Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

School of Computer Science and Statistics

Distributed Lag Regression Methods and Compartmental Models for Analysis of Disease Progression

Candidate: Daniel Dempsey

Supervisors: Jason Wyse and Mark Little

April 21, 2023

A dissertation submitted in partial fulfilment

of the requirements for the degree of

PhD (Statistics)

Declaration

I declare that this thesis has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work.

I agree to deposit this thesis in the University's open access institutional repository or allow the Library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

I consent to the examiner retaining a copy of the thesis beyond the examining period, should they so wish (EU GDPR May 2018).

Signed: _____ Date: _____

Summary

Here we present a summary of this thesis, included as per the TCD thesis guidelines found here. What follows is a very brief outline of each chapter:

- Chapter 1: Introduction. In this chapter we introduce the main research problem we are attempting to address, namely the development of a model that can detect a correlation between ANCA vasculitis flare propensity and environmental exposure.
- Chapter 2: This chapter discusses the background and literature behind the methods that we will use in later chapters of the thesis.
- Chapter 3: A more in-depth look at fitting MIDAS models and how to generalise them for binary response data. We perform a simulation study and find that the inferential methods usually perform well for large enough sample sizes, though can fail under certain erroneous modelling choices.
- Chapter 4: Here we discuss a Bayesian implementation of the MIDAS model that performs variable selection and accounts for binary response imbalance. This model is the primary novel contribution of this thesis. We perform a simulation study to examine how effective it is at selecting the correct covariate set. We apply this method to ANCA vasculitis flare data, which was the motivating problem for pursuing this research. In the end we find that no evidence that the given environmental covariates are predictive of flare event propensity.
- Chapter 5: We discuss research conducted in collaboration with statisticians from the University of Limerick to estimate the effects of lockdown measures during the COVID–19 pandemic in Dublin, Ireland. The novel aspect of this research was that our model attempted to account for the heterogeneous nature of

social mixing between/within age groups. The effects of lockdown are estimated, with bootstrapped uncertainty intervals, and the results are used to showcase a prototype app that can project the growth of COVID–19 under differing lockdown conditions.

- Chapter 6: Overall conclusion and discussion of potential avenues for future research.

- Appendix: There are three appendices. Appendix A is simply a step-by-step algorithm for the Bayesian MCMC method from chapter 4. Appendix B provides alternate results for the flare data analysis from section 4.6 under different assumptions about the onset of flare data, to examine the sensitivity of the results in the main analysis. Appendix C is a brief overview of code we built to implement the methods shown in chapters 3 and 4.

Project code is available in the following Github link: `https://github.com/DanDempsey/DD_Thesis_Files`. We have tried to be as transparent as possible, but for reasons of medical privacy we cannot share the clinical data that we used in our analysis.

Abstract

ANCA vasculitis is an autoimmune disease characterised by relapses, or *flares*, that can have a severe detrimental impact on patient health. Flares can be prevented by suppressing the immune system but this exposes the patient to infection. It is hard to prepare patients for flares since clinicians are still unclear on how to predict flare events. Some attention has been given on uncovering any environmental predictors but so far results have been inconclusive. Investigating this for ourselves is the main focus of this thesis.

We construct a distributed lag / MIDAS model to analyse the accumulation of environmental exposure over time in a parsimonious manner, and how that may impact the probability of a flare occurring. Our model employs Bayesian variable selection and adjustment for imbalanced response data using latent variable representation and reversible-jump MCMC. The construction of this model is the primary novel contribution of this thesis.

The method is validated via simulation study, and then applied to real data comprising of clinical information for flare events and satellite data that tracks weather and pollution indices for the region of residence of each patient. Despite our focus on vasculitis, we believe this model is applicable to many similar research problems.

We also look at a compartmental model to estimate the effect of lockdowns of combating the COVID–19 pandemic in Dublin, Ireland. The compartments are split into age groups and the flow between/within each compartment is adjusted to account for non-homogeneous age mixing between/within age groups. Uncertainty estimates are constructed using parametric bootstraps. With these, we can create projections of compartmental growth under different lockdown measures; a proof-of-concept app is

discussed to demonstrate this.

Acknowledgements

I will start with academic acknowledgements, the most important of which is of course my supervisor Jason Wyse. Jason has been a unending fountain of knowledge, patience and encouragement throughout my whole course. I consider myself very lucky to have had him as a supervisor - he's honestly the best teacher I've ever had.

I owe a great deal of thanks to the rest of the Department of Statistics in Trinity for building a very helpful and kind community of researchers. I encountered a lot of difficulty during my PhD but never failed to find support with my colleagues. I'd like to especially thank Arthur White who has become a frequent collaborator on side projects and another strong source of encouragement and advice.

