 describes our proposed model and its different components. We provide the distributional properties of our proposed model along with the h-step ahead forecasting distribution in Sections 4.4 and 4.5, respectively. In Section 4.3.4, we discuss the estimation method for our proposed model. Section 4.7 includes extensive simulation studies concerning our proposed model. In Section 4.8, we examine the COVID-19 datasets with respect to our proposed methodology. Section 4.9 summarizes the findings from this study. Appendix (Section 4.10) contains all the proofs for the theoretical results.

4.2 Motivating data examples: COVID-19 data

The world just experienced the most severe global health crisis in recent memory, the COVID-19 pandemic. The outbreak was first detected in Wuhan, China, in

early December 2019. On January 30, the World Health Organisation declared the outbreak a Public Health Emergency of International Concern, and on March 11, it was classified as a pandemic.

To limit the spread of this virus in its early stages, authorities around the world implemented stringent measures such as nationwide lockdowns, rapid testing, strict social distancing, the use of masks and sanitizers in public places, and so on. As a result, in some parts of the world, the situation with COVID-19 improved in early stages, and the lockdowns were lifted. During that time, some Gulf evacuations took place in various countries. As a result of the highly infectious nature of this virus, Community Transmission began in those parts of the world, and the number of infected cases again began to rise. In Sections 4.2.1 and 4.2.2, we discuss two such real COVID-19 datasets.

4.2.1 COVID-19 data of Italy

This dataset is an example of Case (i), as described in Section 4.1. We only see one change-point in the data of daily new cases (and hence in the data of daily active cases) in Italy (see Figures 4.1 and 4.2). Therefore, the study is based on single change-point analysis. The data (see Worldometer, n.d.) was collected from February 15 to June 6, totaling 113 days. Though the first case in this country was discovered in January 2020, cases began to increase rapidly at the beginning of March. Following continuous measurements by the authorities, the curve of cases began to fall. As of June 6, 2020, there were over 234,000 confirmed cases and over 33.8 thousand deaths. The active number of cases exceeded 35,000. Here, we mainly focus on the order 1 process. That is, we fit an AR(1) model to the given dataset with the assumption that Y_t depends only on its recent lag value Y_{t-1}.

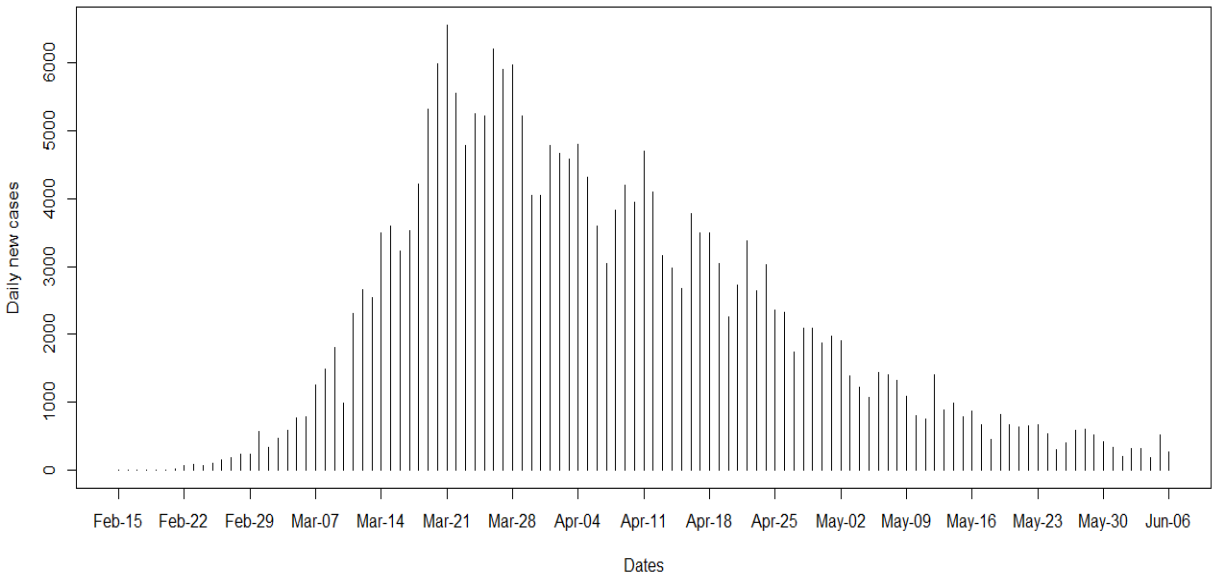


FIGURE 4.1: Daily new cases in Italy

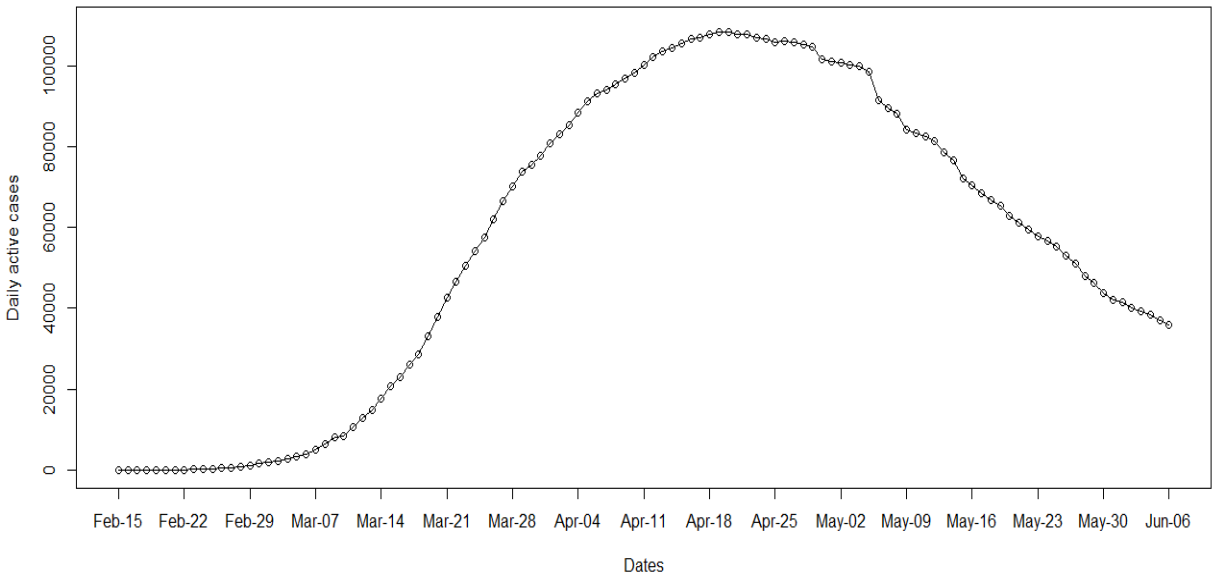


FIGURE 4.2: Daily active cases in Italy

4.2.2 COVID-19 data of Kerala

This dataset is an example of Case (ii), where we can observe two change-points in the data of daily new cases and hence in the data of daily active cases. Therefore, the study is based on the analysis of two change-points. In Kerala, the first

case was discovered in January 2020, but cases began to pile up around mid-March. The curve of cases fell due to heavy measurements taken by the Kerala state government, but the cases began to rise again in mid-May as Gulf evacuees began to arrive in the state. The data for Kerala (see GoK Dashboard, n.d.) was collected from March 9 to June 6, totaling 90 days. As of June 6, 2020, more than 1800 cases were reported in Kerala, with 15 deaths, and there were more than 1000 active cases. Figures 4.3 and 4.4 show new daily cases and daily active cases. Here also, we mainly focus on the order 1 process. That is, we fit an AR(1) model to the given dataset with the assumption that Y_t depends only on its recent lag value Y_{t-1}.

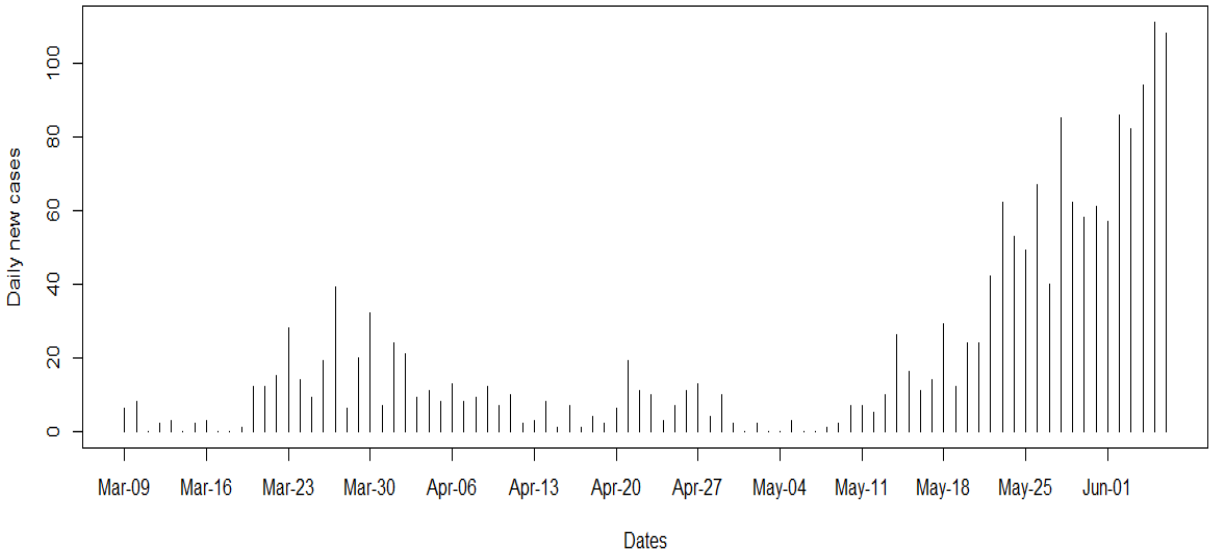


FIGURE 4.3: Daily new cases in Kerala

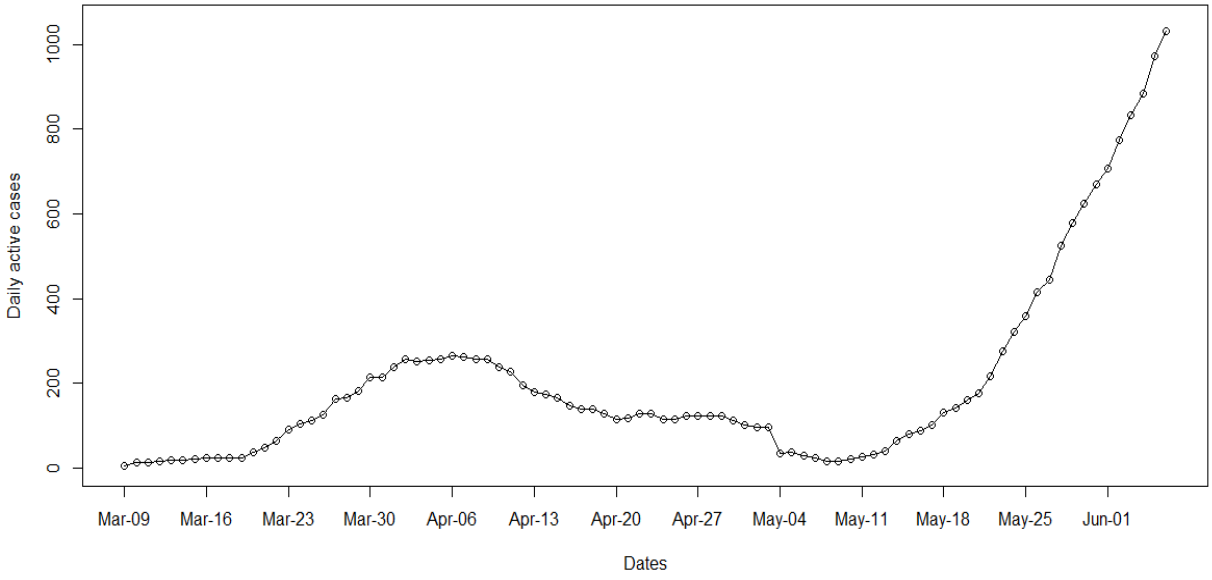


FIGURE 4.4: Daily active cases in Kerala

4.3 The Model

In this section, we discuss a new INAR(1) model to capture the change-points in the count time series datasets like the COVID-19 datasets of Italy and Kerala.

Here, we employ the INAR(1) process (see McKenzie, 1985 and Al-Osh and Alzaid, 1987) with the binomial thinning operator (see Steutel and Van Harn, 1979) to develop our proposed model for change-point analysis, which is given by

$$Y_t = \alpha \circ Y_{t-1} + \varepsilon_t, \tag{4.3.1}$$

where Y_t denotes the number of daily active cases at time-point t and ε_t represents daily new cases reported at time-point t. We assume that ε_t with one change-point follows Poisson(λ_t^{c1}), where λ_t^{c1} is assumed to have the following form

$$\lambda_t^{c1} = \exp\left(\beta_0^{c1} + \beta_1^{c1}\frac{(t-t_{ch})\exp(\delta_n(t-t_{ch}))}{1+\exp(\delta_n(t-t_{ch}))} + \beta_2^{c1}t\right), \tag{4.3.2}$$

where the tuning parameter $\delta_n(>0)$ helps to capture the changing curvature of the data. Here, t_{ch} denotes the single change-point in the data like the COVID-19 data of Italy. The above model is defined for one change-point. We denote $\boldsymbol{\beta^{c1}}$ = $(\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1})$ as the regression parameters, where β_1^{c1} is the regression coefficient for the time-varying covariate consisting of the change-point. However, we

can easily extend the model for two change-points. For example, for two change-points, the functional form of λ_t^{c2} is given by

$$\lambda_t^{c2} = \exp\left(\beta_0^{c2} + \beta_1^{c2}\frac{(t-t_{ch1})\exp(\delta_n(t-t_{ch1}))}{1+\exp(\delta_n(t-t_{ch1}))} + \beta_2^{c2}\frac{(t-t_{ch2})\exp(\delta_n(t-t_{ch2}))}{1+\exp(\delta_n(t-t_{ch2}))} + \beta_3^{c2}t\right), \tag{4.3.3}$$

where t_{ch1} and t_{ch2} are two change-points in the data. Here, $\boldsymbol{\beta^{c2}} = (\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2})$ are the regression parameters, in which β_1^{c2} and β_2^{c2} are the regression coefficients for the time-varying covariates consisting of the change-points. In the next section, we provide a general idea about our proposed model.

Remark 4.1 *For data with two change-points, it is possible to use two tuning parameters, such as δ_{1n} and δ_{2n}, instead of just one. However, in our proposed process for two change-points, we use only δ_n instead of δ_{1n} and δ_{2n}, primarily because using δ_n reduces computational difficulty and simplifies the form of the proposed model.*

4.3.1 Idea behind the model

The idea behind the forms of λ_t^{c1} and λ_t^{c2}, discussed in equations (4.3.2) and (4.3.3), comes from the threshold regression model setup (see Chan and Tong, 1986; Hansen, 2000; Fong et al., 2017). From the concept of the segmented model in the threshold regression setup, we can write the form of $\log(\lambda_t^{c1})$ for one change-point as $\log(\lambda_t^{c1}) = \beta_0^{c1} + \beta_1^{c1}(t-t_{ch})_+ + \beta_2^{c1}t$, where $(t-t_{ch})_+ = (t-t_{ch})$ for $t > t_{ch}$ and $(t-t_{ch})_+ = 0$ for $t \leq t_{ch}$. In this segmented form of $\log(\lambda_t^{c1})$, we notice sharp changes (upwards to downwards or downwards to upwards) in the curve of daily new cases and thus in the curve of daily active cases. However, in real-life scenarios, such as the COVID-19 data, we do not see sharp changes; instead, we notice changing curvature(s). So, we attempt to capture those changing curvature(s) in the data of daily active cases by modelling the data of daily new cases (innovation terms) in the proposed model using some time-varying covariates and smoothing change-point functions. Moreover, the function $(t-t_{ch})_+$ is not differentiable at t_{ch}. So, we replace $(t-t_{ch})_+$ (for $\delta_n > 0$) by a smooth differentiable maximum function (see Smooth Maximum, n.d.), which is given by

$$(t-t_{ch})_+ \approx \frac{0\times\exp(0\times\delta_n) + (t-t_{ch})\exp(\delta_n(t-t_{ch}))}{\exp(0\times\delta_n)+\exp(\delta_n(t-t_{ch}))}.$$

Hence, the functional form of λ_t for one change-point is given by

$$\log(\lambda_t^{c1}) = \beta_0^{c1} + \beta_1^{c1}\frac{(t-t_{ch})\exp(\delta_n(t-t_{ch}))}{1+\exp(\delta_n(t-t_{ch}))} + \beta_2^{c1}t \quad \text{for } \delta_n > 0,$$

i.e.,

$$\lambda_t^{c1} = \exp\left(\beta_0^{c1} + \beta_1^{c1}\frac{(t-t_{ch})\exp(\delta_n(t-t_{ch}))}{1+\exp(\delta_n(t-t_{ch}))} + \beta_2^{c1}t\right) \quad \text{for } \delta_n > 0.$$

In the similar way, we can find the functional form of λ_t^{c2} for two change-points, which is given by

$$\lambda_t^{c2} = \exp\left(\beta_0^{c2} + \beta_1^{c2}\frac{(t-t_{ch1})\exp(\delta_n(t-t_{ch1}))}{1+\exp(\delta_n(t-t_{ch1}))} + \beta_2^{c2}\frac{(t-t_{ch2})\exp(\delta_n(t-t_{ch2}))}{1+\exp(\delta_n(t-t_{ch2}))} + \beta_3^{c2}t\right) \quad \text{for } \delta_n > 0.$$

4.3.2 Conditions on model parameters

The changing behaviours of these datasets depend on some conditions on β_i^{c1}'s and β_i^{c2}'s. We provide the conditions through the form of the segmented model of threshold regression setup for both sets: (i) $\beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1}$ in equation (4.3.2) and (ii) $\beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2}$ in equation (4.3.3). These conditions help our proposed model to capture the change-point(s). The required conditions for both the studies of one change-point and two change-points are given below.

(i) In the segmented form of $\log(\lambda_t^{c1})$ for one change-point analysis, we model $\log(\lambda_t^{c1})$ as

$$\log(\lambda_t^{c1}) = \begin{cases} \beta_0^{c1} + \beta_2^{c1}t & \text{for } t \le t_{ch} \\ \beta_0^{c1} + \beta_1^{c1}(t-t_{ch}) + \beta_2^{c1}t & \text{for } t > t_{ch} \end{cases}$$

So, the derivatives of $\log(\lambda_t^{c1})$ are

$$\begin{cases} \beta_2^{c1} & \text{for } t \le t_{ch} \\ (\beta_1^{c1} + \beta_2^{c1}) & \text{for } t > t_{ch} \end{cases}$$

in this segmented model setup for one change-point. So, for $\beta_2^{c1} > 0$ and $(\beta_1^{c1} + \beta_2^{c1}) < 0$, $\log(\lambda_t^{c1})$ increases when $t \le t_{ch}$ and decreases when $t > t_{ch}$, i.e., λ_t^{c1} increases when $t \le t_{ch}$ and decreases when $t > t_{ch}$. Hence, the change-point in the data of daily new cases is t_{ch}. So, for the count time series data of one change-point, the condition: $\{\beta_2^{c1} > 0, (\beta_1^{c1} + \beta_2^{c1}) < 0\}$ must hold.

(ii) Similarly for the study of two change-points, we model $\log(\lambda_t^{c2})$ as

$$\log(\lambda_t^{c2}) = \begin{cases} \beta_0^{c2} + \beta_3^{c2} t & \text{for } t \leq t_{ch1} \\ \beta_0^{c2} + \beta_1^{c2}(t - t_{ch1}) + \beta_3^{c2} t & \text{for } t_{ch1} < t \leq t_{ch2} \\ \beta_0^{c2} + \beta_1^{c2}(t - t_{ch1}) + \beta_2^{c2}(t - t_{ch2}) + \beta_3^{c2} t & \text{for } t > t_{ch2} \end{cases}$$

Hence, the derivatives of $\log(\lambda_t^{c2})$ are

$$\begin{cases} \beta_3^{c2} & \text{for } t \leq t_{ch1} \\ (\beta_1^{c2} + \beta_3^{c2}) & \text{for } t_{ch1} < t \leq t_{ch2} \\ (\beta_1^{c2} + \beta_2^{c2} + \beta_3^{c2}) & \text{for } t > t_{ch2} \end{cases}$$

in the segmented model for two change-points. So, for $\beta_3^{c2} > 0$, $(\beta_1^{c2} + \beta_3^{c2}) < 0$ and $(\beta_1^{c2} + \beta_2^{c2} + \beta_3^{c2}) > 0$, $\log(\lambda_t^{c2})$ increases when $t \leq t_{ch1}$, decreases when $t_{ch1} < t \leq t_{ch2}$ and again increases when $t > t_{ch2}$, i.e., λ_t^{c2} increases when $t \leq t_{ch1}$, decreases when $t_{ch1} < t \leq t_{ch2}$ and again increases when $t > t_{ch2}$. Here, the two change-points in the data of daily new cases are t_{ch1} and t_{ch2}. So, the condition: $\{\beta_3^{c2} > 0, (\beta_1^{c2} + \beta_3^{c2}) < 0, (\beta_1^{c2} + \beta_2^{c2} + \beta_3^{c2}) > 0\}$ must hold for the count time series data containing two change-points.

4.3.3 Choices of the tuning parameter δ_n

The tuning parameter δ_n in our proposed model captures the changing curvature(s) in the data. Here, $\delta_n > 0$. To compute the optimal value of the tuning parameter δ_n from the data, we consider a grid search method (idea taken from Chakraborty, Laber, and Zhao, 2013 and James et al., 2013). This method uses a goodness-of-fit measure to calculate the optimal value of δ_n. The concept of δ_n is derived from Smooth Maximum, n.d. As the value of δ_n increases, the changing curvature becomes sharper. Figures 4.5 and 4.6 demonstrate this property. As the values of δ_n shift from 0.05 to 1, the changing curvatures become sharper for both the studies of one and two change-points. We also include the non-smoothing version (without δ_n) of the generated data, i.e., segmented data, for comparison purpose.