I of course want to acknowledge the huge amount of help my co-supervisor Mark Little in providing clinical insight and for establishing the AVERT research group as a whole. It's because of him this research was even possible.

My thanks extends to the AVERT group as a whole, and there are a few I would like to give particular thanks to. Firstly, Jennifer Scott who, along with Mark, took on the extremely time-consuming task of poring over medical records one-by-one to retrospectively determine the flare status of patients. The analysis would have been impossible if not for that, and only people of their expertise could have done it. Secondly, I'd like to acknowledge Alan Meehan who was in charge of maintaining the database and building the original version of the GUI that we used to extract the necessary data. He was extremely prompt and helpful with any issues that I had with the data or the GUI, and always a pleasure to speak with. I'd also like to extend thanks to PatientMPower for their support early on in my PhD.

I would also like to thank the patients who agreed to take part in a focus group to help

me better understand their disease and inform some of the modelling decisions, as well as all those who helped me set up the focus group to begin with.

I also want to acknowledge that chapter 6 of this thesis is heavily based on work performed in conjunction with researchers from University of Limerick: James Sweeney, Kevin Burke, and Fatima-Zahra Jaouimaa. Fatima deserves special praise as she contributed as much, if not more, to the project than I did. I am very thankful that they gave me the opportunity to work on the project.

My biggest thanks must go to those who supported me outside of academia. My parents Lorna, Brian and Edward have encouraged me my whole life and provided for everything I ever needed. I can't possibly express how grateful I am to have them, and I hope they realize that if I ever accomplish anything worthwhile it is all thanks to them. This extends to my brothers Darren, Matthew and Niall, my grandparents Margaret, Christy and Rose, and the rest of my family.

Finally, I want to give special thanks my grandfather Tony Connors and my grandmother Myra O'Toole. My grandfather was an intelligent man who helped me with my maths homework when I struggled as a child. I often wonder if I would have the capability today to attempt a PhD in statistics if not for his guidance during those formative years. My grandmother was the most compassionate and loving person I have ever known, and I miss her dearly. I would like to devote this thesis to the both of them.

Funding Disclosure:

The majority of this research was funded by the Irish Research Council's Enterprise Partnership Scheme (website: https://research.ie/). The research conducted for chapter 5 was funded by Science Foundation Ireland (website: https://www.sfi.ie/).

Contents

1 Introduction

1.1 ANCA Vasculitis

Anti-Neutrophil Cytoplasmic Antibody (ANCA) associated vasculitis (Kitching et al., 2020; Yates and Watts, 2017) is an autoimmune disease that causes inflammation of blood vessels around the body due to the immune system erroneously attacking healthy cells potentially leading to death (Karangizi and Harper, 2018). Treatment against vasculitis involves a heavy dosage of immunosupressive treatment, called *induction* therapy, until the disease is forced into remission. At that point the patient is prescribed a lighter *maintenance* immunosuppressive therapy. Maintenance therapy is regulated so as to be strong enough to keep vasculitis in remission but weak enough that the immune system should remain functional. However there is a risk that vasculitis will relapse and the patient will suffer severe symptoms as a result; we call this a *flare* event.

This leads to a delicate balancing act; maintenance therapy is not strong enough to prevent serious illness from a flare event, but induction therapy suppresses the immune system to such a degree that the patient may be exposed to other infections. If we could predict these flare events then patients could be much better prepared to handle it, potentially averting harm. Research into possible genetic predictors has been conducted (for example McKinney et al. (2015); Chen et al. (2022); Mehta et al. (2022); Cho et al. (2021) among others). The AVERT group, an organization based in Trinity College Dublin and a strong collaborator of the research in this thesis (webpage: `https://www.tcd.ie/medicine/thkc/avert/`), was formed partly to explore the hypothesis that flare events occur as a reaction from exposure to bad air quality in their environment. Here we are restricting our discussion of the 'environment' to only

weather and pollution variables, but AVERT wishes to expand future research to other forms of environmental exposure. The attempt to uncover and understand the link between the environment and health is a common pursuit in science. Ter Horst et al. (2016) found that cytokine production (a protein important for facilitating a working immune system) is strongly seasonal. Dixon et al. (2019) leveraged smartphone data to find how the intensity of chronic pain changed with the weather. Warren et al. (2020a); Stojan et al. (2019); Lee (2018); Wilson et al. (2017a); Rushworth et al. (2014); Schwartz (2000) all examined how air pollution might adversely affect different health outcomes, and there are many more examples we could point to. The affect of the environment on the human body is considered an important component an the emerging field of science called the *exposome* (Miller, 2013). As for vasculitis specifically, there is strong evidence that suggests it seasonally correlated (Draibe et al., 2018). Scott et al. (2020) provide a very useful overview of the current literature surrounding the linkage between vasculitis and air quality, but it is still unclear what (if any) environmental factors significantly impact the probability of a patient suffering a flare event.