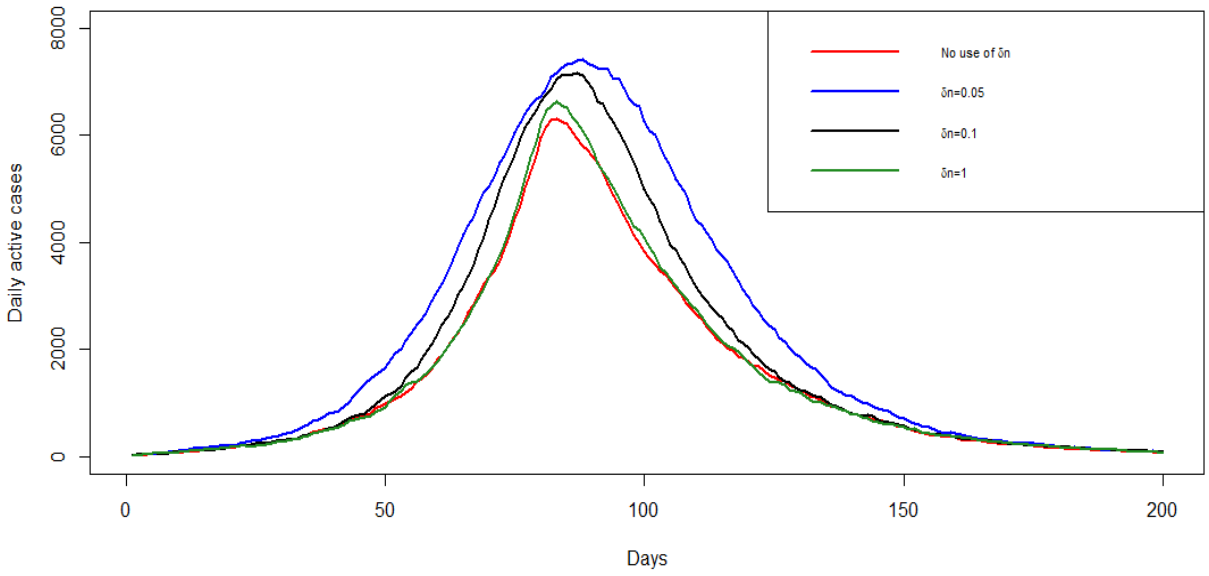


FIGURE 4.5: The changing curvatures for one change-point study for $\delta_n = 0.05, 0.1, 1$ along with segmented data (no use of δ_n)

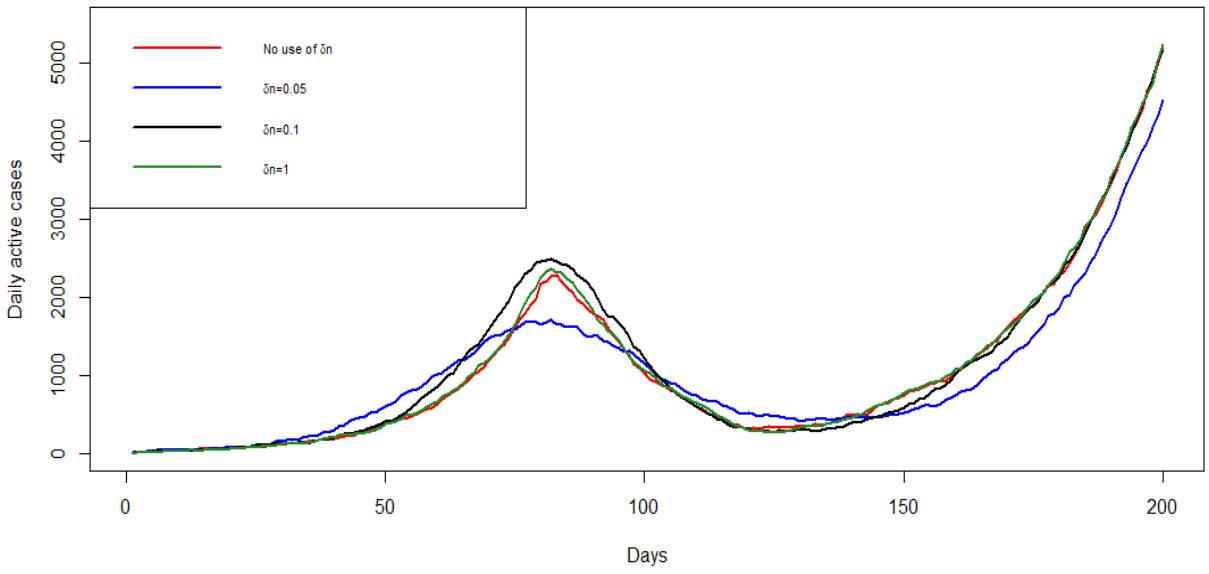


FIGURE 4.6: The changing curvatures for two change-point study for $\delta_n = 0.05, 0.1, 1$ along with segmented data (no use of δ_n)

4.3.4 Detection of change-point(s)

To identify the change-point(s), we first calculate the difference between two consecutive observations (i.e., Δ_t = the value of the observation at time-point t minus the value of the observation at time-point $(t-1)$). Then, we consider the sign of those differences denoted by $S_t = sign(\Delta_t)$, where S_t takes $'+'$ if $\Delta_t > 0$, and $'-'$ otherwise. For a dataset with one change-point, the sequence $\{S_t\}$ should yield two runs: (1) $'+'$ and (2) $'-'$ (see Wald and Wolfowitz, 1940). The run of $'+'$ and run of $'-'$ will be reversed depending on whether the curve of the observations is increasing or decreasing. If the original time series plot of the observations is bell-shaped (i.e., initially the observations are increasing and then after a certain time-point (say, t_{ch}) the observations are decreasing), we will have a run of $'+'$ first and then a run of $'-'$ after the time-point t_{ch}. We can identify the change-point t_{ch} by determining when the first run of $'+'$ ends. However, in real-life scenarios, time series data with a single change-point may not be smooth, and random fluctuations are common. As a result, there may be many small $'+'$ and $'-'$ runs, making the above detection procedure difficult to determine the change-point which is close to the true one. As a result, we use a pre-smoothing technique before implementing the above run-based change-point detection method. That is, rather than working with actual time series data, we smooth the data using appropriate statistical approaches.

Here, we employ the popular LOESS method for this smoothing procedure. LOESS (locally estimated scatterplot smoothing) is a nonparametric method for smoothing a series of data in which no assumptions are made about the underlying structure of the data (see Cleveland, Grosse, and Shyu, 2017). We use the "loess()" function in R software with the default degree choice and a grid of span, which controls the degree of smoothing. The grid is taken as the interval [0.10, 0.99] with an increment of 0.01. In our study, we use that value of the span parameter in the interval, for which we, for the first time, obtain two runs, i.e., a run of $'+'$ and a run of $'-'$. That is, we consider the first span parameter value that gives two runs ($'+'$ and $'-'$) from the smoothed data. Note that, the higher the value of span parameter compared to the first value for which we obtain two runs, the less the model attempts to hug the data-points. Therefore, the span value for which two runs are detected for the first time from the smoothed data should provide us with a satisfactory result for obtaining the change-point. This is because the smoothed data obtained by that first span parameter value embraces the real data more than any other values of span parameter that provide two runs ($'+'$ and $'-'$). Hence, we smooth the real data with that span value and calculate the change-point from the point at which the first run ends (the value of

the point at which the first run ends + 1) as the desired change-point.

For time series data with two change-points (say, t_{ch1} and t_{ch2}), like the Kerala data, the sequence $\{S_t\}$ should produce three runs: (1) run of '+', (2) run of '−', and (3) run of '+'. The run of '+' and the run of '−' will be switched twice, i.e., a run of '+' for increasing curvature, a run of '−' for decreasing curvature, and another run of '+', when observations begin to rise again (another increasing curvature). We also use the LOESS method here, taking into account the first value of span parameter from which we obtain three runs ('+', '−' and again '+'). Then, using that span value, we obtain two change-points in the same way that we would for data with one change-point.

Note 4.1 *The idea for this study stems primarily from the datasets used in the analysis, which are the Italy COVID-19 data and the Kerala COVID-19 data. For this type of dataset, having data of daily new cases as well as daily active cases is both useful and important. Curvature changes in this type of dataset (e.g., increasing to decreasing or decreasing to increasing) occur first with daily new cases, and then the changes reflect with daily active cases. The active cases at the current time-point can be divided into two parts: (i) survivors from the previous time-point and (ii) new cases reported at the current time-point. Because one of two components of daily active cases is the survivors from previous time-point, the change that occurs first with daily new cases is noticed later with daily active cases. And when a large number of cases are reported in a single location, such as the Italy COVID-19 data, this phenomenon is clearly visible. As a result, change-point detection should be performed using data of daily new cases, if available. Therefore, we use the above-mentioned method to detect change-points through smoothing by LOESS on the data of daily new cases.*

If the data of daily new cases is not available, we have no option but to work with daily active cases. However, one way may be employed to get a better understanding of the analysis by factoring in some lag values in the experiment. For example, in the case of COVID-19 data, if daily data of new cases is not available (which is rare), we should use some lag values (around one to two weeks because the incubation period for this virus is around 14 days) of the change-point detected from the daily data of active cases, in our analysis. This should provide a more accurate picture of the situation than the study that is only based on the change-point detected from the daily data of active cases. However, for our study, we have daily data of new cases available for both cases, which we use to detect change-points.

Some numerical results

The above-mentioned method is an ad-hoc one. Therefore, we perform some simulation studies to see whether this method can provide satisfactory results. For this study, we generate 100 datasets of new cases for both studies through the equations (4.3.2) and (4.3.3) since the innovation terms (new cases) follow Poisson distribution with parameters λ_t^{c1} and λ_t^{c2} for one change-point and two change-point studies, respectively. Each study is repeated for 100 times.

1. Using the above-mentioned method, we perform this study for one change-point for following four cases, where n is sample size. $n = 100$ for our study. The simulation results are given as the detected change-points with their frequencies in bracket, i.e., change-point(frequency). The results are given as follows:

   - The setup is: $(\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1}) = (0.8, 3, -0.2, 0.12)$ with $\delta_n = 0.5$ with $0.4n$ as the change-point. From the simulation results, we obtain the change-points with their frequencies as *40(49), 41(44), 43(7)*.
   - The setup is: $(\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1}) = (0.8, 3, -0.2, 0.12)$ with $\delta_n = 0.5$ with $0.5n$ as the change-point. From the simulation results, we obtain the change-points with their frequencies as *50(96), 51(4)*.
   - The setup is: $(\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1}) = (0.5, 2, -0.16, 0.08)$ with $\delta_n = 0.1$ with $0.4n$ as the change-point. From the simulation results, we obtain the change-points with their frequencies as *36(1), 38(7), 39(53), 40(24), 41(14), 43(1)*.
   - The setup is: $(\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1}) = (0.5, 2, -0.16, 0.08)$ with $\delta_n = 0.1$ with $0.5n$ as the change-point. From the simulation results, we obtain the change-points with their frequencies as *48(5), 49(20), 50(56), 51(13), 52(5), 53(1)*.

2. Now, we perform this study for two change-points for four cases. The simulation results are given as the detected change-points with the frequencies in bracket, i.e., first change-point(frequency) and second change-point(frequency).

   - The setup is: $(\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2}) = (0.8, 2, -0.15, 0.1, 0.1)$ with $\delta_n = 0.5$ with $0.3n$ and $0.6n$ as the change-points. From the simulation results, we obtain the change-points with their frequencies as (a) first change-point: *29(1), 30(29), 31(52), 32(18)* and (b) second change-point: *57(3), 58(8), 59(19), 60(42), 61(17), 62(10), 63(1)*.

- The setup is: $(\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2}) = (0.8, 2, -0.15, 0.1, 0.1)$ with $\delta_n = 0.5$ with $0.25n$ and $0.5n$ as the change-points. From the simulation results, we obtain the change-points with their frequencies as (a) first change-point: *24(1), 25(25), 26(57), 27(16), 28(1)* and (b) second change-point: *47(4), 48(10), 49(18), 50(38), 51(12), 52(11), 53(5), 54(1), 55(1)*.
- The setup is: $(\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2}) = (0.5, 3.5, -0.1, 0.1, 0.05)$ with $\delta_n = 0.3$ with $0.3n$ and $0.6n$ as the change-points. From the simulation results, we obtain the change-points with their frequencies as (a) first change-point: *28(2), 29(32), 30(37), 31(25), 32(4)* and (b) second change-point: *56(2), 57(5), 58(10), 59(21), 60(31), 61(17), 62(10), 63(3), 64(1)*.
- The setup is: $(\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2}) = (0.5, 3.5, -0.1, 0.1, 0.05)$ with $\delta_n = 0.3$ with $0.25n$ and $0.5n$ as the change-points. From the simulation results, we obtain the change-points with their frequencies as (a) first change-point: *23(5), 24(22), 25(47), 26(20), 27(6)* and (b) second change-point: *47(3), 48(10), 49(18), 50(34), 51(20), 52(13), 53(1), 54(1)*.

Overall, looking at the simulation results for both the studies, we can say that the method provides satisfactory results.

4.4 Distributional properties

In this section, we investigate the conditional and marginal distributions of the proposed model. This study uses λ_t^{c1} and λ_t^{c2} for one and two change-point analyses, respectively. For simplicity, we use λ_t^c to derive theoretical results, where λ_t^c can be λ_t^{c1} or λ_t^{c2} depending on the analysis.

4.4.1 Conditional distribution

Under our proposed setup, the conditional distribution of Y_t given Y_{t-1} and $\mathcal{X}_t^c$ (the set of all covariates up to time-point t including smooth time-varying and simple time-varying covariates up to time-point t) can be derived as

$$p(j \mid i) = P(Y_t = j \mid Y_{t-1} = i, \mathcal{X}_t^c) = \sum_{k=0}^{\min(i,j)} \binom{i}{k} \alpha^k (1-\alpha)^{i-k} \exp(-\lambda_t^c) \, (\lambda_t^c)^{j-k} \, [(j-k)!]^{-1} I_{((j-k)=(0,1,\ldots.))}, \tag{4.4.1}$$

where $I_{(\cdot)}$ is the indicator function. Note that, equation (4.4.1) is the probability of going from state i to state j in a single step. The conditional mean and variance

can be given as $E(Y_t \mid Y_{t-1}, \mathcal{X}_t^c) = \alpha Y_{t-1} + \lambda_t^c$, and $V(Y_t \mid Y_{t-1}, \mathcal{X}_t^c) = \alpha(1-\alpha)Y_{t-1} + \lambda_t^c$ respectively.

4.4.2 Marginal distribution

The marginal distribution of Y_t is difficult to obtain, so we find the partial marginal distribution of Y_t given $\mathcal{X}_t^c$ for $t > 1$, henceforth called the marginal distribution. We calculate the probability generating function (PGF) of Y_n given $\mathcal{X}_n^c$.

The derivation is valid for $t > 1$ and hence we assume that given $\mathcal{X}_1^c$, the marginal distribution of Y_1 is Poisson(λ_1^c). The reason behind this assumption can be given as follows. We know the elements that enter the system in the interval $(t-1, t]$ are the innovation terms at time-point t, which is ε_t. Now for $t = 1$, the interval is (0,1], and there is no previous existing interval in the system. So, in the interval (0,1], the elements that enter the system can be seen as the first count process Y_1. Hence, we can assume $Y_1 \mid \mathcal{X}_1^c \sim$ Poisson(λ_1^c).

Result 4.1 *Under the assumptions that $Y_1 \mid \mathcal{X}_1^c \sim$ Poisson(λ_1^c) and $\varepsilon_n \mid \mathcal{X}_n^c \sim$ Poisson(λ_n^c), we can show that the PGF of $Y_n \mid \mathcal{X}_n^c$ is*

$$\Phi_{Y_n|\mathcal{X}_n^c}(s) = \exp\left[-(\alpha^{n-1}\lambda_1^c + \alpha^{n-2}\lambda_2^c + ... + \lambda_n^c)(1-s)\right], \tag{4.4.2}$$

i.e., Y_n given $\mathcal{X}_n^c$, follows Poisson distribution with mean $\left(\alpha^{n-1}\lambda_1^c + \alpha^{n-2}\lambda_2^c + ... + \lambda_n^c\right)$.

The derivation of this result is presented in Appendix (see Section 4.10).
Additionally, a recursive formula can be employed as an alternative method for deriving the marginal distribution, which is given by

$$\begin{aligned}
p_{t,j} &= P(Y_t = j \mid \mathcal{X}_t^c) \\
&= \sum_{i=0}^{\infty} p(j \mid i) P(Y_{t-1} = i \mid \mathcal{X}_t^c) \\
&= \sum_{i=0}^{\infty} \sum_{k=0}^{\min(i,j)} \binom{i}{k} \alpha^k (1-\alpha)^{i-k} \exp(-\lambda_t^c) \, (\lambda_t^c)^{j-k} \, [(j-k)!]^{-1} P(Y_{t-1} = i \mid \mathcal{X}_t^c) I_{((j-k)=(0,1,....))} \\
&= \sum_{i=0}^{\infty} \sum_{k=0}^{\min(i,j)} \binom{i}{k} \alpha^k (1-\alpha)^{i-k} \exp(-\lambda_t^c) \, (\lambda_t^c)^{j-k} \, [(j-k)!]^{-1} p_{t-1,i} \; I_{((j-k)=(0,1,....))},
\end{aligned} \tag{4.4.3}$$

where $I_{(\cdot)}$ is the indicator function.
Here, the marginal mean and the marginal variance are given by

$$E(Y_n \mid \mathcal{X}_n^c) = \alpha^{n-1}\lambda_1^c + \alpha^{n-2}\lambda_2^c + ... + \lambda_n^c, \tag{4.4.4}$$

and

$$V(Y_n \mid \mathcal{X}_n^c) = \alpha^{n-1}\lambda_1^c + \alpha^{n-2}\lambda_2^c + ... + \lambda_n^c. \tag{4.4.5}$$

Result 4.2 *Under the above setup, the auto-covariance function (ACVF) of Y_t given $\mathcal{X}_{t+h}^c$ using the equation $Y_{t+h} = \alpha^h \circ Y_t + \sum_{i=1}^h \alpha^{h-i} \circ \varepsilon_{t+i}$ can be derived as*

$$\gamma_y(h) = Cov(Y_t, Y_{t+h} \mid \mathcal{X}_{t+h}^c) = \alpha^h \left(\alpha^{t-1}\lambda_1^c + \alpha^{t-2}\lambda_2^c + ... + \lambda_t^c\right).$$

The derivation of this result is presented in Appendix (see Section 4.10).
Hence, for $h \neq 0$, the ACF can be derived as follows:

$$\begin{aligned}
\rho_y(h) &= \frac{Cov(Y_t, Y_{t+h} \mid \mathcal{X}_{t+h}^c)}{\sqrt{Var(Y_t \mid \mathcal{X}_{t+h}^c)\, Var(Y_{t+h} \mid \mathcal{X}_{t+h}^c)}} \\
&= \frac{\alpha^h \left(\alpha^{t-1}\lambda_1^c + \alpha^{t-2}\lambda_2^c + ... + \lambda_t^c\right)}{\sqrt{\left(\alpha^{t-1}\lambda_1^c + ... + \lambda_t^c\right)\left(\alpha^{t+h-1}\lambda_1^c + ... + \lambda_{t+h}^c\right)}} \\
&= \alpha^h \sqrt{\left(\frac{\alpha^{t-1}\lambda_1^c + ... + \lambda_t^c}{\alpha^{t+h-1}\lambda_1^c + ... + \lambda_{t+h}^c}\right)}.
\end{aligned}$$

The above expression decays exponentially to 0 as h goes to ∞ for $\alpha \in (0,1)$ and the restrictions on β_i^c's.

4.5 Forecasting properties

4.5.1 h-step ahead forecasting distribution

To find the h-step ahead forecasting distribution, we use the following recursive method:

$$\begin{aligned}
Y_{n+h} &= \alpha \circ Y_{n+h-1} + \varepsilon_{n+h} \\
&= \alpha \circ \{\alpha \circ Y_{n+h-2} + \varepsilon_{n+h-1}\} + \varepsilon_{n+h} \\
&\cdots \\
&= \alpha^h \circ Y_n + \sum_{i=1}^{h} \alpha^{h-i} \circ \varepsilon_{n+i}.
\end{aligned}$$

Hence, the h-step ahead conditional mean and conditional variance can be given as

$$E(Y_{n+h} \mid Y_n, \mathcal{X}_{n+h}^c) = \alpha^h Y_n + \sum_{i=1}^{h} \alpha^{h-i}\lambda_{n+i}^c,$$

and

$$V(Y_{n+h} \mid Y_n, \mathcal{X}^c_{n+h}) = \alpha^h \left(1 - \alpha^h\right) Y_n + \sum_{i=1}^{h} \alpha^{h-i} \lambda^c_{n+i}.$$

The h-step ahead forecasting distribution of PINAR(1) process was derived by Freeland and McCabe, 2004a using the binomial thinning operator (discussed by Al-Osh and Alzaid, 1987) and it turned out to be a convolution of binomial and Poisson distributions. Here, we can calculate the conditional PGF of Y_{n+h} given Y_n and $\mathcal{X}^c_{n+h}$ and then derive the forecasting distribution using this.

Result 4.3 *The conditional PGF of Y_{n+h} given Y_n and $\mathcal{X}^c_{n+h}$ can be shown as*

$$\Phi_{Y_{n+h}|Y_n,\mathcal{X}^c_{n+h}}(s) = \left(1 - \alpha^h + \alpha^h s\right)^{Y_n} \left[\exp\left(-\sum_{i=1}^{h} \lambda^c_{n+i}\, \alpha^{h-i}(1-s)\right)\right].$$

The derivation of this result is presented in Appendix (see Section 4.10).

Corollary 4.1 *From the above result, we can say that the h-step ahead prediction distribution of Y_{n+h} given Y_n and $\mathcal{X}^c_{n+h}$ is a convolution of $Bin\left(Y_n, \alpha^h\right)$ and some random variable Z_{n+h} having the PGF of the form $\left[\exp\left(-\sum_{i=1}^{h} \lambda^c_{n+i}\, \alpha^{h-i}(1-s)\right)\right]$. Therefore, Z_{n+h} follows Poisson distribution with mean $\left(\sum_{i=1}^{h} \lambda^c_{n+i}\, \alpha^{h-i}\right)$.*
Thus, the prediction distribution can be presented as

$$Y_{n+h} \mid Y_n, \mathcal{X}^c_{n+h} \overset{d}{=} Bin(Y_n, \alpha^h) * Z_{n+h},$$

where $*$ is called the convolution between two distributions.

Corollary 4.2 ***Result 4.4*** *Using Corollary 4.1, the h-step ahead forecasting distribution of Y_{n+h} given Y_n and $\mathcal{X}^c_{n+h}$ can be derived as*

$$p_h(j_1 \mid j_2) = \sum_{q=0}^{\min(j_1,j_2)} \binom{j_2}{q} \alpha^{qh} (1-\alpha^h)^{j_2-q} \exp(-\lambda^c(h)) \frac{(\lambda^c(h))^{j_1-q}}{(j_1-q)!} I_{(0,1,..)}(j_1-q), \tag{4.5.1}$$

where $I_{(.)}$ is the indicator function, $\lambda^c(h) = \left(\sum_{i=1}^{h} \alpha^{h-i} \lambda^c_{n+i}\right)$, and $p_h(k_1 \mid k_2) = P(Y_{n+h} = k_1 \mid Y_n = k_2, \mathcal{X}^c_{n+h})$.
The derivation of this result is presented in Appendix (see Section 4.10).

In our study, We consider $\hat{Y}_{t+h}$, the mean of the estimated h-step ahead forecasting distribution of Y_{t+h} given Y_t and $\mathcal{X}_{t+h}$, mentioned in Result 4.4.

4.6 Parameter Estimation

Conditional least squares estimation

Conditional least squares estimation is generally considered for estimating the regression parameters of the model in the context of time series models. Freeland and McCabe, 2004a; Freeland and McCabe, 2005 used this approach for PINAR(1) process.

In order to perform the conditional least squares estimation method, we need to minimize the sum of squared deviation about the conditional expectation, which is given as $Q^*(\boldsymbol{\beta}^c) = \sum_{t=2}^{n}[Y_t - E(Y_t \mid Y_{t-1}, \mathcal{X}_t^c)]^2$ instead of $Q(\boldsymbol{\beta}^c) = \sum_{t=2}^{n}[Y_t - E(Y_t \mid \mathcal{X}_t^c)]^2$ with respect to the regression parameters of the model, where $E(Y_t \mid Y_{t-1}, \mathcal{X}_t^c) = \alpha Y_{t-1} + \lambda_t^c$ and $\boldsymbol{\beta}^c$ is the vector for regression parameters. Here, numerical methods are being employed to obtain the CLS estimates of the regression parameters of the model as there are no closed forms of the CLS estimators. We use the "constrOptim()" function in R software for this purpose.

In the subsequent section, some extensive simulation studies for both one change-point and two change-points are examined. From the simulation results, we establish the consistency of the CLS method.

4.7 Simulation study

4.7.1 The setup

In this section, we conduct extensive simulation studies for (i) the estimation of model parameters (empirical consistency of the estimated parameters) and (ii) the forecasting performances of the proposed model. The simulation experiments are performed for varying sample sizes along with different choices of model parameters and tuning parameter for both one change-point and two change-point study.

For the simulation studies for the analysis of one change-point (t_{ch}), $\mathcal{X}_n^{c1}$ (the set of all covariates up to time-point n for one change-point study) is equal to $\{1, Z_1, P_1, Z_2, P_2, \ldots, Z_n, P_n\}$, where $Z_t = \frac{(t-t_{ch})\exp(\delta_n(t-t_{ch}))}{1+\exp(\delta_n(t-t_{ch}))}$, which is the smooth time-varying component and $P_t = t$, which is the simple time-varying component.

For the simulation studies regarding the analysis of two change-points (t_{ch1} and t_{ch2}), $\mathcal{X}_n^{c2} = \{1, Z_{11}, Z_{21}, Q_1, \ldots, Z_{1n}, Z_{2n}, Q_n\}$, where $Z_{1t} = \frac{(t-t_{ch1})\exp(\delta_n(t-t_{ch1}))}{1+\exp(\delta_n(t-t_{ch1}))}$, $Z_{2t} = \frac{(t-t_{ch2})\exp(\delta_n(t-t_{ch2}))}{1+\exp(\delta_n(t-t_{ch2}))}$ and $Q_t = t$; here, Z_{1t}'s and Z_{2t}'s are the smooth time-varying components and Q_t's are the simple time-varying components.

In the simulation experiments, we use these components for each study to generate datasets of varying sample sizes by the data generating processes mentioned in equation (4.3.2) for one change-point and equation (4.3.3) for two change-points.

In the simulation experiments for forecasting performances, we compare our proposed model to the following model

$$Y_t = \alpha \circ Y_{t-1} + \varepsilon_t. \tag{4.7.1}$$

Here, ε_t follows Poisson(λ_t^{c0}), where λ_t^{c0} is assumed to have the form

$$\lambda_t^{c0} = \exp\left(\beta_0^{c0} + \beta_1^{c0} t\right).$$

This model involves no change-point. But the innovation terms depend on time-varying covariates.