The specific purpose of our research is to investigate the correlation between flare propensity and environmental exposure. Our analysis is based off of electronic clinical records of vasculitis patients and satellite reanalysis of weather and pollution indices from the Copernicus Atmosphere Monitoring Service, or CAMS (Inness et al., 2019; Hersbach et al., 2020). The CAMS data is linked to each patient spatially (on a roughly county level granularity) based on, in order of preference where available, their smartphone geolocation (Beukenhorst et al., 2017), their home address, or the address of their hospital. The occurrence of a flare is coded as a binary variable based on information from corresponding hospital visits on those dates.

The methodology we employ revolves around the class of distributed lag models (DLMs). DLMs are regression models that assume the variation of the response is related to sustained, accumulated exposure of the covariates over some period of time. This is more complex than standard (contemporaneous) models as it not only requires estimation of the covariate effect sizes, but also how that effect is distributed over time. This problem can be approached in a number of ways: one idea is to simply assign a parameter to each of the lags (Schwartz, 2000). Another option is to impose a parametric form of the for the weights (Ghysels et al., 2002, 2006; Clements and Galvão, 2008; Foroni

et al., 2015; Mogliani and Simoni, 2020). We can also employ non-parametric methods (Gasparrini et al., 2010; Wilson et al., 2017a; Urban et al., 2021). These class of models are very flexible, but yet no established method to the best of our knowledge quite satisfies our needs for this project. Because of the rarity of flare events, our response variable is heavily imbalanced. We also have many environmental variables in our data set, most (if not all) of which we presume will not have any correlation with flare events. Establishing such a method is the main contribution of this thesis as we believe it represents a unique synthesis of distributed lag modelling, variable selection and rare event analysis that has not yet been seen in the literature. The reasoning, derivation and implementation of this methodology will make up the bulk of this thesis.

1.2 COVID–19

Over the course of my PhD, the world was swept up in the COVID–19 pandemic. Before vaccines were developed, and even for a while after, the response was to impose non-pharmaceutical interventions (NPIs), primarily lockdowns measures, to prevent social contact and inhibit spread of the virus. While lockdowns might be effective at reducing disease mobility, it deprives many of social interaction leading to deteriorating mental health (Kwong et al., 2020) and financial difficulty (Darmody et al., 2020). 'Softer' NPIs can be implemented to lessen the societal strain, but must be balanced out to prevent health systems from becoming overwhelmed. As a result countries increased and eased the intensity of lockdown as they felt appropriate given case and death monitoring, but in order to make truly informed decisions about what NPIs and properly assess risk between competing lockdown proposals we need to quantify the effectiveness of different NPIs. This is what we aimed to address in collaboration with statisticians based in the University of Limerick (Jaouimaa et al., 2021).

The method we employed was a Susceptible Exposed Infected Removed (SEIR) model. This is a system of ordinary differential equations (ODEs) that describes the rate of change of the different states of disease progression: from being uninfected but susceptible, to being exposed, to having your symptoms manifest, until finally being 'removed' (either recovery or death). How quickly a disease spreads depends on many factors, but one of the most important are the rates of social contact. However it is

common in these studies to simply average over the total population contact rates when in reality contacts will be age-heterogeneous (and likely heterogeneous in other ways also). It's reasonable to expect, for example, that people are more likely to spend time around others of the same age, and aggregating over this may mask important variation. We addressed this by incorporated age structuring in our model using contact matrices (Prem et al., 2017).

Since the main purpose of this project is to construct more informed projections, we developed an app that creates an 8 week forecast for compartment growth, estimated deaths and economic cost (also based on Jaouimaa et al. (2021), but that aspect of the research will not be discussed in this thesis). The app is written in the statistical programming language `R` version 4.1.1 (R Core Team, 2021) using the `shiny` package (Chang et al., 2022). The interactive visuals produced by the app are created using the `plotly` package (Sievert et al., 2020) in conjunction with the `ggplot2` package (Wickham, 2016).