4.7.2 Empirical consistency

We perform a simulation study to examine the consistency of the estimation method, used in the proposed model. To conduct this simulation study, we simulate data from the proposed model with (1) one change-point (given in equation (4.3.2)), and (2) two change-points (given in equation (4.3.3)). For each of the two data generating cases listed above, three sets of regression parameters are considered each. These values are mentioned in the following sections. The kind of data that we have considered in our study, like Italy COVID-19 data or Kerala COVID-19 data, total data-points of 500 can be considered as large sample size. Based on this, the simulation studies regarding the sample size of 500 provide us the properties of large sample size. Therefore, four distinct sample sizes (n) of 100, 200, 300, and 500 are investigated in this experiment. Throughout the simulation study, we consider three different values of δ_n: 0.1, 0.5, and 1. All simulation results are based on 1000 Monte Carlo replications.

Case 1: Analysis of one change-point

For one change-point simulation study, we consider the change-point t_{ch} as $0.4n$, where n is the sample size of the data. Three sets of regression parameters used in the data generating process are $\boldsymbol{\beta^{c1}} = (\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1}) = (0.4, -0.1, -0.1, 0.05)$, $(0.5, 0.1, -0.15, 0.04)$ and $(0.8, -0.4, -0.09, 0.04)$. For each set of regression parameters and each tuning parameter δ_n, we simulate the data using model (4.3.1) with λ_t^{c1} given in equation (4.3.2). Here, we employ the CLS estimation method.

Here, for the data generating method of one change-point, $\mathcal{X}_n^{c1}$, set of all covariates up to time-point n, contains both the smooth time-varying components and the simple time-varying components up to time-point n as described in Section 4.7.1, where n is the sample size of the simulated dataset. The process is repeated for 1000 times and we report the mean estimates and MSEs of the regression parameters in Tables 4.1, 4.2 and 4.3. From Tables 4.1, 4.2 and 4.3, we can see that as the sample size increases MSE of the estimated regression parameters decreases. This empirically establishes the consistency of the CLS estimation.

TABLE 4.1: Mean estimates of the regression parameters $\boldsymbol{\beta^{c1}}$ with their respective MSEs for different sample sizes, where the data generating process is the proposed method with one change-point and true $\boldsymbol{\beta^{c1}} = (0.4, -0.1, -0.1, 0.05)$

$\delta_n = 0.1$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.3307(0.0170)	−0.0819(0.1403)	−0.1028(2e-04)	0.0520(1e-04)
200	0.3425(0.0141)	−0.0595(0.0495)	−0.1004(0e+00)	0.0505(0e+00)
300	0.3463(0.0114)	−0.0613(0.0229)	−0.1000(0e+00)	0.0503(0e+00)
500	0.3960(0.0006)	−0.0991(0.0008)	−0.1000(0e+00)	0.0500(0e+00)
$\delta_n = 0.5$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.3302(0.0176)	−0.0553(0.1334)	−0.1019(3e-04)	0.0513(1e-04)
200	0.3432(0.0133)	−0.0591(0.0547)	−0.1004(0e+00)	0.0505(0e+00)
300	0.3581(0.0084)	−0.0618(0.0219)	−0.0999(0e+00)	0.0502(0e+00)
500	0.3988(0.0003)	−0.0998(0.0008)	−0.1000(0e+00)	0.0500(0e+00)
$\delta_n = 1$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.3285(0.0194)	−0.0599(0.1403)	−0.1019(3e-04)	0.0513(1e-04)
200	0.3457(0.0129)	−0.0610(0.0531)	−0.1003(0e+00)	0.0504(0e+00)
300	0.3609(0.0091)	−0.0712(0.0205)	−0.1000(0e+00)	0.0502(0e+00)
500	0.3977(0.0003)	−0.0989(0.0008)	−0.1000(0e+00)	0.0500(0e+00)

TABLE 4.2: Mean estimates of the regression parameters $\boldsymbol{\beta^{c1}}$ with their respective MSEs for different sample sizes, where the data generating process is the proposed method with one change-point and true $\boldsymbol{\beta^{c1}} = (0.5, 0.1, -0.15, 0.04)$

$\delta_n = 0.1$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.3929(0.0274)	0.1776(0.1566)	−0.1554(1e-03)	0.0424(1e-04)
200	0.4184(0.0192)	0.1887(0.0769)	−0.1490(1e-04)	0.0406(0e+00)
300	0.4319(0.0143)	0.1744(0.0396)	−0.1486(0e+00)	0.0403(0e+00)
500	0.4770(0.0026)	0.1154(0.0043)	−0.1494(0e+00)	0.0401(0e+00)
$\delta_n = 0.5$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.3998(0.0253)	0.1761(0.4487)	−0.1603(0.0123)	0.0422(3e-04)
200	0.4298(0.0167)	0.1757(0.0771)	−0.1489(0.0002)	0.0405(0e+00)
300	0.4483(0.0105)	0.1563(0.0387)	−0.1487(0.0000)	0.0402(0e+00)
500	0.4935(0.0008)	0.1052(0.0033)	−0.1498(0.0000)	0.0400(0e+00)
$\delta_n = 1$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.4076(0.0239)	0.1735(0.1509)	−0.1594(0.0122)	0.0421(1e-04)
200	0.4301(0.0160)	0.1871(0.0773)	−0.1484(0.0002)	0.0403(0e+00)
300	0.4482(0.0100)	0.1563(0.0382)	−0.1485(0.0000)	0.0403(0e+00)
500	0.4954(0.0008)	0.1061(0.0035)	−0.1498(0.0000)	0.0400(0e+00)

TABLE 4.3: Mean estimates of the regression parameters $\boldsymbol{\beta^{c1}}$ with their respective MSEs for different sample sizes, where the data generating process is the proposed method with one change-point and true $\boldsymbol{\beta^{c1}} = (0.8, -0.4, -0.09, 0.04)$

$\delta_n = 0.1$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.6993(0.0174)	−0.2880(0.4926)	−0.0975(0.0031)	0.0462(4e-04)
200	0.7309(0.0095)	−0.2891(0.1230)	−0.0891(0.0001)	0.0418(0e+00)
300	0.7587(0.0041)	−0.3481(0.0468)	−0.0890(0.0000)	0.0409(0e+00)
500	0.7969(0.0001)	−0.3997(0.0038)	−0.0899(0.0000)	0.0401(0e+00)
$\delta_n = 0.5$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.7057(0.0171)	−0.3245(0.9243)	−0.1032(0.0168)	0.0464(7e-04)
200	0.7430(0.0078)	−0.2892(0.1310)	−0.0899(0.0005)	0.0413(0e+00)
300	0.7777(0.0022)	−0.3645(0.0448)	−0.0894(0.0000)	0.0404(0e+00)
500	0.7989(0.0001)	−0.4007(0.0044)	−0.0900(0.0000)	0.0400(0e+00)
$\delta_n = 1$				
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c1}}$ (MSE)	$\hat{\beta_1^{c1}}$ (MSE)	$\hat{\beta_2^{c1}}$ (MSE)
100	0.7044(0.0170)	−0.3072(0.9083)	−0.0952(0.0018)	0.0463(7e-04)
200	0.7440(0.0075)	−0.2849(0.1281)	−0.0891(0.0002)	0.0411(0e+00)
300	0.7726(0.0028)	−0.3560(0.0498)	−0.0893(0.0000)	0.0405(0e+00)
500	0.7989(0.0001)	−0.4011(0.0045)	−0.0900(0.0000)	0.0400(0e+00)

Case 2: Analysis of two change-points

For two change-point simulation study, the change-points t_{ch1} and t_{ch2} are considered $0.3n$ and $0.6n$, respectively. Three sets of regression parameters, used in the data generating process, are $\boldsymbol{\beta^{c2}} = (\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2}) = (0.4, -0.8, -0.1, 0.05, 0.07)$, $(0.5, 0.1, -0.08, 0.04, 0.06)$ and $(0.8, -0.8, -0.08, 0.04, 0.06)$. For each set of regression parameters and each tuning parameter δ_n, we simulate the data using model (4.3.1) with λ_t^{c2} given in equation (4.3.3). Here, for the data generating method of two change-points, $\mathcal{X}_n^{c2}$, set of all covariates up to time-point n, contains both the smooth time-varying components and the simple time-varying components up to time-point n as described in Section 4.7.1. The process is repeated for 1000 times and we report the mean estimates and MSEs of the regression parameters in Tables 4.4, 4.5 and 4.6. From Tables 4.4, 4.5 and 4.6, we can see that as the sample size increases MSE of the estimated regression parameters decreases. This empirically establishes the consistency of the CLS estimation.

TABLE 4.4: Mean estimates of the regression parameters $\boldsymbol{\beta^{c2}}$ with their respective MSEs for different sample sizes, where the data generating process is the proposed method with two change-points and true $\boldsymbol{\beta^{c2}} = (0.4, -0.8, -0.1, 0.05, 0.07)$

$\delta_n = 0.1$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.3226(0.0169)	−0.8269(0.4180)	−0.1038(9e-04)	0.0498(3e-04)	0.0739(6e-04)
200	0.3507(0.0084)	−0.7827(0.0952)	−0.1010(0e+00)	0.0500(0e+00)	0.0709(0e+00)
300	0.3620(0.0058)	−0.7864(0.0243)	−0.1004(0e+00)	0.0499(0e+00)	0.0705(0e+00)
500	0.3967(0.0004)	−0.8010(0.0009)	−0.1000(0e+00)	0.0500(0e+00)	0.0700(0e+00)
$\delta_n = 0.5$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.3257(0.0160)	−0.9088(3.1633)	−0.1076(0.0039)	0.0502(3e-04)	0.0772(0.0036)
200	0.3546(0.0082)	−0.7675(0.0995)	−0.1005(0.0000)	0.0499(0e+00)	0.0705(0.0000)
300	0.3687(0.0052)	−0.7872(0.0257)	−0.1003(0.0000)	0.0499(0e+00)	0.0704(0.0000)
500	0.3981(0.0003)	−0.7999(0.0010)	−0.1000(0.0000)	0.0500(0e+00)	0.0700(0.0000)
$\delta_n = 1$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.3236(0.0166)	−0.8356(0.6893)	−0.1041(0.0013)	0.0491(3e-04)	0.0744(9e-04)
200	0.3598(0.0070)	−0.7774(0.0934)	−0.1005(0.0000)	0.0499(0e+00)	0.0706(0e+00)
300	0.3666(0.0055)	−0.7934(0.0254)	−0.1005(0.0000)	0.0499(0e+00)	0.0705(0e+00)
500	0.3989(0.0003)	−0.8014(0.0010)	−0.1000(0.0000)	0.0500(0e+00)	0.0700(0e+00)

TABLE 4.5: Mean estimates of the regression parameters $\boldsymbol{\beta^{c2}}$ with their respective MSEs for different sample sizes, where the data generating process is the proposed method with two change-points and true $\boldsymbol{\beta^{c2}} = (0.5, 0.1, -0.08, 0.04, 0.06)$

$\delta_n = 0.1$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.4189(0.0158)	0.1390(0.2120)	−0.0828(4e-04)	0.0395(1e-04)	0.0629(3e-04)
200	0.4478(0.0080)	0.1387(0.0497)	−0.0808(0e+00)	0.0399(0e+00)	0.0608(0e+00)
300	0.4712(0.0039)	0.1104(0.0189)	−0.0804(0e+00)	0.0400(0e+00)	0.0604(0e+00)
500	0.4968(0.0004)	0.1066(0.0013)	−0.0800(0e+00)	0.0400(0e+00)	0.0600(0e+00)
$\delta_n = 0.5$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.4238(0.0161)	0.1336(0.1715)	−0.0834(4e-04)	0.0404(1e-04)	0.0630(2e-04)
200	0.4601(0.0070)	0.1248(0.0533)	−0.0807(0e+00)	0.0400(0e+00)	0.0607(0e+00)
300	0.4672(0.0047)	0.1284(0.0197)	−0.0803(0e+00)	0.0400(0e+00)	0.0603(0e+00)
500	0.4967(0.0003)	0.1032(0.0012)	−0.0800(0e+00)	0.0400(0e+00)	0.0600(0e+00)
$\delta_n = 1$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.4225(0.0161)	0.1489(0.1687)	−0.0825(4e-04)	0.0400(1e-04)	0.0624(2e-04)
200	0.4522(0.0080)	0.1376(0.0520)	−0.0807(0e+00)	0.0400(0e+00)	0.0607(0e+00)
300	0.4695(0.0039)	0.1151(0.0185)	−0.0805(0e+00)	0.0400(0e+00)	0.0600(0e+00)
500	0.4981(0.0003)	0.1043(0.0014)	−0.0800(0e+00)	0.0400(0e+00)	0.0600(0e+00)

TABLE 4.6: Mean estimates of the regression parameters $\boldsymbol{\beta}^{c2}$ with their respective MSEs for different sample sizes, where the data generating process is the proposed method with two change-points and true $\boldsymbol{\beta}^{c2} = (0.8, -0.8, -0.08, 0.04, 0.06)$

$\delta_n = 0.1$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.6897(0.0195)	−0.8378(1.5343)	−0.0934(0.0023)	0.0395(7e-04)	0.0735(0.0018)
200	0.7506(0.0052)	−0.7839(0.1552)	−0.0826(0.0001)	0.0395(0e+00)	0.0629(0.0001)
300	0.7816(0.0011)	−0.8459(0.0450)	−0.0813(0.0000)	0.0399(0e+00)	0.0614(0.0000)
500	0.7989(0.0000)	−0.8067(0.0035)	−0.0801(0.0000)	0.0400(0e+00)	0.0601(0.0000)
$\delta_n = 0.5$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.6921(0.0196)	−0.8450(9.4422)	−0.0927(0.0111)	0.0376(8e-04)	0.0735(0.0105)
200	0.7552(0.0047)	−0.7630(0.1680)	−0.0820(0.0001)	0.0395(0e+00)	0.0622(0.0001)
300	0.7854(0.0009)	−0.8288(0.0464)	−0.0810(0.0000)	0.0399(0e+00)	0.0610(0.0000)
500	0.7995(0.0000)	−0.8047(0.0036)	−0.0800(0.0000)	0.0400(0e+00)	0.0600(0.0000)
$\delta_n = 1$					
n	$\hat{\alpha}$ (MSE)	$\hat{\beta_0^{c2}}$ (MSE)	$\hat{\beta_1^{c2}}$ (MSE)	$\hat{\beta_2^{c2}}$ (MSE)	$\hat{\beta_3^{c2}}$ (MSE)
100	0.6945(0.0185)	−0.7970(1.8192)	−0.0913(0.0037)	0.0390(0.0035)	0.0716(0.0023)
200	0.7540(0.0046)	−0.7715(0.1508)	−0.0821(0.0001)	0.0392(0.0000)	0.0624(0.0001)
300	0.7846(0.0009)	−0.8152(0.0447)	−0.0808(0.0000)	0.0399(0.0000)	0.0609(0.0000)
500	0.7992(0.0000)	−0.8039(0.0034)	−0.0800(0.0000)	0.0400(0.0000)	0.0601(0.0000)

4.7.3 Forecasting

Another simulation study is performed to study the h-step ahead forecasting performance of the proposed predure for varying h, compared to the comparison method mentioned in equation (4.7.1). For comparison, we look at the PRMSE(h) measure (see Section 3.7 of Chapter 3). Each study considers two sets of regression parameters: (i) one change-point and (ii) two change-points. Throughout the simulation study, we consider three different values of δ_n: 0.1, 0.5, and 1. Each time, we generate a total sample of size 100 of which a training set of size 90 is used to fit the two models under consideration for comparison, and a test set of size 10 is used to calculate PRMSE(h) for $h = 1, 2, 3, 4$. This procedure is repeated 1000 times.

Case 1: Analysis of one change-point

For this simulation study, we consider the change-point t_{ch} as $0.4n$. Two sets of regression parameters, used in the data generating process, are $\boldsymbol{\beta^{c1}}=(\alpha, \beta_0^{c1}, \beta_1^{c1}, \beta_2^{c1})$ $= (0.4, 1, -0.1, 0.06)$ and $(0.6, -0.1, -0.12, 0.08)$. For each set of regression parameters and each tuning parameter δ_n, we simulate the data using model (4.3.1) with λ_t^{c1} given in equation (4.3.2). Here, for the data generating method of one change-point, $\mathcal{X}_n^{c1}$, set of all covariates up to time-point n, contains both the smooth time-varying components and the simple time-varying components up to time-point n as described in Section 4.7.1. The process is repeated for 1000 times. We report the h-step ahead forecasting performances for both the proposed model and the comparison model for $h = 1, 2, 3, 4$ in Tables 4.7 and 4.8. From the tables, we mostly notice the average PRMSE(h) of the proposed process is relatively smaller than that of the comparison process.

TABLE 4.7: PRMSE(h) values for varying h for different δ_n, where the data generating process is the proposed method with one change-point and true $\boldsymbol{\beta^{c1}} = (0.4, 1, -0.1, 0.06)$

	$\delta_n = 0.1$		$\delta_n = 0.5$		$\delta_n = 1$	
h	Proposed	Comparison	Proposed	Comparison	Proposed	Comparison
1	2.4297	2.6121	2.3828	2.5327	2.3466	2.5185
2	2.6103	3.1817	2.5710	3.0654	2.5191	3.0331
3	2.5760	3.4210	2.5497	3.2739	2.4956	3.1700
4	2.5040	3.5396	2.4987	3.4671	2.4492	3.2973

TABLE 4.8: PRMSE(h) values for varying h for different δ_n, where the data generating process is the proposed method with one change-point and true $\boldsymbol{\beta^{c1}} = (0.6, -0.1, -0.12, 0.08)$

	$\delta_n = 0.1$		$\delta_n = 0.5$		$\delta_n = 1$	
h	Proposed	Comparison	Proposed	Comparison	Proposed	Comparison
1	2.2816	2.2588	2.2228	2.2245	2.2013	2.2059
2	2.6797	2.8975	2.6321	2.8400	2.5959	2.8105
3	2.7467	3.2576	2.7236	3.1907	2.6925	3.1557
4	2.7135	3.4815	2.7188	3.3835	2.6657	3.3188

Case 2: Analysis of two change-points

Here, we consider the change-points t_{ch1} and t_{ch2} as $0.3n$ and $0.6n$, respectively. Two sets of regression parameters are used here. Those are $\boldsymbol{\beta^{c2}}=(\alpha, \beta_0^{c2}, \beta_1^{c2}, \beta_2^{c2}, \beta_3^{c2})$ $= (0.3, -2.5, -0.2, 0.1, 0.15)$ and $(0.6, -2, -0.15, 0.07, 0.12)$. For each set of regression parameters and each tuning parameter δ_n, we simulate the data using model

(4.3.1) with λ_t^{c2} given in equation (4.3.3). Here, $\mathcal{X}_n^{c2}$ is the set of all covariates up to time-point n, which contains both the smooth time-varying components and the simple time-varying components up to time-point n as described in Section 4.7.1. The process is repeated for 1000 times. We report the h-step ahead forecasting performances for both the proposed model and the comparison model for $h = 1, 2, 3, 4$ in Tables 4.9 and 4.10. From the tables, we mostly notice the average PRMSE(h) of the proposed process is relatively smaller than that of the comparison process.

TABLE 4.9: PRMSE(h) values for varying h for different δ_n, where the data generating process is the proposed method with two change-points and true $\boldsymbol{\beta}^{c2} = (0.3, -2.5, -0.2, 0.1, 0.15)$

	$\delta_n = 0.1$		$\delta_n = 0.5$		$\delta_n = 1$	
h	Proposed	Comparison	Proposed	Comparison	Proposed	Comparison
1	4.4999	4.4436	4.6278	4.6514	4.6139	4.5731
2	4.8700	5.8125	4.9868	5.8791	4.9850	5.7521
3	4.8929	6.7093	5.0139	6.7937	4.9635	6.6499
4	4.8889	7.3429	5.0029	7.4148	4.8871	7.3061

TABLE 4.10: PRMSE(h) values for varying h for different δ_n, where the data generating process is the proposed method with two change-points and true $\boldsymbol{\beta}^{c2} = (0.6, -2, -0.15, 0.07, 0.12)$

	$\delta_n = 0.1$		$\delta_n = 0.5$		$\delta_n = 1$	
h	Proposed	Comparison	Proposed	Comparison	Proposed	Comparison
1	4.3435	4.1632	4.4185	4.2732	4.4701	4.2537
2	5.3703	5.6435	5.4791	5.7448	5.5107	5.6586
3	5.7873	6.7474	5.9081	6.6270	5.9717	6.6829
4	6.0024	7.3965	5.9931	7.2352	6.0607	7.4008

4.8 Data analyses

4.8.1 COVID-19 data of Italy

In this section, we analyze the Italy COVID-19 dataset (see Section 4.2.1) through our proposed method. We also fit the comparison model (see equation (4.7.1)) to this dataset. We consider $n = 113$ data-points from February 15 to June 6. Here, $\mathcal{X}_n^{c1}$ is the set of all time-varying covariates up to time-point n containing both the smooth time-varying covariates that have the change-point and the simple time-varying covariates up to n time-points as described in Section 4.7.1. From

the plot of daily time series data, we notice that there is only one change-point during that period and hence we fit the proposed model with one change-point (see equation (4.3.2)).

For the proposed model with one change-point, the change-point t_{ch} for the COVID-19 data of Italy is estimated using the method described in Section 4.3.4 from the data of daily new cases. The value of estimated change-point is 44. In order to estimate the optimal δ_n, we consider a set of points in the interval [0.1, 10] with an increment of 0.1. Previously, we observe that the values of δ_n around 0.1 can capture the curvature that is similar to that of this dataset of Italy. Hence, we take this interval of [0.1,10]. For each of the δ_n in the set, we fit our proposed model to the data. For every fit, we calculate the goodness-of-fit measure, namely Root Mean Squared Error (RMSE). Then, we consider the minimum value of this measure to obtain the optimal δ_n. We observe that $\delta_n = 0.1$ gives the minimum value of RMSE, which is 938.06. So, the value of δ_n is taken to be 0.1 for this study.

For this dataset, the estimates of the regression parameters of our proposed model by CLS method are $(\hat{\alpha}, \hat{\beta_0^{c1}}, \hat{\beta_1^{c1}}, \hat{\beta_2^{c1}}) = (0.9657, 5.4430, -0.1149, 0.0742)$, and that of the comparison model are $(\hat{\alpha}_{cls}, \hat{\beta_0^{c0}}, \hat{\beta_1^{c0}}) = (0.9925, 7.5794, -0.0155)$. The RMSE corresponding to our proposed model for the dataset is 938.06, which is much lower compared to that for the comparison model, which is 1940.96. In Figure 4.7, we provide the plot of RMSEs against each of δ_n's in the set [0.1, 10]. Overall, we can say that the fit of our proposed model is satisfactory.

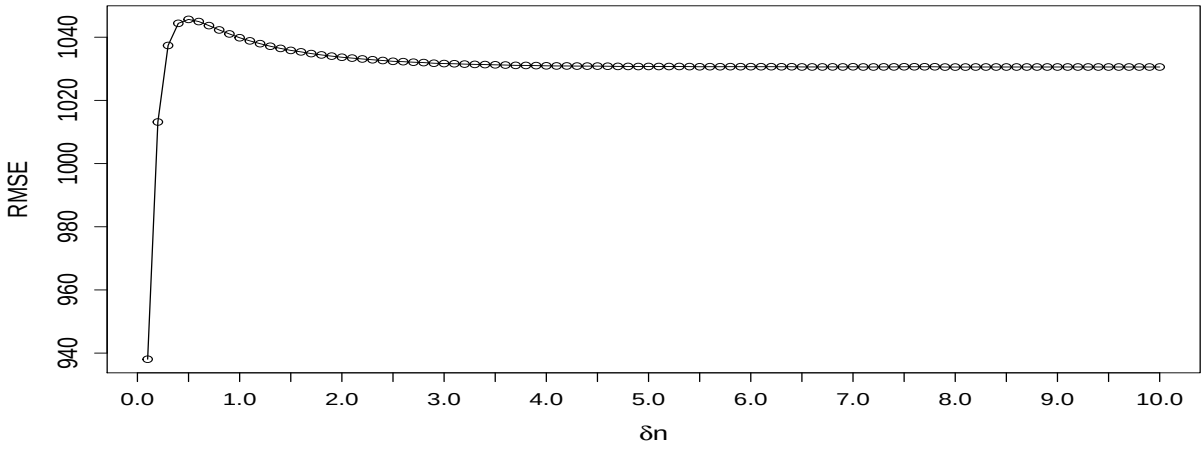


FIGURE 4.7: Plot of δ_n vs RMSE for Italy data

To study the forecasting performance, we partition the data into two sets: (i) the training set containing the first 101 observations is used to fit the models and (ii) the test set with the remaining 12 observations, is used for calculating PRMSE

measure for both models. For 1-step ahead forecasting ($h = 1$), the PRMSE values for the proposed model and the comparison model are 872.08 and 1925.25, respectively. For 2-step ahead forecasting ($h = 2$), the PRMSE values for the proposed model and the comparison model are 1560.44 and 3748.89, respectively. So, for both the one-step and two-step ahead forecasting results, our proposed model performs better than the comparison model.