1.3 Chapter Layout

Chapter 2 will introduce the methodology used in this thesis, and a review of the literature around the advancements and research around those methods. Chapter 3 will expand on the MIDAS model for generalised response data and frequentist inference of its parameters. Chapter 4 further expands the model from a Bayesian perspective. We use this model to analyse the flare data in this chapter also. In chapter 5 we discuss the application of an age-structured SEIR model in Jaouimaa et al. (2021) to estimate the impact of lockdowns on the spread of COVID–19 in Dublin. We also present a prototype of a `Shiny` app (Chang et al., 2022) that can make projections of disease spread under different lockdown rules based on our model estimates. Chapter 6 concludes the thesis and suggests avenues for additional research.

An appendix is given at the end that includes some extra information pertinent to the research. Appendix A provides a step-by-step algorithm of the Bayesian inference model derived in chapter 4. Appendix B provides some extra flare data analysis to examine how sensitive our results in section 4.6 are to the assumed flare onset date. Appendix C showcases some of the `R` code we used to implement the methods presented

in chapters 3 and 4.

2 Methods and Background

2.1 Binary Regression and Latent Variable Methods

2.1.1 Linear Regression

Regression models are flexible tools for estimation around how a set of response variables changes alongside the values of a set of predictor variables. The specifics of the relationship between predictors and response are imposed by the assumptions of the model; for example, linear regression assumes that the (one-dimensional) response is distributed as a normal distribution around a linear combination of the (possibly multi-dimensional) covariates, expressed as

$$y_i \sim \mathrm{N}\left(\mathbf{x}_i^\top \boldsymbol{\beta}, \sigma^2\right), i = 1, \cdots, n \tag{2.1}$$

where $y_i, i = 1, \cdots, n$ is the i^{th} response variable and $\mathbf{x}_i$ are the vector of covariate values for each explanatory variable corresponding to observation i. $\boldsymbol{\beta}$ are linear slope parameters and σ^2 is the variance of the residuals around the plane that is formed by the model. A common frequentist estimate of $\boldsymbol{\beta}$ (see for example section 1.2 of McCullagh (1983)) is the *least squares* estimate, the value of $\hat{\boldsymbol{\beta}}$ that minimises the sum of the squared residuals. This works out to be

$$\hat{\boldsymbol{\beta}}_{LS} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} \tag{2.2}$$

where $\mathbf{X}$ is the full matrix of predictors. In contrast, a Bayesian will treat the parameter itself as a random variable and the goal of the analysis is to find its distribution

conditioned on the observed data, $\pi(\boldsymbol{\beta}, \sigma^2|\mathbf{X}, \mathbf{y})$, called the posterior distribution. By Bayes theorem, the posterior distribution of the linear regression model is

$$\pi(\boldsymbol{\beta}, \sigma^2|\mathbf{X}, \mathbf{y}) = \frac{\pi(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2)\pi(\boldsymbol{\beta}, \sigma^2)}{\int\int \pi(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma)\pi(\boldsymbol{\beta}, \sigma^2)\mathrm{d}\boldsymbol{\beta}\mathrm{d}\sigma^2}. \tag{2.3}$$

$\pi(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2)$ is the likelihood, given by equation (2.1). $\pi(\boldsymbol{\beta}, \sigma)$ is the prior distribution, which is the probabilistic model for the analyst's state of beliefs about the parameters *before* observing the data (in contrast to the posterior, the state of beliefs *after* observing the data). In practice, the distributional family is often chosen in such a way that the integral in the denominator of the rhs of equation (2.3) can be resolved. The parameters of the prior distribution, called *hyperparameters*, are set to reflect to the available information about the modelling parameters a priori. If we have no prior information about the parameters we can use a non-informative prior, for example, an improper uniform prior on $\boldsymbol{\beta}$ and the log of σ^2. As a result the posterior becomes a Gaussian centered on the frequentist least squares estimate for $\boldsymbol{\beta}$. This prior essentially implies we truly know nothing about the parameters, so every value on the real line is equally likely. However, it is rarely the case that truly nothing is known about these parameters; even if we don't know where $\boldsymbol{\beta}$ may lie, we can at least rule out absurdly large estimates using the prior distribution. For example, we could set a Gaussian distribution as the prior for $\boldsymbol{\beta}$ centered at 0 and standard deviation equal to, say, 10. This means we are effectively ruling out values much larger than 20 in absolute value, which is often a reasonable assumption for covariates centered at 0 and scaled to unit standard deviation. Since the Gaussian is conjugate to itself, the resulting posterior is also Gaussian, centered at the frequentist ridge regression estimate (Hoerl and Kennard, 1970).