4.8.2 COVID-19 data of Kerala

In this section, we analyze the Kerala COVID-19 dataset (see Section 4.2.2). we consider 90 data-points from March 9 to June 6. Here, $\mathcal{X}_n^{c2}$ is the set of all time-varying covariates up to time-point n containing both the smooth time-varying covariates that have the change-points and the simple time-varying covariates up to n time-points as described in Section 4.7.1, for $n = 90$. From the plot of daily time series data, we observe that there are two change-points during that period and hence we fit the proposed model with two change-points (see equation (4.3.3)).

The change-points t_{ch1} and t_{ch2} for this data are estimated through the method described in Section 4.3.4 from the data of daily new cases. The values of estimated change-points are 23 (first change-point) and 56 (second change-point). We estimate the optimal δ_n in the same way as we do for the Italy data. Here, $\delta_n = 0.3$ gives the minimum value of RMSE, which is 13.18. So, the value of δ_n is 0.3, for this study.

For this dataset, the estimates of the regression parameters of our proposed model by CLS method are $(\hat{\alpha}, \hat{\beta_0^{c2}}, \hat{\beta_1^{c2}}, \hat{\beta_2^{c2}}, \hat{\beta_3^{c2}}) = (0.8460, 1.3739, -0.1504, 0.1353, 0.1063)$. For the comparison model, those are $(\hat{\alpha}_{cls}, \hat{\beta_0^{c0}}, \hat{\beta_1^{c0}}) = (0.9921, -2.8131, 0.0811)$. The RMSE corresponding to our proposed model for the dataset is 13.18. The value is lower than the RMSE for the comparison model, which is 15.09. The plot of δ_n versus RMSEs for this analysis is provided in Figure 4.8. Overall, we can conclude that our proposed model fits well.

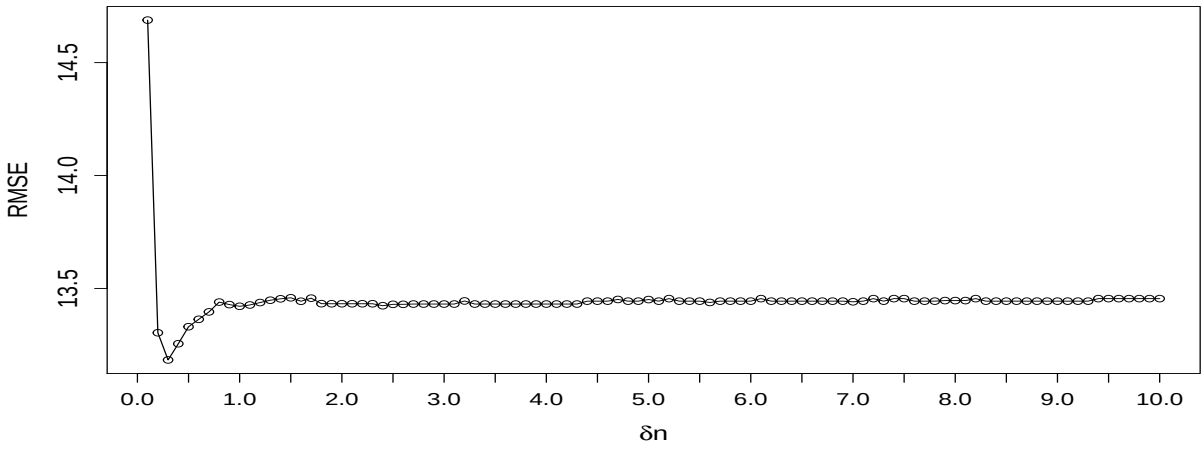


FIGURE 4.8: Plot of δ_n vs RMSE for Kerala data

For examining the forecasting performance, we partition the data into two sets: (i) the training set containing the first 78 observations is used to fit the models and (ii) the test set with the remaining 12 observations, is used for calculating PRMSE measure for both models. For 1-step ahead forecasting ($h = 1$), the PRMSE values for the proposed model and the comparison model are 168.69 and 186.99, respectively. For 2-step ahead forecasting ($h = 2$), the PRMSE values for the proposed model and the comparison model are 297.02 and 348.68, respectively. So, for both cases, our proposed model performs better than the comparison model.

4.9 Conclusions

The PINAR(1) process (see Al-Osh and Alzaid, 1987) has received significant attention for its simplicity and is widely used in the field of count time series data. However, this process cannot model count time series data, such as the COVID-19 data, which includes change-points and time-varying covariates. In this chapter, we have presented a new PINAR(1) model based on the binomial thinning operator that addresses the problem of change-point analysis using time-varying covariates. The idea behind our proposed model is inspired by Chan and Tong, 1986; Hansen, 2000 and Fong et al., 2017, who primarily worked with continuous data. The proposed model incorporates the Smooth Maximum, n.d. concept to create a smoothing change-point function that can capture changing curvatures in the data. The key feature of our proposed model is its ability to account for both change-points and time-varying covariates. As previously stated, we can see these characteristics in the COVID-19 datasets, which have inspired us

to create our proposed model for both single and two change-point studies. We have investigated the distributional forms of our proposed model and the h-step ahead forecasting distribution. We have employed the CLS estimation method. The simulation results have shown that the CLS estimation method is consistent when estimating parameters. From the data applications, we have seen that our proposed model outperformed the comparison model in terms of standard statistical measures such as RMSE. In both the simulation study and the data applications, our proposed model outperformed the comparison model for forecasting as measured by the accuracy metric PRMSE. Overall, we hope that our proposed model will be a viable option for modelling these types of count time series data.

4.10 Appendix

Appendix A. Proof of Result 4.1

We have

$$Y_n = \alpha \circ Y_{n-1} + \varepsilon_n = \sum_{i=1}^{Y_{n-1}} N_{1i} + \varepsilon_n,$$

where $\{N_{1i}\}$ is a sequence of iid Bernoulli(α) random variables, independent of Y_{n-1}, and we know that given $\mathcal{X}_n^c$, $\alpha \circ Y_{n-1}$ and ε_n are independent. So, we can write the PGF of $Y_n \mid \mathcal{X}_n^c$ as

$$\begin{aligned}
\Phi_{Y_n|\mathcal{X}_n^c}(s) &= E\left[s^{\left(\sum_{i=1}^{Y_{n-1}} N_{1i}+\varepsilon_n\right)} \mid \mathcal{X}_n^c\right] \\
&= E_{Y_{n-1}|\mathcal{X}_n^c} E\left[s^{\left(\sum_{i=1}^{Y_{n-1}} N_{1i}\right)} \mid Y_{n-1}, \mathcal{X}_n^c\right] \Phi_{\varepsilon_n|\mathcal{X}_n^c}(s) \\
&= E_{Y_{n-1}|\mathcal{X}_n^c}(1-\alpha+\alpha s)^{Y_{n-1}} \Phi_{\varepsilon_n|\mathcal{X}_n^c}(s) \\
&= \Phi_{Y_{n-1}|\mathcal{X}_n^c}(1-\alpha+\alpha s)\, \Phi_{\varepsilon_n|\mathcal{X}_n^c}(s).
\end{aligned}$$

So, we have the recursive formula

$$\Phi_{Y_n|\mathcal{X}_n^c}(s) = \Phi_{Y_{n-1}|\mathcal{X}_n^c}(1-\alpha+\alpha s)\, \Phi_{\varepsilon_n|\mathcal{X}_n^c}(s). \tag{4.10.1}$$

Putting $n = 2$ in equation (4.10.1), we have

$$\begin{aligned}
\Phi_{Y_2|\mathcal{X}_2^c}(s) &= \Phi_{Y_1|\mathcal{X}_2^c}(1-\alpha+\alpha s)\,\Phi_{\varepsilon_2|\mathcal{X}_2^c}(s) \\
&= \exp[-\lambda_1^c(1-(1-\alpha+\alpha s)) - \lambda_2^c(1-s)] \quad [\text{Since } Y_1 \mid \mathcal{X}_2^c \sim \text{Poisson}(\lambda_1^c)] \\
&= \exp[-\lambda_1^c\alpha(1-s) - \lambda_2^c(1-s)] \\
&= \exp[-(\alpha\lambda_1^c + \lambda_2^c)(1-s)].
\end{aligned}$$

Therefore,

$$Y_2 \mid \mathcal{X}_2^c \sim \text{Poisson}(\alpha\lambda_1^c + \lambda_2^c).$$

Now, for $n = (k-1)$, we assume that

$$Y_{k-1} \mid \mathcal{X}_{k-1}^c \sim \text{Poisson}\left(\alpha^{k-2}\lambda_1^c + \alpha^{k-3}\lambda_2^c + \ldots + \lambda_{k-1}^c\right).$$

For $n = k$,

$$\begin{aligned}
\Phi_{Y_k|\mathcal{X}_k^c}(s) &= \Phi_{Y_{k-1}|\mathcal{X}_k^c}(1-\alpha+\alpha s)\,\Phi_{\varepsilon_k|\mathcal{X}_k^c}(s) \\
&= \exp[-\left(\alpha^{k-2}\lambda_1^c + \alpha^{k-3}\lambda_2^c + \ldots + \lambda_{k-1}^c\right)(1-(1-\alpha+\alpha s)) - \lambda_k^c(1-s)] \\
&= \exp[-\left(\alpha^{k-1}\lambda_1^c + \alpha^{k-2}\lambda_2^c + \ldots + \alpha\lambda_{k-1}^c\right)(1-s) - \lambda_k^c(1-s)] \\
&= \exp[-\left(\alpha^{k-1}\lambda_1^c + \alpha^{k-2}\lambda_2^c + \ldots + \lambda_k^c\right)(1-s)].
\end{aligned}$$

So,

$$Y_k \mid \mathcal{X}_k^c \sim \text{Poisson}\left(\alpha^{k-1}\lambda_1^c + \alpha^{k-2}\lambda_2^c + \ldots + \lambda_k^c\right).$$

This completes the proof.

Appendix B. Proof of Result 4.2

The ACVF is given by

$$\begin{aligned}
\gamma_y(h) = Cov(Y_t, Y_{t+h} \mid \mathcal{X}^c_{t+h}) &= E(Y_t\, Y_{t+h} \mid \mathcal{X}^c_{t+h}) - E(Y_t \mid \mathcal{X}^c_{t+h})E(Y_{t+h} \mid \mathcal{X}^c_{t+h}) \\
&= E(Y_t\, E(Y_{t+h} \mid Y_t, \mathcal{X}^c_{t+h}) \mid \mathcal{X}^c_{t+h}) \\
&\qquad -E(Y_t \mid \mathcal{X}^c_{t+h})E(Y_{t+h} \mid \mathcal{X}^c_{t+h}) \\
&= E\left[Y_t \left(\alpha^h Y_t + \sum_{i=1}^{h} \alpha^{h-i}\lambda^c_{t+i}\right) \middle| \mathcal{X}^c_{t+h}\right] \\
&\qquad -E(Y_t \mid \mathcal{X}^c_{t+h})E(Y_{t+h} \mid \mathcal{X}^c_{t+h}) \\
&= \alpha^h E\left(Y_t^2 \mid \mathcal{X}^c_{t+h}\right) - E(Y_t \mid \mathcal{X}^c_{t+h}) \times \\
&\qquad \left[\alpha^{t+h-1}\lambda^c_1 + ... + \lambda^c_{t+h} - \alpha^{h-1}\lambda^c_{t+1} - ... - \lambda^c_{t+h}\right] \\
&= \alpha^h \left[V(Y_t \mid \mathcal{X}^c_{t+h}) + E^2(Y_t \mid \mathcal{X}^c_{t+h})\right] \\
&\qquad -E(Y_t \mid \mathcal{X}^c_{t+h})\left(\alpha^{t+h-1}\lambda^c_1 + ... + \alpha^h\lambda^c_t\right) \\
&= \alpha^h \left[(\alpha^{t-1}\lambda^c_1 + ... + \lambda^c_t)(1 + \alpha^{t-1}\lambda^c_1 + ... + \lambda^c_t)\right] \\
&\qquad -(\alpha^{t-1}\lambda^c_1 + ... + \lambda^c_t)\left(\alpha^{t+h-1}\lambda^c_1 + ... + \alpha^h\lambda^c_t\right) \\
&= \alpha^h \left(\alpha^{t-1}\lambda^c_1 + \alpha^{t-2}\lambda^c_2 + ... + \lambda^c_t\right).
\end{aligned}$$

This completes the proof.

Appendix C. Proof of Result 4.3

The conditional PGF is given by

$$\begin{aligned}
\Phi_{Y_{n+h}|Y_n,\mathcal{X}^c_{n+h}}(s) &= E\left(s^{\alpha^h \circ Y_n + \sum_{i=1}^{h} \alpha^{h-i} \circ \varepsilon_{n+i}} \mid Y_n, \mathcal{X}^c_{n+h}\right) \\
&= E\left(s^{\alpha^h \circ Y_n} \mid Y_n, \mathcal{X}^c_{n+h}\right) \prod_{i=1}^{h} E\left(s^{\alpha^{h-i} \circ \varepsilon_{n+i}} \mid \mathcal{X}^c_{n+h}\right).
\end{aligned}$$

Since $\alpha^h \circ Y_n \mid Y_n, \mathcal{X}^c_{n+h} \sim \text{Bin}\left(Y_n, \alpha^h\right)$, we can write $E\left(s^{\alpha^h \circ Y_n} \mid Y_n, \mathcal{X}^c_{n+h}\right) = \left(1 - \alpha^h + \alpha^h s\right)^{Y_n}$.

Now,

$$\begin{aligned}
\prod_{i=1}^{h} E\left(s^{\alpha^{h-i}\circ\varepsilon_{n+i}} \mid \mathcal{X}_{n+h}^{c}\right) &= \prod_{i=1}^{h} E_{\varepsilon_{n+i}|\mathcal{X}_{n+h}^{c}} E\left(s^{\alpha^{h-i}\circ\varepsilon_{n+i}} \mid \varepsilon_{n+i}, \mathcal{X}_{n+h}^{c}\right) \\
&= \prod_{i=1}^{h} E_{\varepsilon_{n+i}|\mathcal{X}_{n+h}^{c}} \left(1-\alpha^{h-i}+\alpha^{h-i}s\right)^{\varepsilon_{n+i}} \\
&= \prod_{i=1}^{h} \exp\left[-\lambda_{n+i}^{c}\left(1-\left(1-\alpha^{h-i}+\alpha^{h-i}s\right)\right)\right] \\
&= \prod_{i=1}^{h} \exp\left[-\lambda_{n+i}^{c}\,\alpha^{h-i}(1-s)\right] \\
&= \exp\left[-\sum_{i=1}^{h}\lambda_{n+i}^{c}\,\alpha^{h-i}(1-s)\right].
\end{aligned}$$

This completes the proof.

Appendix D. Proof of Result 4.4

The h-step ahead forecasting distribution is given by

$$\begin{aligned}
p_h(j_1 \mid j_2) &= P\left(\alpha^h \circ Y_n + \sum_{k=1}^{h}\alpha^{h-k}\circ\varepsilon_{n+k} = j_1 \,\middle|\, Y_n = j_2, \mathcal{X}_{n+h}^{c}\right) \\
&= \sum_{q=0}^{\min(j_1,j_2)} P\left(\alpha^h \circ Y_n = q \mid Y_n = j_2, \mathcal{X}_{n+h}^{c}\right) P\left(\sum_{k=1}^{h}\alpha^{h-k}\circ\varepsilon_{n+k} = j_1 - q \,\middle|\, \mathcal{X}_{n+h}^{c}\right) \\
&= \sum_{q=0}^{\min(j_1,j_2)} \binom{j_2}{q}\alpha^{hq}\left(1-\alpha^h\right)^{j_2-q}\exp\left[-\sum_{k=1}^{h}\alpha^{h-k}\lambda_{n+k}^{c}\right]\left(\sum_{k=1}^{h}\alpha^{h-k}\lambda_{n+k}^{c}\right)^{j_1-q} \\
&\qquad [(j_1-q)!]^{-1} I_{((j_1-q)=(0,1,\ldots))} \\
&= \sum_{q=0}^{\min(j_1,j_2)} \binom{j_2}{q}\left(\alpha^h\right)^{q}\left(1-\alpha^h\right)^{j_2-q}\exp(-\lambda^{c}(h))\,(\lambda^{c}(h))^{j_1-q} \\
&\qquad [(j_1-q)!]^{-1} I_{((j_1-q)=(0,1,\ldots))}.
\end{aligned}$$

This completes the proof.

Chapter 5

Analysis of count time series through INAR process with zero-inflation and seasonality

5.1 Introduction

In this chapter, we develop a PINAR process to model count time series data with zero-inflation and seasonality. In the first part of Weiß, 2018, count time series was studied in detail. The usual PINAR process based on the binomial thinning operator is very restrictive in the sense that its marginal mean and marginal variance are equal and hence cannot capture the overdispersion (including zero-inflation) or underdispersion problems. Furthermore, the usual PINAR process fails to capture many real-life count time series datasets that incorporate both zero-inflation and seasonality. However, in our study, we come across the data of weekly dengue cases in Kaohsiung City, Taiwan, from 2009 to 2012 (see Chan, Hu, and Hwang, 2015), which is an example of count time series data having both zero-inflation and seasonality properties.

To handle the overdispersed count time series data, some illustrious methods have been explored in the literature. Lambert, 1992; Al-Osh and Aly, 1992 proposed INAR models with geometric and negative binomial marginals, respectively (see, e.g., Maiti and Biswas, 2015b). Ristić, Bakouch, and Nastić, 2009; Ristić, Nastić, and Bakouch, 2012 studied new geometric and negative binomial INAR models based on the negative binomial thinning operator. Schweer and Weiß, 2014 proposed a compound Poisson INAR(1) process that can capture both

the overdispersion and undedispersion of the data. But these models are not sufficient to capture the overdispersion due to zero-inflation. In the i.i.d. setup, Lambert, 1992 proposed the zero-inflated Poisson distribution for modelling zero-inflated count data (also see Böhning et al., 1999; Angers and Biswas, 2003). In the context of the zero-inflated count time series data, Jazi, Jones, and Lai, 2012a; Wang, 2001; Porter and White, 2012; Maiti, Biswas, and Das, 2015; Bourguignon, 2018 proposed different types of count time series processes to capture the zero-inflation. For example, Jazi, Jones, and Lai, 2012a studied an INAR process with zero-inflated Poisson innovations, which has gained significant attention in this field. In this chapter, we develop the proposed zero-inflated PINAR process, borrowing the idea from the study of Jazi, Jones, and Lai, 2012a.

The standard de-seasonalizing methods used in the continuous time series framework do not work for handling seasonality in count time series, as the modified data may show negative values. One way to tackle this problem is to use the conditional Poisson regression models proposed by Cameron and Trivedi, 1986, Fokianos, 2011, and many others. Peng, Dominici, and Louis, 2006 proposed some kernel based smoothing Poisson regression to tackle such problems. Chattopadhyay et al., 2021 used some smoothing time-varying covariates in the INAR process to model the time-varying innovations. Using the ideas from these existing models, we incorporate the seasonality feature into the proposed model in our study. In order to do that, we consider two popular seasonal models used in the continuous setup, namely the seasonal model with sine-cosine functions and the seasonal dummy model (see, e.g., Lovell, 1963; Abeysinghe, 1994; Stolwijk, Straatman, and Zielhuis, 1999; Naumova et al., 2007; Ramanathan et al., 2020). Here, we study the connection between these two seasonal models. This connection provides the flexibility to use the seasonal model involving sine-cosine functions in modelling the innovations of our proposed process with a reduced set of model parameters. This cannot be done with the seasonal dummy model, where the full set of model parameters is required for the seasonal dummy variables to catch the seasonal structure of a time series data. Hence, working with the seasonal dummy model brings a lot of computational difficulties, which can make the estimation of model parameters inefficient. For the seasonal model with sine-cosine terms, it is best to use the full set of model parameters. However, because sine-cosine functions oscillate smoothly, our proposed model can still capture seasonality even with a reduced set of model parameters. This is discussed in detail in Section 5.3 in this chapter. We inherit this idea in modelling the innovations of our proposed INAR model to handle the seasonality feature. In this direction, some works have also been executed; see, e.g., Moriña et al.,

2011; Bourguignon et al., 2016; Buteikis and Leipus, 2020; Tian, Wang, and Cui, 2020; Prezotti-Filho et al., 2021.

These above-mentioned existing methods in the literature can be employed for modelling either the zero-inflation or the seasonal structure of a count time series data. However, to incorporate both of these features together in an INAR model to study a count time series, like the data of weekly dengue cases, we propose a zero-inflated PINAR model with seasonality based on the binomial thinning operator. The main feature of the proposed PINAR process is that it blends both zero-inflation and seasonality features together in a single setup. We mostly use ideas from the model of Jazi, Jones, and Lai, 2012a and the study of how the seasonal dummy model and the seasonal model with sine-cosine functions match up to create this PINAR process. We capture both the features of zero-inflation and seasonality in our proposed PINAR process. Data on the weekly maximum temperature of Kaohsiung City, Taiwan, from 2009 to 2012 is considered in the innovations of our proposed PINAR process to incorporate the effect of temperature, as temperature is a significant factor in spreading dengue all over the world. Especially in tropical and sub-tropical countries and on continents like North America and Europe, dengue disease increases with temperature. Each year, an estimated 100–400 million people get infected globally (see World Health Organization, n.d.), which is an important public health concern. Chan, Hu, and Hwang, 2015 too considered temperature covariates in their study. In our proposed PINAR process, we model the zero-inflated Poisson innovations incorporating the temperature data along with the sine-cosine functions to capture the underlying seasonal structure of the data of weekly dengue cases. The idea to include covariates in the innovations is from the study of Freeland and McCabe, 2004b. Hence, our study can be seen as a case study mainly based on Jazi, Jones, and Lai, 2012a and Freeland and McCabe, 2004b. However, the main limitation of incorporating zero-inflated innovations with seasonality is that it transforms the proposed process into a time-varying process, making it challenging to determine how many lag values from the original model to include. In this study, we solely focus on the order 1 process, but we would like to pursue this study for more than order 1 in future. Therefore, without any loss of generality, we assume that the original time series process Y_t depends only on its recent lag value Y_{t-1}. So, we coin it as a zero-inflated PINAR process of order 1 with seasonality.

The rest of this chapter is organized as follows. Section 5.2 discusses the weekly data of dengue cases along with the data of the weekly maximum temperature. The elaborate discussion of our proposed model is given in Section 5.3. The distributional and the h-step ahead forecasting properties are discussed

in Sections 5.4 and 5.5, respectively. In Section 5.6, we describe the estimation method for our proposed model. Some simulation results are given in Section 5.7 to show the empirical consistency of the estimated parameters. In Section 5.8, we analyze the data of weekly dengue cases. Finally, some conclusions are drawn in Section 5.9. All the proofs of the theoretical results are provided in Appendix (Section 5.10).

5.2 Motivating data example: Dengue data

In our analysis, we consider a disease incidence data over a period of four years. The data, consisting of a total of 209 observations, lists the weekly number of dengue cases, reported in Kaohsiung City, Taiwan, during the period from 2009 to 2012. We also incorporate the data of weekly maximum temperature, reported over that same period in the innovations of our proposed model, as mentioned earlier.

The raw dataset contains onset date, village ID, and other variables, which include the weather data on the onset date. The weather data contains maximum and minimum temperatures, air pressure, etc. on the onset date and their lag values (up to lag 7). Here, we use the columns of the onset dates and the maximum temperature recorded on those dates to aggregate the data-points to construct two weekly datasets, i.e., the weekly number of dengue cases and the weekly maximum temperature data. We have some missing observations in the aggregated data of weekly maximum temperature. To get rid of the missingness, we use the popular Kalman Smoothing method. For a seasonal time series with some missingness, like the continuous weekly maximum temperature data, the Kalman Smoothing method works well (see Moritz and Bartz-Beielstein, 2017). We have around 28, 39, 32 and 40 data-points for each year during the period of 2009 to 2012, respectively, in our aggregated weekly maximum temperature dataset. We employ the "na.kalman()" function in "imputeTS" package of R software with default model choice of "StructTS" (a structural model fitted by maximum likelihood). Moritz and Bartz-Beielstein, 2017 extensively studied this and showed that the Kalman smoothing method performs good for seasonal time series data with StructTS.

Figures 5.1 and 5.2 display the data of weekly dengue cases and the complete data of weekly maximum temperature. From the study of You et al., 2022, we can say that this complete dataset of weekly maximum temperature is a good representative of the true dataset. In Figures 5.1 and 5.2, we can clearly observe both zero-inflation and seasonality features that mainly trigger the idea to develop the

zero-inflated PINAR process with seasonal innovations. We discuss the proposed INAR process in detail in the subsequent section (see Section 5.3). As mentioned earlier, the main focus of our study is to fit an AR(1) model to the given dataset. However, we would like to pursue this study for higher order in some future work.