2.1.2 Binary Regression

The main motivating problem of this research is to investigate the relationship between ANCA vasculitis flare events and air quality exposure. The response variable for this analysis is binary, denoting whether or not a flare was observed for each hospital visit, determined retroactively by expert clinicians. A binary response variable clearly violates the assumption of Gaussian residuals from equation (2.1), though the idea

of using a linear combination to describe the relationship between covariates and the expected value of the response is still valid, and is known as the *linear probability model* (see for example the first chapter of Aldrich and Nelson (1984)). While this is a simple model to interpret and fit, it can clearly result in predictions that go below 0 or above 1. A more suitable approach is to use generalised linear modelling (GLMs) (McCullagh, 1983), where we assume the expected value of the response distribution determined by a function (called a *link* function) of a linear combination of the covariates. An example used in binary regression is the probit model. This uses the compositional inverse of the Gaussian cumulative distribution function (CDF) as the link function. Another example is logistic regression, which uses log odds (or *logit* function) as the link function.

Inference for GLMs is more complex than for linear regression. For logistic regression, the likelihood is

$$L(\mathbf{y}|\boldsymbol{\beta}) \propto \prod_{i=1}^{N} g^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta})^{y_i} \left(1 - g^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta})\right)^{1-y_i} \tag{2.4}$$

where $g^{-1}(\cdot)$ denotes the inverse of the link function, which in this case is the CDF of the standard Logistic distribution. For frequentists, the maximum likelihood estimate (MLE) can be found by numerical approximation, a common approach being Iteratively Re-Weighted Least Squares (IRLS) (Green, 1984). For Bayesians, inference is trickier as there is no conjugate prior that will allow us to immediately derive the posterior as was the case for linear regression. We can use the normal approximation to derive point estimates of the mode and standard error of the posterior (or any other statistic) (Knuiman and Speed, 1988) but these approximations are unstable for small sample sizes (Griffiths et al., 1987). Instead, Bayesians have found some success by representing the model as a *random utility model* (Horowitz et al., 1994),

$$\begin{aligned} y_i &= \mathrm{I}(z_i \geq 0) \\ z_i &= \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i \\ \epsilon_i &\sim \text{Logistic}(0, 1) \end{aligned} \tag{2.5}$$

The full-conditional distribution of the newly introduced latent variable z_i is Gaussian

truncated to only the positive axis if $y_i = 1$, or to negative axis otherwise. This is easily sampled from which at first makes Gibbs sampling seem like a possible route for inference (Geman and Geman, 1984; Gelfand and Smith, 1990). However the full-conditional of $\boldsymbol{\beta}$ is a Logistic distribution, which is not quite so easy to sample from. Albert and Chib (1993) suggests instead approximating the residual term of the logistic regression model with a t-distribution of 8 or 9 degrees of freedom due to the similarity of their CDFs up to a scaling factor. This is somewhat supported by the fact that the kurtosis of the t-distribution with 9 degrees of freedom is the same as for the Logistic distribution are the same (Mudholkar and George, 1978). Another option is to use the Metropolis-Hastings algorithm (Hastings, 1970) as explored by Frühwirth-Schnatter and Frühwirth (2010). They suggest using $\mathrm{N}(0, \pi^2/3)$ as the proposal distribution as this also resembles the standard logistic distribution. In the same article, they also propose an approximation to the logistic distribution using a finite mixture of normal distributions with variances of the mixture components drawn from a small pool of possible variances with fixed probability. Holmes and Held (2006) suggest expressing the logistic distribution as a scale mixture of Gaussians with transformed Kolmogorov-Smirnov distributed scale parameters. This is not an approximation; this mixture distribution *is* the Logistic distribution (Andrews and Mallows, 1974). While exact inference is appealing, this requires yet another layer of latent variables that are sampled using a rejection method proposed by Devroye (1986). As demonstrated by Frühwirth-Schnatter and Frühwirth (2010), this algorithm is computationally slow. Polson et al. (2013) suggest a latent variable sampled from a new distribution they introduce called the Pólya-Gamma distribution, that can be used to directly create a scale mixture of Gaussians without need of further latent variables or a truncated augmented likelihood. Like Holmes and Held (2006), their sampler requires an accept/reject step, though they demonstrate that rejection rates are extremely low on average. Zens et al. (2020) expanded on this work, introducing yet more latent variables to improve convergence rates of the MCMC by incorporating the *parameter expansion* method described by Liu and Wu (1999).

Although it's clear that logistic regression presents its share of challenges even in random utility form, other forms of binary regression benefit greatly from this representation. For example, the residual term for probit regression is simply a standard

Gaussian, making it simple to infer through Gibbs sampling (Albert and Chib, 1993). In chapter 4 we will go through in detail how to implement binary quantile regression (Kozumi and Kobayashi, 2011), which is also more straightforward to implement.

2.2 Distributed Lag and MIDAS Models

The impact of air quality on human health cannot be measured by single instances in time, but by the accumulated exposure of adverse-health environments over a period of time. Furthermore, flare data and environmental data are sampled in different ways. The environmental data we are using is aggregated daily data, i.e. it is observed on a fixed periodic basis. On the other hand, the flare status of a patient is only observed during clinical visits which can occur at any point in time. Standard regression models, such as those we've described in the preceding section, will only match the patient's current flare status to the pollution values observed in the same unit of time when estimating effect sizes, and are not adequately equipped to handle the differing sampling mechanisms between the response and covariates. In this section we will talk about how distributed lag models (DLMs) (Schwartz, 2000) and MIDAS models (Ghysels et al., 2002) can resolve these issues.

DLMs are regression models that incorporate the temporal lags of the covariates when assessing their relationship with the response. The model assumes that an overall effect size $\boldsymbol{\beta}$ for each covariate is distributed across the lags of the covariate in some way. The nature of this distribution can be estimated in a number of ways. A basic example of a (univariate, continuous response, discrete time) DLM is where we simply parameterise every lag within a window of time as follows,

$$y_t = \beta_0 + \sum_{i=0}^{\tau} \beta_i x_{t-i} + \epsilon_t. \tag{2.6}$$

The time window set by τ defines how many lags we include in the model. It should be chosen large enough to capture all the necessary lags for explaining the variation in y_t. This model is very flexible but can quickly become unwieldy as τ grows larger, especially as we incorporate more covariates. For example, a tau of length 30 time units for 10 covariates would result in a model with 301 parameters. This version of a

DLM, where the effect of each lag is parameterised, is called an *unconstrained* model. An example of a *constrained* DLM would be

$$y_t = \beta_0 + \sum_{i=0}^{\tau} w(i, \boldsymbol{\theta}) x_{t-i} + \epsilon_t. \tag{2.7}$$

where $w(\cdot)$ denotes a weight applied to each lag i. These weights are calculated using a function often referred to as a distributed lag function (DLF), parameterised by $\boldsymbol{\theta}$. $\boldsymbol{\theta}$ usually only contains a small number of elements, making it much more parsimonious than the unconstrained model. An example of a DLF (that we will refer back to often throughout this thesis) is the normalised exponential Almon function, hereafter referred to as the *Nealmon* function;

$$w(k, \boldsymbol{\theta}) = \frac{\exp\left(\sum_{i=1}^{P} \theta_i k^i\right)}{\sum_{s=0}^{\tau} \exp\left(\sum_{i=1}^{P} \theta_i s^i\right)} \tag{2.8}$$

where k is the lag index and P is the user chosen degree of the polynomial inside the exponentail function. Larger values offer a more flexible fit at the cost of more parameters. Note that in order to ensure that the lag effects decline to zero asymptotically, θ_P must be negative, but the remaining elements of $\boldsymbol{\theta}$ can take any value on the real line. Figure 2.1 displays some examples of Nealmon curves with $P = 2$. Ghysels et al. (2016) provide a number of other common choices for the DLF.

Distributed lags have been used in econometrics for decades - see Almon (1965), Hannan (1965), Chen et al. (1972), Sims (1971) and Dhrymes (1981) for some early examples, though many more can be easily found. A related idea that has emerged more recently are mixed data sampling (MIDAS) models (Ghysels et al., 2006). These models were created to resolve the issue of the response variable not being sampled as frequently as the model predictors. They were first presented as essentially constrained DLMs (Ghysels et al., 2002). A basic example is the following:

$$y_t = \beta_0 + \beta_1 \sum_{k=0}^{\tau} w(k, \boldsymbol{\theta}) x_{t-\frac{k}{m}} + \epsilon_t \tag{2.9}$$

where m is the ratio of the covariate sampling frequency over the sampling frequency