In Figure 5.3, two boxplots are provided to see the pattern of weekly dengue counts along with the changes in the pattern of the weekly maximum temperature over the period of 52 weeks. These boxplots are also similar to overall trend analyses of weekly dengue incidence rates and weekly maximum temperatures during 2007-2013 and 2016-2017 (non-epidemic periods), studied by You et al., 2022. As we can observe in Figure 5.3, around the week of 45, the number of dengue cases is maximum whereas the data of weekly maximum temperature reaches the peak at around the week of 35.

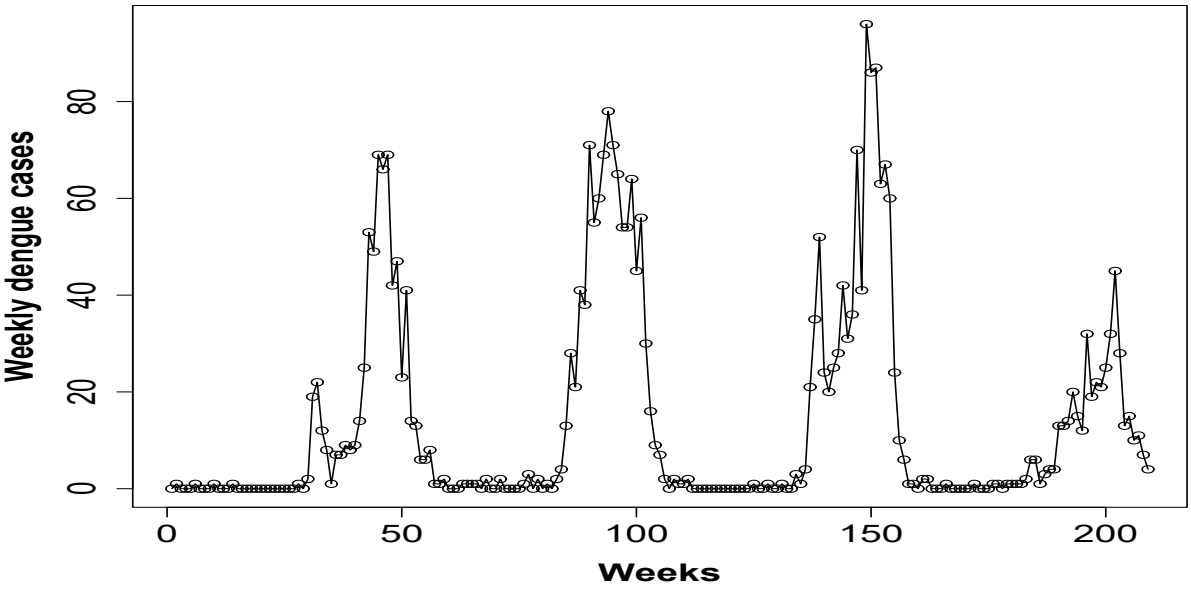


FIGURE 5.1: The weekly dengue data

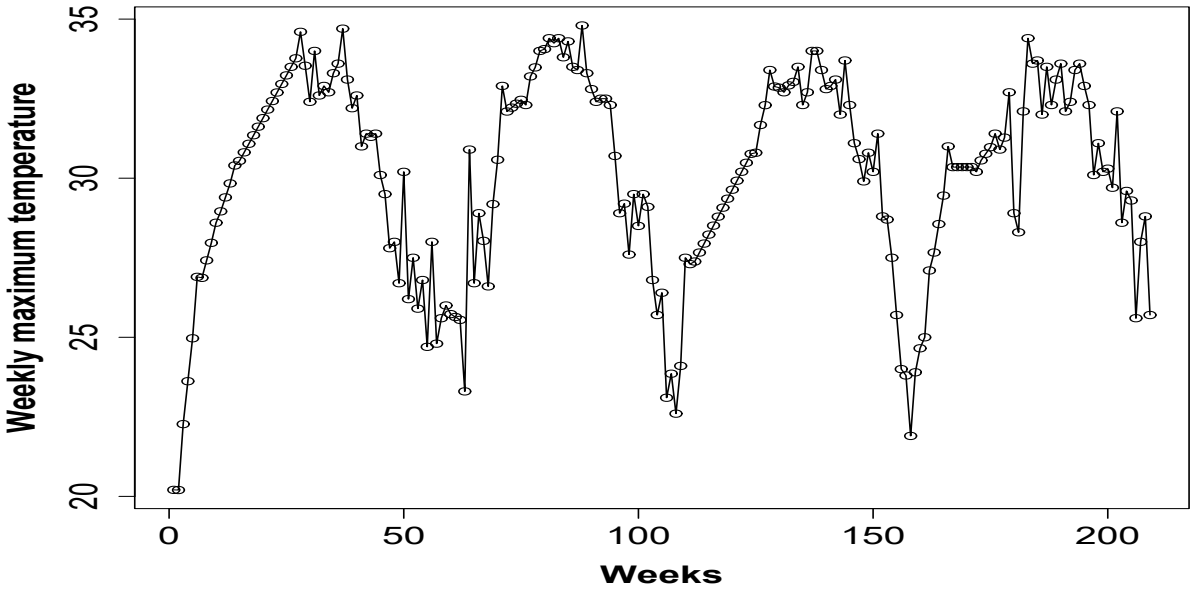


FIGURE 5.2: The weekly maximum temperature data

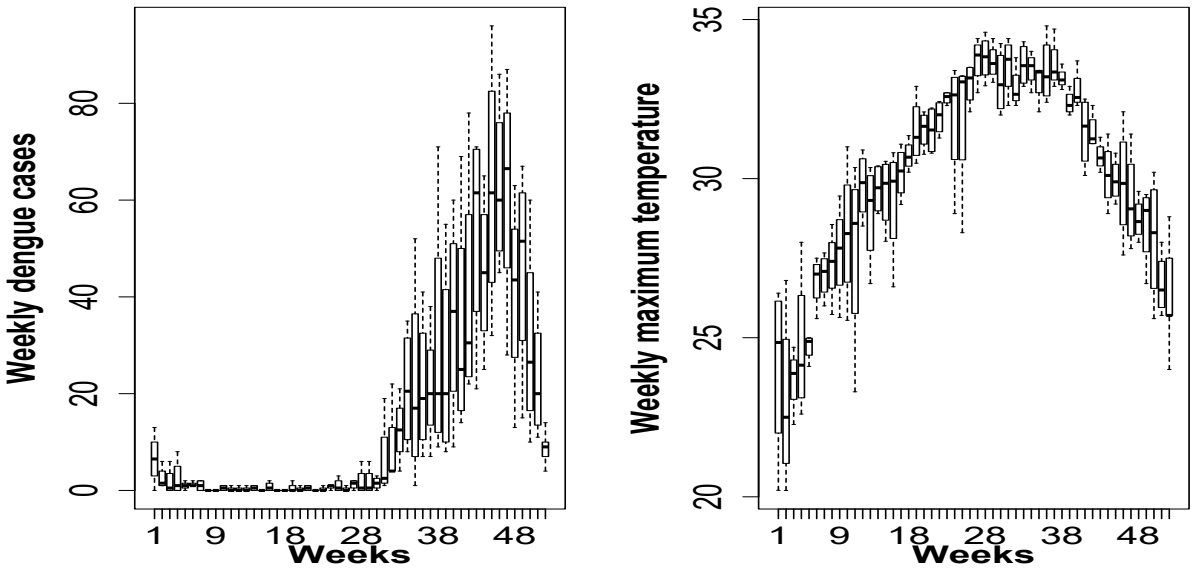


FIGURE 5.3: The boxplots showing how the number of weekly dengue cases behave with the changes in weekly maximum temperature over 52 weeks

5.3 The Model

Here, we provide a detailed discussion on our proposed zero-inflated PINAR model with seasonality based on the binomial thinning operator.

5.3.1 The Proposed model

The proposed model for a zero-inflated seasonal count time series data, like the weekly dengue data, using zero-inflated Poisson innovations is given by

$$Y_t = \alpha \circ Y_{t-1} + \varepsilon_t, \tag{5.3.1}$$

where ε_t represents new cases at time-point t. Here, ε_t follows a zero-inflated Poisson distribution with parameters ρ and λ_t^s, i.e., ZIP(ρ,λ_t^s) (see Jazi, Jones, and Lai, 2012a), where λ_t^s is assumed to have the following form

$$\lambda_t^s = \exp\left[\beta_1^s \cos\left(\frac{2\pi t}{l}\right) + \beta_2^s \sin\left(\frac{2\pi t}{l}\right) + \beta_3^s X_t\right], \tag{5.3.2}$$

where l is the number of seasons in a count time series data with seasonality, and X_t represents the covariate component at time-point t, i.e., the weekly maximum temperature at time-point t in our analysis. For a weekly time series, like the data of dengue cases, l is equal to 52. β_1^s and β_2^s are the model parameters associated with the seasonal components, and β_3^s is the model parameter associated with the covariate data $\{X_t\}$. The probability mass function (pmf) of ε_t's is given by

$$P(\varepsilon_t = m|\mathcal{X}_t^s) = \rho I_{\{0\}}(m) + (1-\rho)\frac{e^{-\lambda_t^s}(\lambda_t^s)^m}{m!}, \quad m = 0, 1, 2, \ldots, \tag{5.3.3}$$

where $\mathcal{X}_t^s$ is the set of all time-varying components up to time-point t which includes the seasonal functions, i.e., the sine-cosine functions and the covariate term, i.e., X_t; $\rho \in (0, 1)$; $I_{(\cdot)}$ is the indicator function.

This special form of λ_t^s is considered to incorporate the sine and cosine functions in our proposed model to capture the effect of seasonality, present in the count time series, like the weekly data of dengue cases. This form is developed from the idea behind the correspondence between the seasonal dummy model and the seasonal model with sine-cosine terms (see, e.g., Lovell, 1963). The idea to incorporate covariates in the innovations is mainly based on Freeland and McCabe, 2004a.

Zheng, Basawa, and Datta, 2007 proposed a random coefficient INAR(1) (RCINAR(1)) process, where they used a similar kind of approach on the parameter

α, by replacing it by a random parameter α_t. Here, α_t's are the realizations of i.i.d. random variables that can take values in the interval $[0, 1)$. We could have employed the RCINAR approach to extend our proposed model further by making both the survival probability and the innovations time-varying. But, since it raises more complexity to the model, and moreover both the seasonality and the zero-inflation features can be incorporated in our proposed INAR process through modelling the innovations (the main purpose of this study), we only use the time-varying innovations in our proposed model. However, the conditional expectation structures for both the RCINAR(1) and our proposed processes are similar, apart from the innovation terms, because of its time-varying nature in our proposed method. The gradual development of our proposed zero-inflated PINAR(1) process with seasonality is discussed in detail in the subsequent section.

5.3.2 Development of the proposed model

Here, we study the correspondence between two seasonal models, mainly used in the continuous framework, namely the seasonal dummy model and the seasonal model with sine-cosine functions. We discuss the gradual development of our proposed model from this correspondence. In order to do that, we first look at the seasonal dummy model which is given by

$$S_t = \delta_1 d_{1t} + \delta_2 d_{2t} + \ldots + \delta_s d_{st} = \sum_{i=1}^{s} \delta_i d_{it}, \tag{5.3.4}$$

where s is the number of seasons, and d_{it} is a dummy variable (d_{it} is equal to 1 for the i-th season of the year, and 0 for all other seasons, where $i = 1, \ldots, s$). The parameters $\delta_1, \ldots, \delta_s$ are the coefficients associated with the seasonal dummy variables.

The seasonal dummy model (5.3.4) (assuming s as even) can be represented equivalently in terms of sine and cosine functions as

$$\sum_{i=1}^{s} \delta_i d_{it} = \mu + \sum_{i=1}^{s/2} \left[\alpha_i^{s*} \cos\left(\frac{2\pi i t}{s}\right) + \beta_i^{s*} \sin\left(\frac{2\pi i t}{s}\right) \right], \tag{5.3.5}$$

where the coefficients α_i^{s*}'s and β_1^{s*}'s are related to the δ_i's. The correspondence between these two representations given in the relation (5.3.5) can be established. In Appendix (see Section 5.10), we derive for the quarterly case, i.e., for $s = 4$. The derivation can easily be extended for more than 4 seasons.

As mentioned earlier, the seasonal dummy model given in equation (5.3.4) involves s number of parameters $\delta_1, \ldots, \delta_s$ for s dummy variables d_{it}'s for the s number of seasons, where d_{it} is equal to 1 in the i-th season of the year, and 0 in all other seasons, $i = 1, \ldots, s$. Hence, apart from seasonality, the zero-inflation property of a time series, like the weekly data of dengue cases, can also be handled using the seasonal dummy model (given in equation (5.3.4)), largely due to the presence of the dummy variables in this framework. Therefore, the correspondence between the seasonal dummy model and the seasonal model with sine-cosine functions allows the seasonal model with sine-cosine functions (equation (5.3.5)) to capture the zero-inflation property too, along with the underlying seasonal structure of a count time series.

Now, we discuss how we connect the two above-mentioned seasonal models and develop the proposed model using a reduced set of model parameters. From the proposed model for l seasons given in equation (5.3.2), we have

$$E\left(\varepsilon_t|\mathcal{X}_t^s\right) = (1-\rho)\exp\left[\beta_1^s \cos\left(\frac{2\pi t}{l}\right) + \beta_2^s \sin\left(\frac{2\pi t}{l}\right) + \beta_3^s X_t\right]. \tag{5.3.6}$$

So, we can rewrite the equation (5.3.6) as

$$\ln E\left(\varepsilon_t|\mathcal{X}_t^s\right) = \ln(1-\rho) + \beta_1^s \cos\left(\frac{2\pi t}{l}\right) + \beta_2^s \sin\left(\frac{2\pi t}{l}\right) + \beta_3^s X_t. \tag{5.3.7}$$

If we compare the form of $\ln E\left(\varepsilon_t|\mathcal{X}_t^s\right)$ with the seasonal model consisting of sine-cosine functions (equation (5.3.5)) for $l = s$, it is clear that we use an approximation of the full seasonal model with sine-cosine terms by considering a reduced number of sine-cosine functions along with a covariate term to model the innovations of the proposed model, whereas $(s-1)$ sine-cosine terms are used in the full seasonal model (equation (5.3.5)). The reason behind this proposition is given in note 5.1. The inclusion of the covariate data $\{X_t\}$ (the weekly data of maximum temperature in this study) in the innovations of our proposed method along with sine-cosine functions aims to incorporate the effect of an important factor directly affecting the outcomes of the count time series data $\{Y_t\}$, like the role of temperature in the global spread of dengue infections. Here, the term $\ln(1-\rho)$ in equation (5.3.7) resonates the kind of role which the intercept μ has in the seasonal model with full sine-cosine functions.

Note 5.1 *We employ the approximation by using a reduced number of sine-cosine functions in the innovations instead of the full set of model parameters associated with the sine-cosine terms given in equation* (5.3.5)*, mainly to avoid the burden of computational*

difficulties that the full parameter set brings. The smooth oscillating natures of sine-cosine functions also provide the flexibility to capture the underlying seasonal structure of a count time series, even with a reduced number of model parameters. This exercise cannot be employed with the seasonal dummy model, where we need the full set of model parameters for the dummy variables. Using the full set of parameters with the seasonal dummy model can make the estimation process inefficient, and hence, we need the flexibility of an approximation which the seasonal model with sine-cosine functions provides. However, the theoretical results given in the subsequent sections are also applicable for the complete set of parameters associated with the sine-cosine terms mentioned in equation (5.3.5).

In Figure 5.4, we provide some sample paths (size of 200, i.e., around a four-year data) of the count time series for a fixed set of $(\alpha, \beta_1^s, \beta_2^s, \beta_3^s)$*, which is* $(0.6, -1.5, -1.2, 0.5)$*, with varying* ρ*. Here, the sample paths are generated from our proposed model using standardized values of the simulated covariate data (discussed in detail in Section 5.7.1). From this figure, we can clearly observe the seasonality feature and the increase in the number of zeros (zero-inflation) with the higher values of* ρ*.*

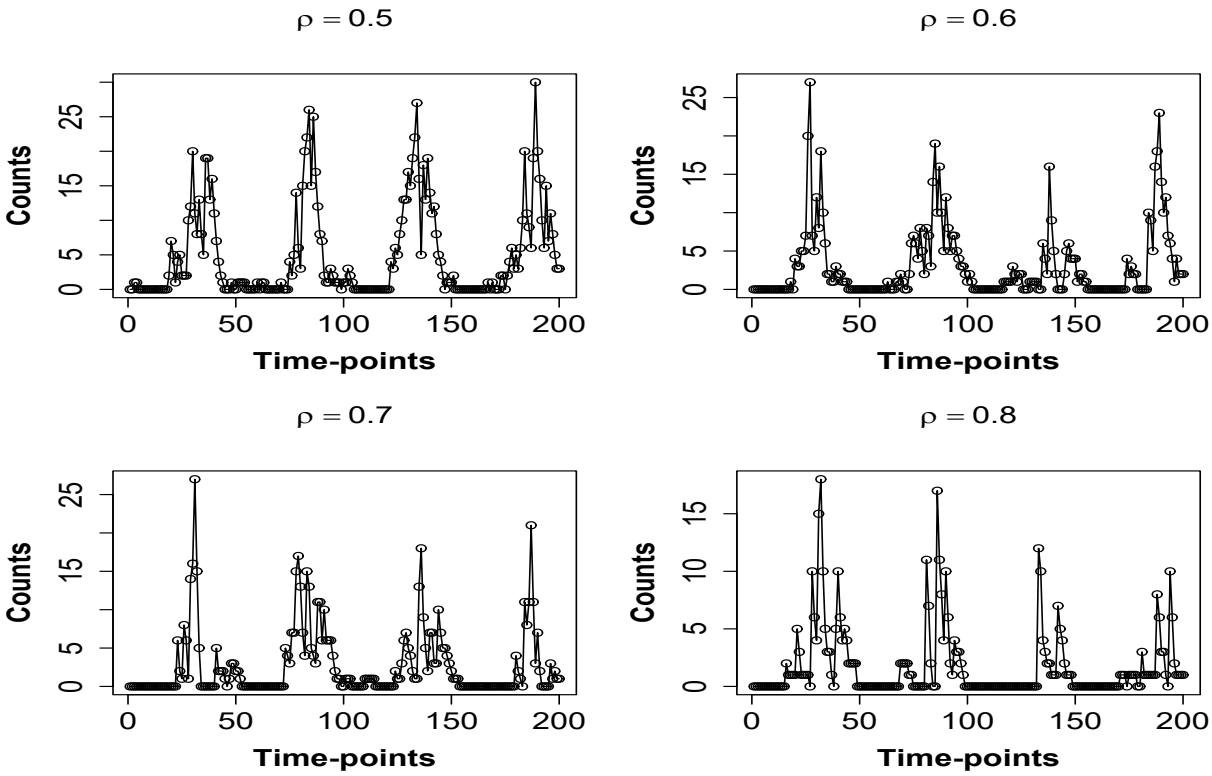


FIGURE 5.4: Some samples of the datasets generated using our proposed process with varying ρ and $(\alpha, \beta_1^s, \beta_2^s, \beta_3^s) = (0.6, -1.5, -1.2, 0.5)$

5.4 Distributional properties

5.4.1 Conditional properties

Under the proposed setup, the conditional distribution of Y_t given Y_{t-1} and $\mathcal{X}_t^s$ can be derived as

$$\begin{aligned} p_{ij} &= P(Y_t = j | Y_{t-1} = i, \mathcal{X}_t^s) \\ &= \sum_{k=0}^{\min(i,j)} P(\alpha \circ Y_{t-1} = k | Y_{t-1} = i) P(\epsilon_t = j - k | \mathcal{X}_t^s) \\ &= \sum_{k=0}^{\min(i,j)} \binom{i}{k} \alpha^k (1-\alpha)^{i-k} \left[\rho I_{\{0\}} + (1-\rho) \frac{e^{-\lambda_t^s} (\lambda_t^s)^{j-k}}{(j-k)!} \right], \quad i, j = 0, 1, 2, \ldots, \end{aligned}$$

where $I_{(\cdot)}$ is the indicator function. This is the probability of going from state i to state j in a single step.

The conditional mean and variance can be derived as

$$E(Y_t | Y_{t-1}, \mathcal{X}_t^s) = \alpha Y_{t-1} + \lambda_t^s (1-\rho),$$

and

$$\begin{aligned} V(Y_t | Y_{t-1}, \mathcal{X}_t^s) &= V(\alpha \circ Y_{t-1} | Y_{t-1}, \mathcal{X}_t^s) + V(\epsilon_t | \mathcal{X}_t^s) \\ &= \alpha(1-\alpha) Y_{t-1} + \lambda_t^s (1-\rho)(1 + \rho \lambda_t^s). \end{aligned}$$

5.4.2 Marginal properties

Since the marginal properties of Y_t are difficult to obtain, we find the partial marginal properties of Y_t given $\mathcal{X}_t^s$ for $t > 1$, and henceforth they are called the marginal properties. We assume that the marginal distribution of $Y_1 | \mathcal{X}_1^s$ follows zero-inflated Poisson distribution with parameters ρ and λ_1^s. The reason behind this assumption is similar to what is discussed in Section 4.4.2 of Chapter 4.

Here, we can use a recursive formula to derive the marginal (partial) probability function of Y_t given $\mathcal{X}_t^s$ for $t > 1$, which is given by

$$\begin{aligned} p_{t,j} &= P(Y_t = j | \mathcal{X}_t^s) \\ &= \sum_{i=0}^{\infty} p_{ij} P(Y_{t-1} = i | \mathcal{X}_t^s) \\ &= \sum_{i=0}^{\infty} \sum_{k=0}^{\min(i,j)} \binom{i}{k} \alpha^k (1-\alpha)^{i-k} \left[\rho I_{\{0\}} + (1-\rho) \frac{e^{-\lambda_t^s} (\lambda_t^s)^{j-k}}{(j-k)!} \right] p_{t-1,i}, \quad j = 0, 1, 2, \ldots, \end{aligned}$$

which is clearly a mixture distribution involving the structures of a binomial probability, a zero-inflated Poisson probability, and the partial marginal probability at previous time-point.

Result 5.1 *Under the assumptions that* $Y_1|\mathcal{X}_1^s \sim ZIP(\rho, \lambda_1^s)$ *and* $\varepsilon_n|\mathcal{X}_n^s \sim Poisson(\rho, \lambda_n^s)$, *we can show that the marginal mean and the marginal variance are given by*

$$E(Y_n|\mathcal{X}_n^s) = \left(\alpha^{n-1}\lambda_1^s + \alpha^{n-2}\lambda_2^s + \ldots + \lambda_n^s\right)(1-\rho),$$

and

$$\begin{aligned} V(Y_n|\mathcal{X}_n^s) &= \left(\alpha^{n-1}\lambda_1^s + \alpha^{n-2}\lambda_2^s + \ldots + \lambda_n^s\right)(1-\rho) + \\ &\qquad \rho(1-\rho)\left[(\alpha^{n-1}\lambda_1^s)^2 + (\alpha^{n-2}\lambda_2^s)^2 + \ldots + (\lambda_n^s)^2\right]. \end{aligned}$$

The derivation of this result is presented in Appendix (see Section 5.10).
Therefore, the variance-to-mean ratio of n-th count process Y_n given $\mathcal{X}_n^s$ is

$$\frac{\sigma^2_{Y_n|\mathcal{X}_n^s}}{\mu_{Y_n|\mathcal{X}_n^s}} = 1 + \rho\frac{\left[(\alpha^{n-1}\lambda_1^s)^2 + (\alpha^{n-2}\lambda_2^s)^2 + \ldots + (\lambda_n^s)^2\right]}{\left[\alpha^{n-1}\lambda_1^s + \alpha^{n-2}\lambda_2^s + \ldots + \lambda_n^s\right]}.$$

Hence, our proposed process can be a viable method for analyzing an overdispersed integer-valued time series like the weekly data of dengue cases. Here, we consider the marginal (partial) distribution of Y_t given $\mathcal{X}_t^s$ for $t > 1$ as the marginal distribution of Y_t; but for this proposed setup, we cannot find any explicit form of known distribution for the partial marginal distribution of Y_t given $\mathcal{X}_t^s$ for $t > 1$ (see Jazi, Jones, and Lai, 2012a), as it can be found for the marginal (partial) distribution for PINAR(1) process with time-varying covariates in Chattopadhyay et al., 2021.

5.4.3 Autocorrelation structure

Result 5.2 *Under the above setup, the auto-covariance function (ACVF) between* Y_t *and* Y_{t+h} *given* $\mathcal{X}_{t+h}^s$ *where* $Y_{t+h} = \alpha^h \circ Y_t + \sum_{i=1}^h \alpha^{h-i} \circ \varepsilon_{t-i}$ *can be derived as*

$$\begin{aligned} \gamma_y(h) &= Cov(Y_t, Y_{t+h}|\mathcal{X}_{t+h}^s) \\ &= \alpha^h(1-\rho)\left[\alpha^{t-1}\lambda_1^s + \ldots + \lambda_t^s\right] + \alpha^h\rho(1-\rho)\left[(\alpha^{t-1}\lambda_1^s)^2 + \ldots + (\lambda_t^s)^2\right]. \end{aligned}$$

The derivation of this result is presented in Appendix (see Section 5.10).

Hence, for $h \neq 0$, the ACF can be derived as follows:

$$\begin{aligned}\rho_y(h) &= \frac{Cov(Y_t, Y_{t+h}|\mathcal{X}^s_{t+h})}{\sqrt{Var(Y_t|\mathcal{X}^s_{t+h})\, Var(Y_{t+h}|\mathcal{X}^s_{t+h})}} \\ &= \alpha^h \sqrt{\frac{Var(Y_t|\mathcal{X}^s_{t+h})}{Var(Y_{t+h}|\mathcal{X}^s_{t+h})}} \\ &= \alpha^h \sqrt{\frac{\left[\alpha^{t-1}\lambda^s_1 + \ldots + \lambda^s_t\right] + \rho\left[(\alpha^{t-1}\lambda^s_1)^2 + \ldots + (\lambda^s_t)^2\right]}{\left[\alpha^{t+h-1}\lambda^s_t + \ldots + \lambda^s_{t+h}\right] + \rho\left[(\alpha^{t+h-1}\lambda^s_1)^2 + \ldots + (\lambda^s_{t+h})^2\right]}}.\end{aligned}$$

It decays exponentially to 0, as h goes to ∞, since λ^s_i's are bounded, due to the fact that the seasonal components (sine-cosine terms) and X_t's involved in the innovations are bounded.

5.4.4 Distribution of zeros

Result 5.3 *Under the assumptions that* $Y_1|\mathcal{X}^s_1 \sim ZIP(\rho, \lambda^s_1)$ *and* $\varepsilon_n|\mathcal{X}^s_n \sim ZIP(\rho, \lambda^s_n)$, *we can show that the PGF of* $Y_n|\mathcal{X}^s_n$ *is*

$$\Phi_{Y_n|\mathcal{X}^s_n}(s) = \prod_{i=0}^{n-1}\left[\rho + (1-\rho)e^{-\lambda^s_{n-i}\alpha^i(1-s)}\right],$$

where $\Phi_X(\cdot)$ *is the PGF of random variable* X.

The derivation of this result is presented in Appendix (see Section 5.10).
For the proposed model, we can write

$$P[Y_n = 0|\mathcal{X}^s_n] = \prod_{i=0}^{n-1}\left[\rho + (1-\rho)e^{-\lambda^s_{n-i}\alpha^i}\right] = p^1_{0,n} \text{ (say)}.$$

And for a simple PINAR(1) process with time-varying covariates (see Chattopadhyay et al., 2021), we have

$$P[Y_n = 0|\mathcal{X}^s_n] = e^{-\sum_{i=0}^{n-1}\lambda^s_{n-i}\alpha^i} = p^2_{0,n} \text{ (say)}.$$

We show $p^1_{0,n}$ $(= P[Y_n = 0|\mathcal{X}^s_n]$ for the proposed zero-inflated PINAR(1) process with seasonality$) > p^2_{0,n}$ $(= P[Y_n = 0|\mathcal{X}^s_n]$ for simple PINAR(1) process with time-varying covariates; see Chattopadhyay et al., 2021) in Appendix (see Section 5.10).

5.5 Forecasting

5.5.1 h-step ahead forecasting properties

In order to derive the mean predictor, we use the following recursive method:

$$\begin{aligned} Y_{n+h} &= \alpha \circ Y_{n+h-1} + \varepsilon_{n+h} \\ &= \alpha \circ \{\alpha \circ Y_{n+h-2} + \varepsilon_{n+h-1}\} + \varepsilon_{n+h} \\ &\vdots \\ &= \alpha^h \circ Y_n + \sum_{i=1}^{h} \alpha^{h-i} \circ \varepsilon_{n+i}. \end{aligned}$$

The h-step ahead conditional mean (the mean predictor) and the conditional variance can be given as

$$\hat{Y}_{n+h}|Y_n, \mathcal{X}^s_{n+h} = E(Y_{n+h}|Y_n, \mathcal{X}^s_{n+h}) = \alpha^h Y_n + \sum_{i=1}^{h} \alpha^{h-i}\lambda^s_{n+i}(1-\rho),$$

and

$$\begin{aligned} V(Y_{n+h}|Y_n, \mathcal{X}^s_{n+h}) &= \alpha^h\left(1-\alpha^h\right)Y_n + \\ &\sum_{i=1}^{h}\left[\alpha^{h-i}\left(1-\alpha^{h-i}\right)\lambda^s_{n+i}(1-\rho) + \alpha^{2(h-i)}\lambda^s_{n+i}(1-\rho)(1+\rho\lambda^s_{n+i})\right] \\ &= \alpha^h\left(1-\alpha^h\right)Y_n + \sum_{i=1}^{h}\left[\alpha^{h-i}\lambda^s_{n+i}(1-\rho) + \rho(1-\rho)\left(\alpha^{h-i}\lambda^s_{n+i}\right)^2\right] \\ &= \alpha^h\left(1-\alpha^h\right)Y_n + \sum_{i=1}^{h}\alpha^{h-i}\lambda^s_{n+i}(1-\rho)\left[1+\rho\left(\alpha^{h-i}\lambda^s_{n+i}\right)\right]. \end{aligned}$$

5.6 Parameter Estimation

Conditional least squares estimation method

As discussed in Chapter 4, the conditional least squares method is usually used for estimating the regression parameters of the model in the context of time series models (see Freeland and McCabe, 2004a; Freeland and McCabe, 2005).

In order to implement the conditional least squares estimation method, we need to minimize the sum of squared deviation about the conditional expectation which is given as

$$Q^*(\boldsymbol{\beta}^s) = \sum_{t=2}^{n}[Y_t - E(Y_t|Y_{t-1}, \mathcal{X}^s_t)]^2, \tag{5.6.1}$$

instead of $Q(\boldsymbol{\beta^s}) = \sum_{t=2}^{n}[Y_t - E(Y_t|\mathcal{X}_t^s)]^2$, with respect to the regression parameters of the model where $E(Y_t|Y_{t-1}, \mathcal{X}_t^s) = \alpha Y_{t-1} + (1-\rho)\lambda_t^s$. The form of λ_t^s, i.e., the form of zero-inflated innovation terms with seasonality is given in equation (5.3.2). In equation (5.6.1), $\boldsymbol{\beta^s}$ indicates the vector for the regression parameters, which are $(\alpha, \rho, \beta_1^s, \beta_2^s, \beta_3^s)$ for our proposed process. Here, numerical methods are being employed to obtain the CLS estimates of the regression parameters of the model, as there are no closed forms of the CLS estimators. We use the "optim()" function in R software for this purpose.

In the subsequent section, simulation studies are performed to establish the consistency of the CLS method.

5.7 Simulation study

5.7.1 The setup

In this section, we perform some simulation experiments regarding the estimation of model parameters and the forecasting performances of the proposed model. We generate the simulated count time series $\{Y_t\}$ from the proposed model (5.3.1) with λ_t^s given in equation (5.3.2), using the original covariate data $\{X_t\}$, i.e., the weekly maximum temperature data. We fit an existing model to this dataset ($\{X_t\}$) from the literature. That is, we use the "auto.arima()" function from "Forecast" package (see Hyndman and Khandakar, 2008) of R software to fit an ARIMA model to the real covariate dataset we have in hand, i.e., the weekly maximum temperature data. Then, for each replication in our simulation study, we generate multiple sample paths (100 covariate datasets) using the fitted model by the "simulate()" function, which produces observations, conditional on the historical observations. Following that, we average them to obtain one good representative dataset of the real covariate dataset for each replication by reducing the fluctuations which we see in one single simulated sample path (see Figure 5.5 in Appendix (Section 5.10)).

The simulation studies are performed for varying sample sizes and different choices of model parameters. In previous sections, $\mathcal{X}_n^s$, the set of all time-varying covariates up to time-point n, is equal to $\{1, S_1^1, S_1^2, X_1, S_2^1, S_2^2, X_2, \ldots, S_n^1, S_n^2, X_n\}$, where $S_t^1 = \cos\left(\frac{2\pi t}{l}\right)$ and $S_t^2 = \sin\left(\frac{2\pi t}{l}\right)$, which are the seasonal components, l being the number of seasons ($l = 52$ for our study), and X_t is related to the covariate data. However, we use the standardized values of the simulated covariate dataset (see Figure 5.5 in Appendix (Section 5.10)) for the simulation study, and the standardized real covariate values for the data analysis (see Section 5.8). That

is, in the simulation experiments, we use the components $\{1, S_1^1, S_1^2, std(X_1), \ldots, S_n^1, S_n^2, std(X_n)\}$ (where $std(X_t)$ is the standardized value of X_t) to generate the count data of varying sample sizes by the data generating process mentioned in equation (5.3.1) with λ_t^s given in equation (5.3.2). Therefore, in both simulation study and data analysis, by $\mathcal{X}_n^s$, we mean the set $\{1, S_1^1, S_1^2, std(X_1), \ldots, S_n^1, S_n^2, std(X_n)\}$. Here, we consider "VGAM" package in R software to use "zipois" functions.

Note 5.2 *In the simulation study as well as the data application, we compare our proposed model to the following INAR models, which are briefly discussed below:*

- *The **comparison model 1**, where INAR(1) model contains only the seasonal feature but no the zero-inflation property, i.e., the comparison model 1 involves four model parameters $(\alpha, \beta_1^s, \beta_2^s, \beta_3^s)$ instead of $(\alpha, \rho, \beta_1^s, \beta_2^s, \beta_3^s)$, which is the case for the proposed zero-inflated INAR(1) model with seasonality.*
- ***ZINAR(1)** model, which is a first-order integer-valued AR process with zero-inflated Poisson innovations. This model was studied by Jazi, Jones, and Lai, 2012a.*

Remark 5.1 *Here, we use CLS method in estimating the model parameters for all the models. Since the conditional mean structures for ZINAR(1), some other INAR processes like GINAR(1) (see McKenzie, 1986; Maiti and Biswas, 2015b), NBINAR(1) (see Ristić, Nastić, and Bakouch, 2012) and NGINAR(1) model (see Ristić, Bakouch, and Nastić, 2009) processes are similar, we should have similar numerical results for these models. Therefore, in simulation studies and data analysis, we provide the numerical results of the ZINAR(1) model only.*

5.7.2 Empirical consistency

Here, we perform a simulation experiment to investigate the empirical consistency of the estimated parameters of the proposed model. In order to execute this study, we simulate the data from the proposed model (5.3.1) with λ_t^s given in equation (5.3.2). Different sets of model parameters are considered for this data generating case. Those sets of values of $\boldsymbol{\beta^s}$, i.e., $(\alpha, \rho, \beta_1^s, \beta_1^s, \beta_1^s)$, are $(0.4, 0.4, 1.2, -1.5, 0.5)$, $(0.4, 0.6, 1.2, -1.5, 0.5)$, $(0.6, 0.4, 1.2, -1.5, 0.5)$, $(0.4, 0.4, -1.5, 1, 0.7)$, $(0.6, 0.4, -1.5, 1, 0.7)$ and $(0.6, 0.6, -1.5, 1, 0.7)$. For this proposed data generating method, $\mathcal{X}_n^s$, the set of all time-varying covariates up to time-point n, contains the seasonal components and the covariate term up to time-point n, as described in Section 5.7.1, where n is the sample size of the simulated data. We use the standardized values of the simulated covariate datasets in this study. Four different sample sizes of 200, 500, 1000, and 5000 are explored. All the simulations results are

based on 1000 Monte Carlo replications. We estimate the regression parameters using CLS method. We report the mean estimates and the MSEs of the regression parameters in Tables 5.1 and 5.2, where we can see that as the sample size increases, the MSEs of the estimated regression parameters $\boldsymbol{\beta}^s$ decrease. This empirically establishes the consistency of the estimated model parameters.

TABLE 5.1: Mean estimates of the regression parameters $\boldsymbol{\beta}^s$ with their respective MSEs for different sample sizes, where the data generating process is the proposed zero-inflated PINAR(1) with seasonality and true $\boldsymbol{\beta}^s = (0.4, 0.4, 1.2, -1.5, 0.5)$, $(0.4, 0.6, 1.2, -1.5, 0.5)$ and $(0.6, 0.4, 1.2, -1.5, 0.5)$

$\boldsymbol{\beta}^s = (0.4, 0.4, 1.2, -1.5, 0.5)$					
n	$\hat{\alpha}$ (MSE)	$\hat{\rho}$ (MSE)	$\hat{\beta}_1^s$ (MSE)	$\hat{\beta}_2^s$ (MSE)	$\hat{\beta}_3^s$ (MSE)
200	0.3488(0.0159)	0.4675(0.0668)	1.3607(0.4768)	−1.8293(0.9404)	0.5459(0.3023)
500	0.3806(0.0056)	0.4304(0.0250)	1.2398(0.1071)	−1.6150(0.2105)	0.4975(0.0863)
1000	0.3906(0.0025)	0.4111(0.0115)	1.2243(0.0450)	−1.5393(0.0867)	0.5070(0.0398)
5000	0.3989(0.0005)	0.4023(0.0024)	1.1995(0.0069)	−1.5104(0.0129)	0.4962(0.0054)
$\boldsymbol{\beta}^s = (0.4, 0.6, 1.2, -1.5, 0.5)$					
n	$\hat{\alpha}$ (MSE)	$\hat{\rho}$ (MSE)	$\hat{\beta}_1^s$ (MSE)	$\hat{\beta}_2^s$ (MSE)	$\hat{\beta}_3^s$ (MSE)
200	0.3417(0.0186)	0.6974(0.0511)	1.5239(1.2412)	−2.1864(2.2919)	0.5732(0.7264)
500	0.3788(0.0060)	0.6488(0.0207)	1.3253(0.3162)	−1.7533(0.6112)	0.5173(0.1880)
1000	0.3883(0.0020)	0.6196(0.0102)	1.2373(0.0993)	−1.5964(0.2017)	0.5051(0.0796)
5000	0.3978(0.0005)	0.6044(0.0018)	1.2099(0.0139)	−1.5175(0.0250)	0.5018(0.0108)
$\boldsymbol{\beta}^s = (0.6, 0.4, 1.2, -1.5, 0.5)$					
n	$\hat{\alpha}$ (MSE)	$\hat{\rho}$ (MSE)	$\hat{\beta}_1^s$ (MSE)	$\hat{\beta}_2^s$ (MSE)	$\hat{\beta}_3^s$ (MSE)
200	0.5463(0.0139)	0.4279(0.0774)	1.3626(0.4579)	−1.8287(0.9660)	0.5368(0.3219)
500	0.5771(0.0047)	0.4081(0.0351)	1.2521(0.1267)	−1.6136(0.2425)	0.5022(0.0964)
1000	0.5906(0.0019)	0.4061(0.0154)	1.2137(0.0479)	−1.5483(0.0921)	0.4974(0.0449)
5000	0.5978(0.0004)	0.4007(0.0031)	1.2025(0.0079)	−1.5112(0.0140)	0.4977(0.0070)

TABLE 5.2: Mean estimates of the regression parameters $\boldsymbol{\beta}^s$ with their respective MSEs for different sample sizes, where the data generating process is the proposed zero-inflated PINAR(1) with seasonality and true $\boldsymbol{\beta}^s = (0.4, 0.4, -1.5, 1, 0.7)$, $(0.6, 0.4, -1.5, 1, 0.7)$ and $(0.6, 0.6, -1.5, 1, 0.7)$

$\boldsymbol{\beta}^s = (0.4, 0.4, -1.5, 1, 0.7)$					
n	$\hat{\alpha}$ (MSE)	$\hat{\rho}$ (MSE)	$\hat{\beta}_1^s$ (MSE)	$\hat{\beta}_2^s$ (MSE)	$\hat{\beta}_3^s$ (MSE)
200	0.3630(0.0132)	0.4115(0.0642)	−1.6087(0.3326)	1.1297(0.3778)	0.7803(0.3346)
500	0.3874(0.0047)	0.4025(0.0293)	−1.5470(0.0870)	1.0253(0.0805)	0.7124(0.0886)
1000	0.3919(0.0023)	0.4020(0.0144)	−1.5290(0.0387)	1.0140(0.0347)	0.7029(0.0410)
5000	0.3985(0.0005)	0.3992(0.0033)	−1.5009(0.0071)	1.0041(0.0066)	0.7043(0.0063)
$\boldsymbol{\beta}^s = (0.6, 0.4, -1.5, 1, 0.7)$					
n	$\hat{\alpha}$ (MSE)	$\hat{\rho}$ (MSE)	$\hat{\beta}_1^s$ (MSE)	$\hat{\beta}_2^s$ (MSE)	$\hat{\beta}_3^s$ (MSE)
200	0.5613(0.0114)	0.4081(0.0842)	−1.6990(0.4992)	1.2019(0.7608)	0.8213(0.4803)
500	0.5831(0.0034)	0.3829(0.0404)	−1.5515(0.0935)	1.0223(0.1282)	0.7217(0.1071)
1000	0.5923(0.0017)	0.3935(0.0216)	−1.5203(0.0403)	1.0126(0.0488)	0.7148(0.0472)
5000	0.5978(0.0003)	0.3975(0.0038)	−1.5069(0.0074)	0.9996(0.0070)	0.6974(0.0059)
$\boldsymbol{\beta}^s = (0.6, 0.6, -1.5, 1, 0.7)$					
n	$\hat{\alpha}$ (MSE)	$\hat{\rho}$ (MSE)	$\hat{\beta}_1^s$ (MSE)	$\hat{\beta}_2^s$ (MSE)	$\hat{\beta}_3^s$ (MSE)
200	0.5525(0.0143)	0.5944(0.0639)	−1.8689(1.1534)	1.3033(1.3990)	0.8630(0.9354)
500	0.5871(0.0042)	0.6101(0.0287)	−1.5985(0.2787)	1.1238(0.4159)	0.7868(0.3091)
1000	0.5912(0.0019)	0.6054(0.0148)	−1.5620(0.1059)	1.0553(0.1216)	0.7208(0.0942)
5000	0.5983(0.0003)	0.6022(0.0028)	−1.5136(0.0149)	1.0075(0.0135)	0.7051(0.0123)

5.7.3 Forecasting

This simulation experiment is done to study the h-step ahead forecasting performances of the proposed model for varying h, compared to two different comparison methods - (i) comparison model 1 and (ii) ZINAR(1) method. For comparison, we consider three measures of forecasting criteria, namely PRMSE(h), PTP(h), and PTPI(h) (where $\delta = 1, 2$) (see 2.8 of Chapter 2 and 3.7 of Chapter 3). In order to perform this study, we simulate the data from the proposed model (5.3.1) with λ_t^s given in equation (5.3.2). Different sets of regression parameters: (i) $(0.4, 0.4, 1.5, 1.2, -0.1)$ and $(0.5, 0.5, 1, -1, -0.5)$ (moderate counts with moderate zero-inflation), (ii) $(0.4, 0.7, 0.5, -0.5, 0.3)$ and $(0.6, 0.7, 0.5, -0.5, 0.3)$ (low counts with high zero-inflation), and (iii) $(0.5, 0.4, 1.5, 1.2, -0.2)$ and $(0.6, 0.4, 1.5, 1.2, -0.4)$ (moderate and high counts with moderate zero-inflation), are considered. As described for the simulation study regarding the empirical consistency, here, $\mathcal{X}_n^s$ also contains the seasonal components and the covariate term up to time-point n,

where n is the sample size of the simulated data. We use the standardized values of simulated the covariate datasets in this experiment. Each time we generate a data of sample size 500 of which a training set of size 400 is used to fit the three models considered for this study and a test set of size 100 is considered to find the forecasting measures for $h = 1, 2, 3, 4$. This procedure is repeated for 1000 times. We report the h-step ahead forecasting performances for all the three models for $h = 1, 2, 3, 4$ in Tables 5.3, 5.4 and 5.5.

From the tables, it is observed that the average PRMSE values of the proposed process are relatively smaller than those of the two comparison methods. For cases with low counts and high zero-inflation, the two zero-inflated models (the proposed model and the ZINAR(1) model) mostly perform better than the comparison model. However, the comparison model shows good performance for PTPI ($\delta = 2$). For moderate counts, the two seasonal models perform better than the ZINAR(1) model, but the ZINAR(1) model shows good results for PTPI values. For high counts, the two seasonal models provide better numerical results than the ZINAR(1) model, which is expected.

TABLE 5.3: PRMSE(h), PTP(h) and PTPI(h) values for varying h, where the data generating process is the proposed zero-inflated PINAR(1) with seasonality and true $\boldsymbol{\beta}^s = (0.4, 0.4, 1.5, 1.2, -0.1)$ and $(0.5, 0.5, 1, -1, -0.5)$ (moderate counts with moderate zero-inflation)

$\boldsymbol{\beta}^s = (0.4, 0.4, 1.5, 1.2, -0.1)$												
h	Proposed model				Comparison model 1				ZINAR(1)			
	PR.	PTP	PI1	PI2	PR.	PTP	PI1	PI2	PR.	PTP	PI1	PI2
1	2.32	41.14	67.67	79.09	2.34	36.09	66.82	78.81	2.57	15.41	68.92	79.64
2	2.52	38.75	66.49	78.25	2.54	31.11	65.15	77.91	2.98	12.82	58.74	77.64
3	2.56	38.07	65.97	77.89	2.58	29.24	64.28	77.43	3.22	10.46	41.66	77.01
4	2.58	37.59	65.59	77.60	2.60	28.51	63.69	77.16	3.38	8.75	31.02	76.64
$\boldsymbol{\beta}^s = (0.5, 0.5, 1, -1, -0.5)$												
1	1.91	44.43	71.92	84.38	1.94	34.35	70.84	84.09	2.07	28.66	75.01	85.69
2	2.16	40.23	69.60	81.64	2.20	21.80	68.24	81.28	2.44	16.82	70.03	82.71
3	2.22	39.45	68.79	80.92	2.26	19.38	67.31	80.64	2.62	15.32	63.51	81.84
4	2.24	39.32	68.49	80.73	2.28	18.88	67.08	80.34	2.74	14.16	57.64	81.53

Note: PR.: PRMSE values, PI1: PTPI values with $\delta = 1$ and PI2: PTPI values with $\delta = 2$

TABLE 5.4: PRMSE(h), PTP(h) and PTPI(h) values for varying h, where the data generating process is the proposed zero-inflated PINAR(1) with seasonality and true $\boldsymbol{\beta}^s = (0.4, 0.7, 0.5, -0.5, 0.3)$ and $(0.6, 0.7, 0.5, -0.5, 0.3)$ (low counts with high zero-inflation)

$\boldsymbol{\beta}^s = (0.4, 0.7, 0.5, -0.5, 0.3)$												
h	Proposed model				Comparison model 1				ZINAR(1)			
	PR.	PTP	PI1	PI2	PR.	PTP	PI1	PI2	PR.	PTP	PI1	PI2
1	0.96	58.98	91.28	97.01	1.11	18.12	92.71	97.85	0.97	67.02	90.72	96.36
2	1.03	52.12	90.33	96.77	1.15	17.94	89.61	97.88	1.08	49.80	90.50	96.07
3	1.03	51.58	90.22	96.70	1.15	17.94	89.37	97.93	1.10	35.97	91.80	96.54
4	1.03	51.71	90.35	96.74	1.15	17.96	89.56	97.98	1.10	31.44	92.41	96.77
$\boldsymbol{\beta}^s = (0.6, 0.7, 0.5, -0.5, 0.3)$												
1	0.97	58.90	90.49	96.96	1.13	22.50	89.81	97.74	0.98	65.13	89.79	96.45
2	1.13	45.43	87.07	96.11	1.28	20.63	79.84	97.40	1.18	39.96	88.20	95.47
3	1.19	40.89	85.66	95.73	1.31	20.30	77.05	97.25	1.22	25.22	89.58	95.83
4	1.20	39.18	85.01	95.66	1.32	20.12	76.15	97.21	1.22	23.99	89.99	95.75

Note: PR.: PRMSE values, PI1: PTPI values with $\delta = 1$ and PI2: PTPI values with $\delta = 2$

TABLE 5.5: PRMSE(h), PTP(h) and PTPI(h) values for varying h, where the data generating process is the proposed zero-inflated PINAR(1) with seasonality and true $\boldsymbol{\beta}^s = (0.5, 0.4, 1.5, 1.2, -0.2)$ (moderate to high counts with moderate zero-inflation) and $(0.6, 0.4, 1.5, 1.2, -0.4)$ (high counts with moderate zero-inflation)

$\boldsymbol{\beta}^s = (0.5, 0.4, 1.5, 1.2, -0.2)$												
h	Proposed model				Comparison model 1				ZINAR(1)			
	PR.	PTP	PI1	PI2	PR.	PTP	PI1	PI2	PR.	PTP	PI1	PI2
1	2.63	40.99	66.48	77.25	2.64	36.72	65.59	76.97	2.89	16.58	66.43	77.65
2	2.97	37.12	64.59	76.07	2.98	30.25	62.98	75.61	3.52	12.44	56.15	74.19
3	3.07	35.91	63.59	75.25	3.08	27.81	61.43	74.73	3.90	9.23	35.99	70.88
4	3.10	35.31	63.17	74.89	3.12	26.49	60.62	74.19	4.14	7.74	27.94	68.37
$\boldsymbol{\beta}^s = (0.6, 0.4, 1.5, 1.2, -0.4)$												
1	3.40	41.02	64.96	74.69	3.40	37.73	64.03	74.44	3.83	13.17	61.56	74.50
2	3.99	36.00	62.14	73.02	3.99	30.96	60.59	72.46	4.99	10.45	49.93	68.03
3	4.20	33.88	60.48	71.75	4.21	27.70	58.17	70.90	5.80	6.75	25.53	61.94
4	4.29	33.01	59.53	70.98	4.30	25.64	56.83	69.85	6.37	5.68	20.43	51.79

Note: PR.: PRMSE values, PI1: PTPI values with $\delta = 1$ and PI2: PTPI values with $\delta = 2$

5.8 Data analysis

In this section, we consider the data of weekly dengue cases, reported in Kaohsiung City, Taiwan, during the period 2009 to 2012 for the practical illustration of the proposed zero-inflated PINAR(1) model with seasonality. As discussed earlier, we incorporate the weekly maximum temperature data in the innovations to include the effects temperature has on the spread of dengue infections globally (see World Health Organization, n.d.). In this dataset, we have total number of 209 observations. Here $\mathcal{X}_n^s$, the set of all time-varying covariates up to time-point n, consists of the seasonal components and the covariate data up to time-point n, where $n = 209$. As mentioned in Section 5.7.1, we use the standardized values of the covariate dataset in this analysis. We fit the proposed model and all the comparison models (see Section 5.7.1) to this dataset and discuss the numerical results. For each model, we provide the CLS estimates and the RMSEs (between observed and predicted values) in Table 5.6. From the numerical results given in Table 5.6, it is seen that the smallest RMSE value is obtained for the proposed method.