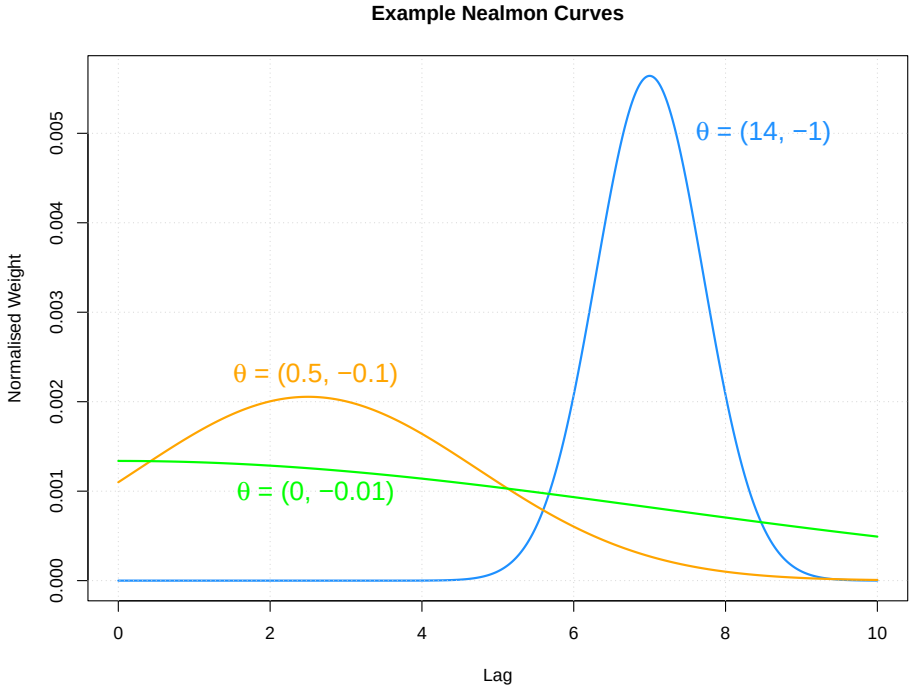


Figure 2.1: An illustration of the two degree Nealmon for three different example parameter settings.

of the response. For example, if the response was sampled yearly and the covariates were sampled monthly then $m = 12$. τ is the time window and $w(\cdot)$ is a DLF, same as in equation (2.7). Our attention will be focused on DLMs and MIDAS models, but we will briefly note here that the issue of mixed sampling frequencies can be resolved in other ways. For example, we can treat the gaps in the lower frequency data as missing data that can be imputed. A crude approach is to simply aggregate over the higher frequency variable (Foroni and Marcellino, 2013). This is in fact equivalent to the basic MIDAS model from equation (2.9) when a simple aggregate is chosen as the DLF. On that matter, Andreou et al. (2010) derived a null-hypothesis test to determine if using the mean as the DLF was consistent with the data, by comparing the asymptotic properties of the MIDAS estimators to standard least squares estimators when a simple aggregate is used. This was later expanded on by Kvedaras and Zemlys (2012) to test for *any* kind of constraint. Other, more sophisticated ideas are to use Kalman filters (Harvey and Pierse, 1984) or regression along related variables (Chow and Lin, 1971, 1976) for example.

MIDAS models and DLMs have been iterated and expanded on in a number of ways

and not just for the purposes of econometrics. In recent years, DLMs have become a popular method for investigating the impact of the environment on different aspects of human health. Schwartz (2000) implemented a DLM into a generalised additive model (Hastie and Tibshirani, 1987) with log link to analyse how daily death count is correlated with air quality. A similar study was performed by Zanobetti et al. (2002). Warren et al. (2020b) used a Bayesian DLM with Gaussian process random effects to account for spatial autocorrelation to investigate the relationship between the birth weight of newborns and air pollution exposure during pregnancy. Wilson et al. (2017a) used a Bayesian DLM to investigate the relationship between air quality and adverse health outcomes of newborns. Clements and Galvão (2008) suggest how to safely incorporate auto-regression terms into MIDAS models, which is not as straightforward as it may first appear since simply adding autoregressive terms implicitly imposes seasonality due to propogating the distributed lag relationship in the autoregressive terms. Foroni et al. (2015) suggests an unconstrained version of the MIDAS, mirroring the unconstrained DLM in equation 2.6. Xu et al. (2019) synthesised the MIDAS model (both constrained and unconstrained versions) within an artificial neural network. This allows for very flexible non-linear fits and performs extremely well in terms of predictive accuracy and goodness-of-fit metrics. Li et al. (2021b) took this a step further by incorporating the effects of temporal correlation. While these perform well in empirical experiments, in both cases the authors admit the methods may be prone to overfit. Antonelli et al. (2021) developed a Bayesian DLM that utilises spike-and-slab priors for variable selection (Mitchell and Beauchamp, 1988). Mogliani and Simoni (2020) similarly suggested a MIDAS model with spike-and-slab priors, but with a Laplacian distribution for the slab to incorporate Bayesian Group-LASSO variable selection (Meier et al., 2008; Xu and Ghosh, 2015). Mork et al. (2021) combines DLMs with Bayesian additive regression trees (Chipman et al., 2010) to analyse the relationship between air quality exposure and the weight of newly born children.