To study the predictive performances based on the real data, we partition the data into two sets: a training set of 150 observations to fit the proposed model and the comparison models, and a test set of remaining 59 observations to find PRMSE(h), PTP(h) and PTPI(h) (where $\delta = 2$) measures for $h = 1, 2, 3, 4$. The results are given in Table 5.7, where we notice that the proposed model performs relatively well. The next closest competitor of the proposed model regarding the forecasting performances is the comparison model 1. This is expected due to the seasonal structure the comparison model 1 has, is similar to that of the proposed method. However, the other comparison model ZINAR(1) also presents some promising results when PTPI(h) is considered.

In this study, the proposed model generally performs better than the other comparison models. As mentioned earlier, the next best model is the comparison model 1 that also records some promising results, especially in the predictive performances. The possible reason for this phenomenon can be described as follows. The weekly data of dengue cases involves some large counts. Our proposed zero-inflated model may not be able to handle the large counts, compared to the comparison model 1 that involves no zero-inflation property. Having no zero-inflation property provides the comparison model 1 a relative edge over the proposed model in order to have more weights towards the higher values. However, overall, the proposed method accomplishes better numerical results, even if the presence of some large counts in this data is not ideal for the construction of

the proposed model. This clearly indicates why blending both zero-inflation and seasonal properties in our proposed PINAR model can be very advantageous for analyzing this kind of count time series.

TABLE 5.6: Estimated parameters and RMSEs for all the models

Models	CLS estimates	RMSE
Proposed model	$(\hat{\alpha}_{cls}, \hat{\rho}_{cls}, \hat{\beta}^{s}_{1cls}, \hat{\beta}^{s}_{2cls}, \hat{\beta}^{s}_{3cls}) = (0.7051, 0.6782, 2.6863, -2.9152, 0.9326)$	8.19
Comparison model 1	$(\hat{\alpha}_{cls}, \hat{\beta}^{s}_{1cls}, \hat{\beta}^{s}_{2cls}, \hat{\beta}^{s}_{3cls}) = (0.6840, 2.0223, -1.9930, 0.8364)$	8.22
ZINAR(1)	$(\hat{\alpha}_{cls}, \hat{\rho}_{cls}, \hat{\lambda}_{cls}) = (0.9041, 0.9054, 14.6958)$	9.4

TABLE 5.7: Comparative forecasting study between all the models

h	Proposed model			Comparison model 1			ZINAR(1)		
	PRMSE	PTP	PTPI	PRMSE	PTP	PTPI	PRMSE	PTP	PTPI
1	7.66	23.73	55.93	7.65	23.73	55.93	7.57	8.47	49.15
2	10.94	20.69	51.72	10.98	22.41	51.72	10.78	5.17	27.59
3	12.76	28.07	49.12	12.76	26.32	49.12	12.91	1.75	8.77
4	13.89	23.21	48.21	13.90	19.64	51.79	15.64	5.36	7.14

5.9 Conclusions

The usual PINAR process is unable to model count data having both zero-inflation and seasonality. But such features can be observed in many real-life count time series datasets, like the data of weekly dengue cases in Kaohsiung City, Taiwan, from 2009 to 2012. Hence, in this chapter, we have proposed a PINAR model based on binomial thinning operator, which can get hold of both zero-inflation and seasonality properties together in a single setup. We have also included the weekly maximum temperature data in our proposed process, along with the sine-cosine functions to model the innovations. The inclusion of the temperature data has been done to incorporate the effect of temperature in our proposed model, as it is a very important factor directly impacting the worldwide spread of dengue infections (see World Health Organization, n.d.). This idea is mainly based on Freeland and McCabe, 2004b. Therefore, as discussed earlier, this study regarding our proposed PINAR with zero-inflation and seasonality can be viewed as a case study in the field of count time series.

The zero-inflated PINAR process with seasonality has been developed mainly borrowing ideas from Jazi, Jones, and Lai, 2012a, Chan, Hu, and Hwang, 2015,

and Chattopadhyay et al., 2021. To capture the zero-inflation feature in the proposed model, the study of Jazi, Jones, and Lai, 2012a has been consulted throughout the chapter. As previously discussed, we have implemented an approximation of the full seasonal model using sine-cosine functions, based on the correspondence between the seasonal dummy model and the seasonal model with sine-cosine functions.

Throughout this chapter, we have studied the distributional and the h-step ahead forecasting features of the proposed model. We have employed the CLS method in order to estimate the model parameters. We have conducted an extensive simulation study to confirm the empirical consistency of the estimated model parameters. The data application has shown our proposed method to be a more favourable model for this kind of count time series data than the other INAR models with respect to standard statistical criteria like RMSE and forecasting measures like PRMSE, PTP, and PTPI.

However, as mentioned earlier, the proposed setup for the zero-inflated PINAR process with seasonality has the limitation how many lag values should be included in the model after the inclusion of time-varying seasonal components along with the zero-inflation property. This is a difficult problem to address and needs further study, which we intend to pursue in future work. But the proposed PINAR(1) process has shown promising results, both in terms of the simulation experiments and the data application. Therefore, we can hope that the proposed zero-inflated PINAR(1) process with seasonality, even with this limitation, would be a useful option to analyze this kind of count time series data based on the theoretical and numerical results obtained throughout this chapter.

5.10 Appendix

Appendix A. Proof of result 5.1

In the INAR(1) model, we have

$$Y_n = \alpha \circ Y_{n-1} + \epsilon_n = \sum_{i=1}^{Y_{n-1}} N_{1i} + \epsilon_n,$$

where $\{N_{1i}\}$ is a sequence of i.i.d. Bernoulli(α) random variables, independent of Y_{n-1}, and we know that given $\mathcal{X}_n^s$, $\alpha \circ Y_{n-1}$ and ε_n are independent. From Wald's identities, we can say

$$E(Y_n|\mathcal{X}_n^s) = E(N_{1i})E(Y_{n-1}|\mathcal{X}_n^s) + E(\varepsilon_n|\mathcal{X}_n^s), \tag{5.10.1}$$

and

$$V(Y_n|\mathcal{X}_n^s) = V(N_{1i})E(Y_{n-1}|\mathcal{X}_n^s) + (E(N_{1i}))^2 V(Y_{n-1}|\mathcal{X}_n^s) + V(\varepsilon_n|\mathcal{X}_n^s). \quad (5.10.2)$$

Here, we have (see Section 5.4)

$$E(Y_1|\mathcal{X}_1^s) = \lambda_1^s(1-\rho). \quad (5.10.3)$$

From (5.10.3) and (5.10.1), we can write

$$E(Y_2|\mathcal{X}_2^s) = \alpha E(Y_1|\mathcal{X}_2^s) + \lambda_2^s(1-\rho) = (\alpha\lambda_1^s + \lambda_2^s)(1-\rho). \quad (5.10.4)$$

Now, we assume that

$$E(Y_{k-1}|\mathcal{X}_{k-1}^s) = (\alpha^{k-2}\lambda_1^s + \alpha^{k-3}\lambda_2^s + \ldots + \lambda_{k-1}^s)(1-\rho). \quad (5.10.5)$$

From (5.10.5), we can write

$$\begin{aligned} E(Y_k|\mathcal{X}_k^s) &= \alpha(\alpha^{k-2}\lambda_1^s + \alpha^{k-3}\lambda_2^s + \ldots + \lambda_{k-1}^s)(1-\rho) + \lambda_k^s(1-\rho) \\ &= (\alpha^{k-1}\lambda_1^s + \alpha^{k-2}\lambda_2^s + \ldots + \lambda_k^s)(1-\rho). \end{aligned}$$

This completes the proof for $E(Y_n|\mathcal{X}_n^s)$.
Now, using (5.10.2) and (5.10.3), and assuming $V(Y_1|\mathcal{X}_1^s) = \lambda_1^s(1-\rho)(1+\rho\lambda_1^s)$ (see Section 5.4), we can write

$$\begin{aligned} V(Y_2|\mathcal{X}_2^s) &= \alpha(1-\alpha)E(Y_1|\mathcal{X}_2^s) + \alpha^2 V(Y_1|\mathcal{X}_2^s) + V(\varepsilon_2|\mathcal{X}_2^s) \\ &= \alpha(1-\alpha)\lambda_1^s(1-\rho) + \alpha^2\lambda_1^s(1-\rho)(1+\rho\lambda_1^s) + \lambda_2^s(1-\rho)(1+\rho\lambda_2^s) \\ &= \alpha\lambda_1^s(1-\rho) - \alpha^2\lambda_1^s(1-\rho) + \alpha^2\lambda_1^s(1-\rho) + \alpha^2(\lambda_1^s)^2\rho(1-\rho) + \\ &\qquad \lambda_2^s(1-\rho) + \rho(1-\rho)(\lambda_t^s)^2 \\ &= (1-\rho)(\alpha\lambda_1^s + \lambda_2^s) + \rho(1-\rho)\left[(\alpha\lambda_1^s)^2 + (\lambda_2^s)^2\right]. \end{aligned}$$

Now, let us assume that

$$V(Y_{k-1}|\mathcal{X}_{k-1}^s) = (\alpha^{k-2}\lambda_1^s + \ldots + \lambda_{k-1}^s)(1-\rho) + \rho(1-\rho)\left[(\alpha^{k-2}\lambda_1^s)^2 + \ldots + (\lambda_{k-1}^s)^2\right].$$

So, we can write

$$\begin{aligned} V(Y_k|\mathcal{X}_k^s) &= \alpha(1-\alpha)(\alpha^{k-2}\lambda_1^s + \ldots + \lambda_{k-1}^s)(1-\rho) + \alpha^2(\alpha^{k-2}\lambda_1^s + \ldots + \lambda_{k-1}^s)(1-\rho) + \\ &\qquad \alpha^2\rho(1-\rho)\left[(\alpha^{k-2}\lambda_1^s)^2 + \ldots + (\lambda_{k-1}^s)^2\right] + \lambda_k^s(1-\rho)(1+\rho\lambda_k^s) \\ &= (\alpha^{k-1}\lambda_1^s + \ldots + \lambda_k^s)(1-\rho) + \rho(1-\rho)\left[(\alpha^{k-1}\lambda_1^s)^2 + \ldots + (\lambda_k^s)^2\right]. \end{aligned}$$

So, we have

$$V(Y_k|\mathcal{X}_k^s) = (\alpha^{k-1}\lambda_1^s + \ldots + \lambda_k^s)(1-\rho) + \rho(1-\rho)\left[(\alpha^{k-1}\lambda_1^s)^2 + \ldots + (\lambda_k^s)^2\right].$$

This completes the proof for $V(Y_n|\mathcal{X}_n^s)$.

Appendix B. Proof of result 5.2

The ACVF is given by

$$\begin{aligned} \gamma_y(h) &= Cov(Y_t, Y_{t+h}|\mathcal{X}_{t+h}^s) \\ &= E(Y_t\,Y_{t+h}|\mathcal{X}_{t+h}^s) - E(Y_t|\mathcal{X}_{t+h}^s)E(Y_{t+h}|\mathcal{X}_{t+h}^s) \\ &= E(Y_t\,E(Y_{t+h}|Y_t, \mathcal{X}_{t+h}^s)|\mathcal{X}_{t+h}^s) - E(Y_t|\mathcal{X}_{t+h}^s)E(Y_{t+h}|\mathcal{X}_{t+h}^s) \\ &= E\left[Y_t\left(\alpha^h Y_t + \sum_{i=1}^{h}\alpha^{h-i}\lambda_{t+i}^s(1-\rho)\right)\Big|\mathcal{X}_{t+h}^s\right] - \\ &\qquad E(Y_t|\mathcal{X}_{t+h}^s)E(Y_{t+h}|\mathcal{X}_{t+h}^s) \\ &= \alpha^h\left[V(Y_t|\mathcal{X}_{t+h}^s) + E^2(Y_t|\mathcal{X}_{t+h}^s)\right] + \sum_{i=1}^{h}\alpha^{h-i}\lambda_{t+i}^s(1-\rho)E(Y_t|\mathcal{X}_{t+h}^s) - \\ &\qquad E(Y_t|\mathcal{X}_{t+h}^s)E(Y_{t+h}|\mathcal{X}_{t+h}^s) \\ &= \alpha^h V(Y_t|\mathcal{X}_{t+h}^s) - E(Y_t|\mathcal{X}_{t+h}^s) \\ &\qquad \left[-\alpha^h E(Y_t|\mathcal{X}_{t+h}^s) - \sum_{i=1}^{h}\alpha^{h-i}\lambda_{t+i}^s(1-\rho) + E(Y_{t+h}|\mathcal{X}_{t+h}^s)\right] \\ &= \alpha^h V(Y_t|\mathcal{X}_{t+h}^s) - E(Y_t|\mathcal{X}_{t+h}^s)\left[-\alpha^h\left(\alpha^{t-1}\lambda_1^s + \ldots + \lambda_t^s\right)(1-\rho)\right] - \\ E(Y_t|\mathcal{X}_{t+h}^s)&\left[-\left(\alpha^{h-1}\lambda_{t+1}^s + \ldots + \lambda_{t+h}^s\right)(1-\rho) + \left(\alpha^{t+h-1}\lambda_1^s + \ldots + \lambda_{t+h}^s\right)(1-\rho)\right] \\ &= \alpha^h(1-\rho)\left[\alpha^{t-1}\lambda_1^s + \ldots + \lambda_t^s\right] + \alpha^h\rho(1-\rho)\left[(\alpha^{t-1}\lambda_1^s)^2 + \ldots + (\lambda_t^s)^2\right]. \end{aligned}$$

This completes the proof.

Appendix C. Proof of result 5.3

We can write (see Chattopadhyay et al., 2021)

$$\Phi_{Y_n|\mathcal{X}_n^s}(s) = \Phi_{Y_{n-1}|\mathcal{X}_n^s}(1-\alpha+\alpha s)\Phi_{\varepsilon_n|\mathcal{X}_n^s}(s). \tag{5.10.6}$$

Now, using (5.10.6), we can write

$$\begin{aligned}
\Phi_{Y_n|\mathcal{X}_n^s}(s) &= \Phi_{Y_{n-1}|\mathcal{X}_n^s}(1-\alpha+\alpha s)\Phi_{\varepsilon_n|\mathcal{X}_n^s}(s) \\
&= \Phi_{Y_{n-2}|\mathcal{X}_n^s}(1-\alpha+\alpha(1-\alpha+\alpha s))\Phi_{\varepsilon_{n-1}|\mathcal{X}_n^s}(1-\alpha+\alpha s)\Phi_{\varepsilon_n|\mathcal{X}_n^s}(s) \\
&= \Phi_{Y_{n-2}|\mathcal{X}_n^s}(1-\alpha^2+\alpha^2 s)\Phi_{\varepsilon_{n-1}|\mathcal{X}_n^s}(1-\alpha+\alpha s)\Phi_{\varepsilon_n|\mathcal{X}_n^s}(s) \\
&\vdots \\
&= \Phi_{Y_1|\mathcal{X}_n^s}(1-\alpha^{n-1}+\alpha^{n-1}s)\Phi_{\varepsilon_2|\mathcal{X}_n^s}(1-\alpha^{n-2}+\alpha^{n-2}s)\ \ldots\ \Phi_{\varepsilon_n|\mathcal{X}_n^s}(s) \\
&= \left[\rho+(1-\rho)e^{-\lambda_1^s\alpha^{n-1}(1-s)}\right]\left[\rho+(1-\rho)e^{-\lambda_2^s\alpha^{n-2}(1-s)}\right]\ \ldots\ \left[\rho+(1-\rho)e^{-\lambda_n^s(1-s)}\right] \\
&= \prod_{i=0}^{n-1}\left[\rho+(1-\rho)e^{-\lambda_{n-i}^s\alpha^i(1-s)}\right].
\end{aligned}$$

This completes the proof.

Appendix D. Proof of $p_{0,n}^1 > p_{0,n}^2$

Let $z_i = \alpha^i\lambda_{n-i}^s$ which is > 0 for $i = 0, 1, \ldots$. We can write $[\rho+(1-\rho)e^{-z_i}]$ as a convex combination of $\{1, e^{-z_i}\}$ where $e^{-z_i} < 1$, and hence, $[\rho+(1-\rho)e^{-z_i}] > e^{-z_i}$. So, we have

$$\prod_{i=0}^{n-1}\left[\rho+(1-\rho)e^{-z_i}\right] > \prod_{i=0}^{n-1}e^{-z_i} \Rightarrow \prod_{i=0}^{n-1}\left[\rho+(1-\rho)e^{-\alpha^i\lambda_{n-i}^s}\right] > e^{-\sum_{i=0}^{n-1}\alpha^i\lambda_{n-i}^s}.$$

This completes the proof.

Appendix E. Correspondence between the seasonal dummy model and the seasonal model with sine and cosine terms for $s = 4$

For $s = 4$, if the four seasons are indexed by $i = 1, 2, 3, 4$, then the values from the year τ can be written as follows:

$$\begin{bmatrix} S_{\tau 1} \\ S_{\tau 2} \\ S_{\tau 3} \\ S_{\tau 4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \quad \text{for equation (5.3.4)},$$

$$\begin{bmatrix} S_{\tau 1} \\ S_{\tau 2} \\ S_{\tau 3} \\ S_{\tau 4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & -1 \\ 1 & -1 & 0 & 1 \\ 1 & 0 & -1 & -1 \\ 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ \alpha_1^{s*} \\ \beta_1^{s*} \\ \alpha_2^{s*} \end{bmatrix} \quad \text{for equation (5.3.5)},$$

since $(\cos\frac{\pi}{2}, \sin\frac{\pi}{2}, \cos\pi) = (0, 1, -1)$, $(\cos\pi, \sin\pi, \cos 2\pi) = (-1, 0, 1)$, $(\cos\frac{3\pi}{2}, \sin\frac{3\pi}{2}, \cos 3\pi) = (0, -1, -1)$, $(\cos 2\pi, \sin 2\pi, \cos 4\pi) = (1, 0, 1)$. Note that, the final sine term is always zero, since $\sin(\pi t) = 0$ for all t, and so, here $\beta_2^{s*} = 0$.
So, we have

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & -1 \\ 1 & -1 & 0 & 1 \\ 1 & 0 & -1 & -1 \\ 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ \alpha_1^{s*} \\ \beta_1^{s*} \\ \alpha_2^{s*} \end{bmatrix}. \tag{5.10.7}$$

From (5.10.7), we can write

$$\delta_1 = \mu + \beta_1^{s*} - \alpha_2^{s*}, \tag{5.10.8}$$

$$\delta_2 = \mu - \alpha_1^{s*} + \alpha_2^{s*}, \tag{5.10.9}$$

$$\delta_3 = \mu - \beta_1^{s*} - \alpha_2^{s*}, \tag{5.10.10}$$

$$\delta_4 = \mu + \alpha_1^{s*} + \alpha_2^{s*}. \tag{5.10.11}$$

By (5.10.8)+(5.10.9)+(5.10.10)+(5.10.11), we find

$$\mu = \frac{\sum_{i=1}^{4} \delta_i}{4}.$$

By (5.10.8)-(5.10.10), we find

$$\delta_1 - \delta_3 = 2\beta_1^{s*} \Rightarrow \beta_1^{s*} = \frac{\delta_1 - \delta_3}{2}.$$

From (5.10.10), we find

$$\frac{\sum_{i=1}^{4}\delta_i}{4} - \frac{2(\delta_1 - \delta_3)}{4} - \delta_3 = \alpha_2^{s*} \Rightarrow \alpha_2^{s*} = \frac{\delta_2 + \delta_4 - \delta_1 - \delta_3}{4}.$$

And, from (5.10.11), we find

$$\alpha_1^{s*} + \frac{\sum_{i=1}^{4}\delta_i}{4} + \frac{\delta_2 + \delta_4 - \delta_1 - \delta_3}{4} = \delta_4 \Rightarrow \alpha_1^{s*} = \frac{\delta_4 - \delta_2}{2}.$$

So, we have the correspondences: $\mu = \frac{\sum_{i=1}^{4}\delta_i}{4}$, $\alpha_1^{s*} = \frac{\delta_4-\delta_2}{2}$, $\alpha_2^{s*} = \frac{\delta_2+\delta_4-\delta_1-\delta_3}{4}$, $\beta_1^{s*} = \frac{\delta_1-\delta_3}{2}$, and $\beta_2^{s*} = 0$.

Appendix F. Some comments on standardization

In this study, we utilize the conventional standardization method, denoted as $\frac{x - mean_x}{sd_x}$, for both simulation study and data analysis. However, one limitation of employing various standardization methods is that we will obtain different estimates of model parameters across different techniques. In this section, we conduct a study utilizing the actual data (see Section 5.8) to demonstrate that, despite obtaining almost varying estimates for different standardization techniques, we will obtain almost identical numerical values for measures such as RMSE (for data fitting) and forecasting measures, which are PRMSE, PTP, and PTPI. Alternatively, we will observe minimal disparities in the numerical values obtained from each technique. For examination, we use four standardization methods, which are:

1. Standardization Method 1 (std1): $\frac{x - mean_x}{sd_x}$, which is used throughout our study.
2. Standardization Method 2 (std2): $\frac{x - min_x}{max_x - min_x}$.
3. Standardization Method 3 (std3): $\frac{x - mean_x}{max_x - min_x}$.
4. Standardization Method 4 (std4): $\frac{x - min_x}{sd_x}$.

The numerical results are given in Tables 5.8 and 5.9. From Table 5.8, we can see that, despite obtaining varying estimates for different standardization techniques, we obtain identical numerical values of RMSEs.

And from Table 5.9, we also obtain identical results except for the 3-step PRMSE value of standardization method 3 (std3), which is 10.90. However, this value is very close to other 3-step PRMSE values that are equal to 10.94. Therefore, as discussed earlier, despite the fact that we get varying estimates of some model parameters, we obtain similar numerical results for different standardization techniques. Therefore, we employ the usual standardization technique (std1) throughout our study.

TABLE 5.8: Estimated parameters and RMSEs for all the models using different standardization methods

Models	CLS estimates	RMSE
Model with std1	$(\hat{\alpha}_{cls}, \hat{\rho}_{cls}, \hat{\beta}^{s}_{1cls}, \hat{\beta}^{s}_{2cls}, \hat{\beta}^{s}_{3cls}) = (0.7051, 0.6782, 2.6863, -2.9152, 0.9326)$	8.19
Model with std2	$(\hat{\alpha}_{cls}, \hat{\rho}_{cls}, \hat{\beta}^{s}_{1cls}, \hat{\beta}^{s}_{2cls}, \hat{\beta}^{s}_{3cls}) = (0.7051, 0.9828, 2.6902, -2.9198, 4.3131)$	8.19
Model with std3	$(\hat{\alpha}_{cls}, \hat{\rho}_{cls}, \hat{\beta}^{s}_{1cls}, \hat{\beta}^{s}_{2cls}, \hat{\beta}^{s}_{3cls}) = (0.7050, 0.6780, 2.6860, -2.9150, 4.3086)$	8.19
Model with std4	$(\hat{\alpha}_{cls}, \hat{\rho}_{cls}, \hat{\beta}^{s}_{1cls}, \hat{\beta}^{s}_{2cls}, \hat{\beta}^{s}_{3cls}) = (0.7050, 0.9827, 2.6862, -2.9145, 0.9327)$	8.19

TABLE 5.9: Forecasting measures for all the models using different standardization methods

h	Model with std1			Model with std2			Model with std3			Model with std4		
	PRMSE	PTP	PTPI	PRMSE	PTP	PTPI	PRMSE	PTP	PTPI	PRMSE	PTP	PTPI
1	7.66	23.73	55.93	7.66	23.73	55.93	7.66	23.73	55.93	7.66	23.73	55.93
2	10.94	20.69	51.72	10.94	20.69	51.72	10.90	20.69	51.72	10.94	20.69	51.72
3	12.76	28.07	49.12	12.76	28.07	49.12	12.76	28.07	49.12	12.76	28.07	49.12
4	13.89	23.21	48.21	13.89	23.21	48.21	13.89	23.21	48.21	13.89	23.21	48.21

Appendix G. Some simulated covariate datasets

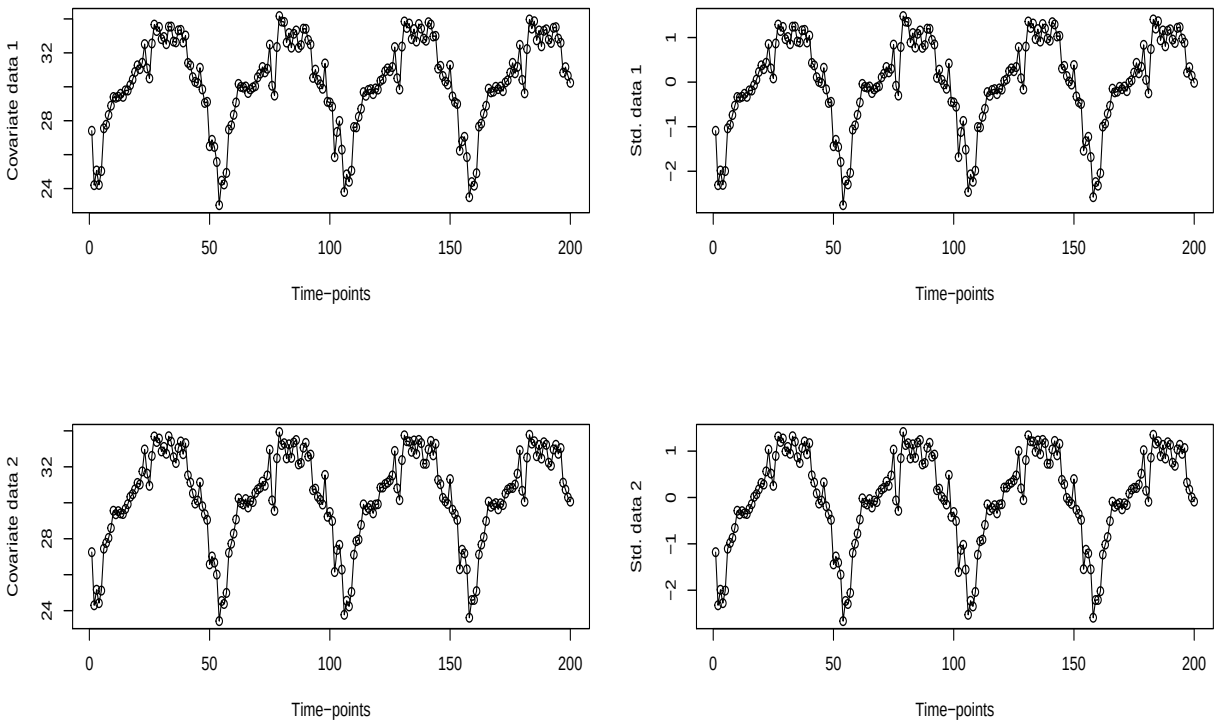


FIGURE 5.5: Some samples of simulated covariate datasets and their standardized values

Chapter 6

Epilogue

Overall, the thesis discusses problems in discrete-valued time series analysis (see Weiß, 2018) with applications to some real datasets. As discussed throughout the thesis, discrete-valued time series are divided into two classes: (i) categorical time series and (ii) count time series. We employed the Pegram's operator based autoregressive (PAR) process to analyze categorical time series in Chapter 2 and truncated counts in Chapter 3. To investigate count time series with time-varying covariates, we considered the integer-valued autoregressive (INAR) process with the binomial thinning operator.

In Chapter 2, we studied the problem of modelling categorical time series with application to AQI data. In this study, we proposed a generalized PAR process of order 1 (GPAR(1)) to address the issue that the PAR(1) process has with the indicator kernel. The indicator kernel provides weights only when the previous category occurs at the current time-point. To address the drawback, we proposed a generalized kernel that assigns weight (possibly different) to all possible cases. For the comparison study, we used the standard PAR(1) process and the mixture transition distribution model of order 1 (MTD(1) model). For a practical example, we analyzed the AQI data using our proposed GPAR(1) method. Then, in Chapter 3, we addressed the issue of analyzing time series with truncated counts. To do so, we proposed a modified PAR(1) (mPAR(1)) process with a modified kernel to address the aforementioned drawback, which also exists for the PAR(1) process when modelling truncated counts. Finally, we analyzed a real dataset through our proposed mPAR(1) process. We can further extend these proposed methods to orders greater than 1.

Chapters 4 and 5 addressed the challenges of modelling count time series datasets with time-varying covariates. In Chapter 4, we discussed count time

series with change-points that can be observed in real datasets, such as various COVID-19 datasets. In Chapter 5, we studied count time series with zero-inflation and seasonality, which we can notice in some datasets, such as dengue data for a specific location. To analyze these types of count time series, we proposed appropriately adjusted INAR processes of order 1 (INAR(1)) with time-varying components using the binomial thinning operator. This INAR process can capture the time-varying properties that some count time series exhibit while preserving the survival component. We looked at some real datasets to see how our proposed models worked in practice. In this direction, we wish to look into this proposed INAR process with order greater than 1 and also study count time series with other types of time-varying features.

Throughout this thesis, we developed various distributional properties and investigated h-step ahead forecasting for our proposed processes. We conducted extensive simulation experiments to determine the usefulness of the proposed processes. Then, to provide illustrations of the proposed methods, we analyzed some real datasets. In this regard, we considered some existing models to conduct detailed comparison studies with our proposed methods in both simulation studies and data analyses. We primarily used three forecasting measures (PRMSE, PTP, and PTPI) for both simulation studies and data analyses. Based on all of the theoretical and numerical results from our studies, we hope that the proposed methods will be viable options in the field of discrete-valued time series analysis.

List of Publication(s)/Pre-print(s)

Published

1. **Chattopadhyay, S.**, Maiti, R., Das, S., and Biswas, A. (2021), Change-point analysis through integer-valued autoregressive process with application to some COVID-19 data. *Statistica Neerlandica*, 76(1), 4-34.

Pre-prints

1. **Chattopadhyay, S.**, Biswas, A., Das, S., Maiti, R., and Hwang, J.-S. (2023), Analysis of count time series through integer-valued autoregressive (INAR) process with zero-inflation and seasonality.
2. **Chattopadhyay, S.**, Maiti, R., Das, S., and Biswas, A. (2024), A generalized Pegram's operator based autoregressive (GPAR) process for modelling categorical time series.
3. **Chattopadhyay, S.**, Biswas, A., Das, S., and Maiti, R. (2024), A modified Pegram's operator based autoregressive (mPAR) process for modelling truncated counts.

Bibliography

Abeysinghe, T. (1994). "Forecasting performance of seasonal-dummy models relative to some alternatives". In: *Economics Letters* 44.4, pp. 365–370.

Al-Osh, M. and E.-E. Aly (1992). "First order autoregressive time series with negative binomial and geometric marginals". In: *Communications in Statistics - Theory and Methods* 21.9, pp. 2483–2492.

Al-Osh, Mohamed A and Aus A Alzaid (1987). "First-order integer-valued autoregressive (INAR (1)) process". In: *Journal of Time Series Analysis* 8.3, pp. 261–275.

Angers, J.-F. and A. Biswas (2003). "A Bayesian analysis of zero-inflated generalized Poisson model". In: *Computational Statistics & Data Analysis* 42.1-2, pp. 37–46.

AQI IN 1 (n.d.). URL: https://aqicn.org/faq/2015-05-15/india-national-air-quality-index/.

AQI IN 2 (n.d.). URL: https://en.wikipedia.org/wiki/Air_quality_index#India.

Berchtold, André and Adrian Raftery (2002). "The Mixture Transition Distribution (MTD) Model for High-Order Markov Chains and Non-Gaussian Time Series". In: *Statistical Science* 17.3, pp. 328–356.

Biswas, Atanu, Maria del Carmen Pardo, and Apratim Guha (2014). "Auto-association measures for stationary time series of categorical data". In: *Test* 23, pp. 487–514.

Biswas, Atanu and Apratim Guha (2009). "Time series analysis of categorical data using auto-mutual information". In: *Journal of Statistical Planning and Inference* 139.9, pp. 3076–3087.

Biswas, Atanu and Peter X.-K. Song (2009). "Discrete-valued ARMA processes". In: *Statistics & Probability Letters* 79.17, pp. 1884–1889.

Böhning, D. et al. (1999). "The zero-inflated Poisson model and the decayed, missing and filled teeth index in dental epidemiology". In: *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 162.2, pp. 195–209.

Bourguignon, M. (2018). "Modelling time series of counts with deflation or inflation of zeros". In: *Statistics and Its Interface* 11, pp. 631–639.

Bourguignon, M. et al. (2016). "A Poisson INAR(1) process with a seasonal structure". In: *Journal of Statistical Computation and Simulation* 86.2, pp. 373–387.

Box, George.E.P. and Gwilym M. Jenkins (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day.

Brockwell, Peter J and Richard A Davis (2002). *Introduction to time series and forecasting*. Springer.

Buteikis, A. and R. Leipus (2020). "An integer-valued autoregressive process for seasonality". In: *Journal of Statistical Computation and Simulation* 90.3, pp. 391–411.

Cameron, A. C. and P. K. Trivedi (1986). "Econometric Models Based on Count Data: Comparisons and Applications of Some Estimators and Tests". In: *Journal of Applied Econometrics* 1.1, pp. 29–53.

Chakraborty, B., E. B. Laber, and Y. Zhao (2013). "Inference for optimal dynamic treatment regimes using an adaptive *m*-out-of-*n* bootstrap scheme". In: *Biometrics* 69.3, pp. 714–723.

Chan, K. S. and H. Tong (1986). "On estimating thresholds in autoregressive models". In: *Journal of Time Series Analysis* 7.3, pp. 179–190.

Chan, T.C., T.H. Hu, and J.S. Hwang (2015). "Daily forecast of dengue fever incidents for urban villages in a city". In: *International Journal of Health Geographics* 14.9.

Chattopadhyay, Subhankar et al. (2021). "Change-point analysis through INAR process with application to some COVID-19 data". In: *Statistica Neerlandica* 76.1, pp. 4–34.

Chattopadhyay, Subhankar et al. (2023). "Analysis count time series through integer-valued autoregressive (INAR) process with zero-inflation and seasonality". In: *pre-print*.

Chattopadhyay, Subhankar et al. (2024a). "A generalized Pegram's operator based autoregressive (GPAR) process for modelling categorical time series". In: *pre-print*.

Chattopadhyay, Subhankar et al. (2024b). "A modified Pegram's operator based autoregressive (mPAR) process for modelling truncated counts". In: *pre-print*.

Cleveland, William S, Eric Grosse, and William M Shyu (2017). "Local regression models". In: *Statistical models in S*. Routledge, pp. 309–376.

Davis, Richard A, William TM Dunsmuir, and Sarah B Streett (2003). "Observation-driven models for Poisson counts". In: *Biometrika* 90.4, pp. 777–790.

— (2005). "Maximum likelihood estimation for an observation driven model for Poisson counts". In: *Methodology and Computing in Applied Probability* 7, pp. 149–159.

Fahrmeir, Ludwig and Heinz Kaufmann (1987). "Regression models for non-stationary categorical time series". In: *Journal of time series Analysis* 8.2, pp. 147–160.

Fokianos, K. (2011). "Some recent progress in count time series". In: *A Journal of Theoretical and Applied Statistics* 45.1, pp. 49–58.

Fokianos, Konstantinos and Benjamin Kedem (1998). "Prediction and classification of non-stationary categorical time series". In: *Journal of multivariate analysis* 67.2, pp. 277–296.

— (2003). "Regression theory for categorical time series". In: *Statistical science* 18.3, pp. 357–376.

Fong, Y. et al. (2017). "chngpt: threshold regression model estimation and inference". In: *BMC Bioinformatics* 18.

Freeland, R. K. and B. McCabe (2004a). "Forecasting discrete valued low count time series". In: *International Journal of Forecasting* 20.3, pp. 427–434.

— (2005). "Asymptotic properties of CLS estimators in the Poisson AR(1) model". In: *Statistics & Probability Letters* 73.2, pp. 147–153.

Freeland, R Keith (1998). "Statistical analysis of discrete time series with application to the analysis of workers' compensation claims data". PhD thesis. University of British Columbia.

Freeland, R Keith and Brendan PM McCabe (2004b). "Analysis of low count time series data by Poisson autoregression". In: *Journal of time series analysis* 25.5, pp. 701–722.

Gaver, Donald P and Peter AW Lewis (1980). "First-order autoregressive gamma sequences and point processes". In: *Advances in Applied Probability* 12.3, pp. 727–745.

GoK Dashboard (n.d.). URL: https://dashboard.kerala.gov.in/index.php.

Hansen, B. E. (2000). "Sample Splitting and Threshold Estimation". In: *Econometrica* 68.3, pp. 575–603.

Heagerty, Patrick J. and Scott L. Zeger (1998). "Lorelogram: A Regression Approach to Exploring Dependence in Longitudinal Categorical Responses". In: *Journal of the American Statistical Association* 93.441, pp. 150–162.

Hyndman, Rob J and Yeasmin Khandakar (2008). "Automatic time series forecasting: the forecast package for R". In: *Journal of statistical software* 27, pp. 1–22.

Jacobs, P. and P. A. W. Lewis (1983). "Stationary Discrete Autoregressive Moving Average Time-Series Generated by Mixtures". In: *Journal of Time Series Analysis* 4, pp. 19–36.

Jacobs, P. A. and P. A. W. Lewis (1978a). "Discrete Time Series Generated by Mixtures. I: Correlational and Runs Properties". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 40.1, pp. 94–105.

Jacobs, Patricia A. and Peter A. W. Lewis (1978b). "Discrete Time Series Generated by Mixtures II: Asymptotic Properties". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 40.2, pp. 222–228.

— (1978c). *Discrete time series generated by mixtures III: Autoregressive processes (DAR(p))*. Tech. rep.

James, Gareth et al. (2013). *An Introduction to Statistical Learning: with Applications in R*. Springer-Verlag New York.

Jazi, M. A., G. Jones, and C.-D. Lai (2012a). "First-order integer valued AR processes with zero inflated Poisson innovations". In: *Journal of Time Series Analysis* 33.6, pp. 954–963.

— (2012b). "Integer valued AR(1) with geometric innovations". In: *Journal of the Iranian Statistical Society JIRSS* 2.

Kaufmann, Heinz (1987). "Regression models for nonstationary categorical time series: asymptotic estimation theory". In: *The Annals of Statistics*, pp. 79–98.

Khoo, Wooi Chen, Seng Huat Ong, and Atanu Biswas (2017). "Modeling time series of counts with a new class of INAR (1) model". In: *Statistical Papers* 58, pp. 393–416.

Lambert, D. (1992). "Zero-Inflated Poisson Regression, With an Application to Defects in Manufacturing". In: *Technometrics* 34.1, pp. 1–14.

Lovell, M. C. (1963). "Seasonal Adjustment of Economic Time Series and Multiple Regression Analysis". In: *Journal of the American Statistical Association* 58.304, pp. 993–1010.

Maiti, R. and A. Biswas (2015a). "Coherent forecasting for stationary time series of discrete data". In: *AStA Advances in Statistical Analysis* 99.3, pp. 337–365.

Maiti, Raju and Atanu Biswas (2015b). "Coherent forecasting for over-dispersed time series of count data". In: *Brazilian Journal of Probability and Statistics* 29.4, pp. 747–766.

— (2018). "Time series analysis of categorical data using auto-odds ratio function". In: *Statistics* 52.2, pp. 1–19.

Maiti, Raju, Atanu Biswas, and Bibhas Chakraborty (2018). "Modelling of low count heavy tailed time series data consisting large number of zeros and ones". In: *Statistical Methods & Applications* 27, pp. 407–435.

Maiti, Raju, Atanu Biswas, and Samarjit Das (2015). "Time series of zero-inflated counts and their coherent forecasting". In: *Journal of Forecasting* 34.8, pp. 694–707.

Maiti, Raju et al. (2014). "Modelling and coherent forecasting of zero-inflated count time series". In: *Statistical Modelling* 14.5, pp. 375–398.

McKenzie, E. (1985). "Some simple models for discrete variate time series". In: *Journal of the American Water Resources Association* 21.4, pp. 645–650.

McKenzie, Ed (1986). "Autoregressive moving-average processes with negative-binomial and geometric marginal distributions". In: *Advances in Applied Probability* 18.3, pp. 679–705.

— (1988). "Some ARMA models for dependent sequences of Poisson counts". In: *Advances in Applied Probability* 20.4, pp. 822–835.

McKenzie, Eddie (2003). "Ch. 16. discrete variate time series". In: *Handbook of statistics* 21, pp. 573–606.

Moriña, David et al. (2011). "A statistical model for hospital admissions caused by seasonal diseases". In: *Statistics in medicine* 30.26, pp. 3125–3136.

Moritz, S. and T. Bartz-Beielstein (2017). "imputeTS: Time Series Missing Value Imputation in R". In: *The R Journal* 9.1, pp. 207–218.

Mumbai AQI Data (n.d.). URL: https://www.aqi.in/dashboard/india/maharashtra/mumbai.

Naumova, E. N. et al. (2007). "Seasonality in six enterically transmitted diseases and ambient temperature". In: *Epidemiology and infection* 135.2, pp. 281–292.

Pegram, G. G. S. (1980). "An Autoregressive Model for Multilag Markov Chains". In: *Journal of Applied Probability* 17.2, pp. 350–362.

Peng, R. D., F. Dominici, and T.A. Louis (2006). "Model choice in time series studies of air pollution and mortality". In: *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 169.2, pp. 179–203.

Porter, M. D. and G. White (2012). "Self-exciting hurdle models for terrorist activity". In: *The Annals of Applied Statistics* 6.1, pp. 106–124.

Prezotti-Filho, P.L. et al. (2021). "A periodic and seasonal statistical model for non-negative integer-valued time series with an application to dispensed medications in respiratory diseases". In: *Applied Mathematical Modelling* 96, pp. 545–558.

Qian, Lianyong, Qi Li, and Fukang Zhu (2020). "Modelling heavy-tailedness in count time series". In: *Applied Mathematical Modelling* 82, pp. 766–784.

Raftery, Adrian E. (1985). "A Model for High-Order Markov Chains". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 47.3, pp. 528–539.

Ramanathan, K. et al. (2020). "Assessing Seasonality Variation with Harmonic Regression: Accommodations for Sharp Peaks". In: *Int J Environ Res Public Health* 17.4.

Ristić, M. M., H. S. Bakouch, and A. S. Nastić (2009). "A new geometric first-order integer-valued autoregressive (NGINAR(1)) process". In: *Journal of Statistical Planning and Inference* 139.7, pp. 2218–2226.

Ristić, M. M., A. S. Nastić, and H. S. Bakouch (2012). "Estimation in an Integer-Valued Autoregressive Process with Negative Binomial Marginals (NBINAR(1))". In: *Communications in Statistics - Theory and Methods* 41.4, pp. 606–618.

Ristić, Miroslav M, Hassan S Bakouch, and Aleksandar S Nastić (2009). "A new geometric first-order integer-valued autoregressive (NGINAR (1)) process". In: *Journal of Statistical Planning and Inference* 139.7, pp. 2218–2226.

Schweer, S. and C. H. Weiß (2014). "Compound Poisson INAR(1) processes: Stochastic properties and testing for overdispersion". In: *Computational Statistics & Data Analysis* 77, pp. 267–284.

Shirozhan, M, M Mohammadpour, and Hassan S Bakouch (2019). "A new geometric INAR (1) model with mixing Pegram and generalized binomial thinning operators". In: *Iranian Journal of Science and Technology, Transactions A: Science* 43, pp. 1011–1020.

Sim, Chiaw-Hock (1990). "First-order autoregressive models for gamma and exponential processes". In: *Journal of Applied Probability* 27.2, pp. 325–332.

Smooth Maximum (n.d.). URL: https://en.wikipedia.org/wiki/Smooth_maximum.

Song, Peter X.-K. et al. (2013). "Statistical analysis of discrete-valued time series using categorical ARMA models". In: *Computational Statistics & Data Analysis* 57.1, 112–124.

Steutel, F. and K. Van Harn (1979). "Discrete analogues of self-decomposability and stability". In: *The Annals of Probability* 7.5, pp. 893–899.

Stoffer, David S et al. (1988). "A Walsh—Fourier Analysis of the Effects of Moderate Maternal Alcohol Consumption on Neonatal Sleep-State Cycling". In: *Journal of the American Statistical Association* 83.404, pp. 954–963.

Stolwijk, A. M., H. Straatman, and G. A. Zielhuis (1999). "Studying seasonality by using sine and cosine functions in regression analysis". In: *Journal of epidemiology and community health* 53.4, pp. 235–238.

Tian, S., D. Wang, and S. Cui (2020). "A seasonal geometric INAR process based on negative binomial thinning operator". In: *Statistical Papers* 61.6, pp. 2561–2581.

Wald, A. and J. Wolfowitz (1940). "On a Test Whether Two Samples are from the Same Population". In: *The Annals of Mathematical Statistics* 11.2, pp. 147–162.

Wang, P. (2001). "Markov zero-inflated Poisson regression models for a time series of counts with excess zeros". In: *Journal of Applied Statistics* 28.5, pp. 623–632.

Weiß, Christian and Rainer Göb (2008). "Measuring serial dependence in categorical time series". In: *AStA Advances in Statistical Analysis* 92.1, pp. 71–89.

Weiß, Christian H (2011a). "Detecting mean increases in Poisson INAR (1) processes with EWMA control charts". In: *Journal of Applied Statistics* 38.2, pp. 383–398.

— (2011b). "Empirical measures of signed serial dependence in categorical time series". In: *Journal of Statistical Computation and Simulation* 81.4, pp. 411–429.

— (2013). "Serial dependence of NDARMA processes". In: *Computational Statistics & Data Analysis* 68, pp. 213–238.

— (2018). *An introduction to discrete-valued time series*. John Wiley & Sons.

Weiß, Christian H and Osama Swidan (2024). "Weighted discrete ARMA models for categorical time series". In: *Journal of Time Series Analysis*.

World Health Organization (n.d.). URL: https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue.

Worldometer (n.d.). URL: https://www.worldometers.info/coronavirus/country/italy/.

You, Shu-Han et al. (2022). "Dengue meteorological determinants during epidemic and non-epidemic periods in Taiwan". In: *Tropical Medicine and Infectious Disease* 7.12, p. 408.

Zeger, Scott L (1988). "A regression model for time series of counts". In: *Biometrika* 75.4, pp. 621–629.

Zheng, H., I. V. Basawa, and S. Datta (2007). "First-order random coefficient integer-valued autoregressive processes". In: *Journal of Statistical Planning and Inference* 137.1, pp. 212–229.

Zhu, Rong and Harry Joe (2006). "Modelling count data time series with Markov processes based on binomial thinning". In: *Journal of Time Series Analysis* 27.5, pp. 725–738.