Some of what we described above is readily implementable in `R` through the `dlnm` package (Gasparrini, 2011) for DLMs, and `midasr` (Ghysels et al., 2016) package for MIDAS models. The associated vignettes contain many more informative references therein.

2.3 Response Imbalance

As we will discuss in chapter 4, clinical visits where a flare event was observed only make up about 15% of all clinical visits in our final dataset. This degree of imbalance can bias parameter estimates and predicted probabilities (Czado and Santner, 1992; Tasci et al., 2022). This is a very active area of research in machine learning, for which He and Garcia (2009) provides a thorough overview. See also Lemaître et al. (2017) for practical implementation of some of these methods. Resampling the data is a common approach to deal with imbalance, which falls under two broad categories: undersampling and oversampling. The most basic undersampling approach is to randomly sample from the majority class to match the number of observations in the minority class. This is not ideal as we effectively throw data away potentially increasing the variance of our estimates (Dal Pozzolo et al., 2015). In contrast, the most basic version of oversampling is to resample with replacement the minority class until it matches the number of observations in the majority class. This will result in a lot of repetitions of the same data points which will result in bias. As a result, oversampling can perform worse than undersampling in some situations (Estabrooks et al., 2004). It can also be very computationally costly as we are multiplying the amount of minority data.

There have been some advancements on how to apply over/undersampling. For example, undersampling data points according to their similarity in a K-nearest neighbours cluster (Mani and Zhang, 2003), or creating an ensemble classifier based on multiple combinations of undersampled datasets (Liu et al., 2008). Some researchers have proposed ways to synthesize new data based on the characteristics of the existing data to generate new minority data without so as not to exactly replicate existing datapoints (Chawla et al., 2002; Fernández et al., 2018).

The appeal of this approach is that it is model agnostic since it operates on the data. However alteration of the data itself is not ideal, especially for Bayesians where the data is assumed fixed under Bayes Theorem and as a result the variation due to data resampling will not be taken into account in the posterior. Putting these issues aside, it is unclear how exactly these methods would work in practice for our specific dataset, since the responses are irregularly sampled and linked to multiple lags of the covariates (and not just one row of observations). With this in mind, our goal is to find a way to

handle the imbalance through the model, rather than through the data.

An approach that seems to have worked well in the context of binary regression is using skewed inverse link functions (Stukel, 1988; Czado, 1994). The idea here is that a skewed inverse link is more appropriate when the data has far more of one class than the other because skewness causes these to favour one class over another by design. Chen et al. (1999); Wang and Dey (2010); Caron et al. (2018); Yin et al. (2020) all suggest skewed inverse link functions such as the Weibull distribution CDF (Hallinan Jr, 1993) or the Fréchet distribution CDF (Ramos et al., 2017). However we follow the advice of Kordas (2006) and choose to use quantile regression, which in our case will be implemented using an Asymmetric Laplacian link function (Benoit and den Poel, 2017; Yu and Zhang, 2005). As the name suggests, this is an asymmetric link function, but has the possibility to collapse to a symmetric function (median regression) when appropriate. Unlike the other methods described above, it also does not require us to estimate the skew parameter, as we can simply set it based on the quantiles of the response distribution (overcoming identifiability issues encountered by Chen et al. (1999)). We will discuss this model and how to infer its parameters in more detail in chapter 4.

2.4 Variable Selection and Reversible-Jump MCMC

2.4.1 General Background on Variable Selection

The main purpose of the analysis of vasculitis is to determine if any of the environmental covariates in our data are predictive of flare events. While regression is an important step, our goal is to use variable selection to answer this question directly. When applied to DLMs/MIDAS, model selection usually revolves around likelihood based penalty metrics (Urban et al., 2021; Gasparrini et al., 2010; Warren et al., 2020a; Ghysels et al., 2016). Antonelli et al. (2021) and Mogliani and Simoni (2020) both utilise spike-and-slab priors to induce a point mass at zero in the posterior of the coefficients. There are other options however, which we will examine in this section.

A well-known but crude method of variable selection is stepwise selection. This is where we sequentially re-fit the model with a new variable included/excluded and choose to

retain the resulting model usually on the basis of statistical significance. For example, we might start with the intercept only model and iteratively add more variables until no more gives us statistical significance; this is known as forward-selection. Working backwards, starting with the saturated model and iteratively removing variables that fail statistical significance tests until no more do, is known as backward elimination. Regardless, these methods are strongly criticised (Harrell et al., 2001). For one, multiple testing on this scale means a highly inflated Type-I error rate (Flom and Cassell, 2007). It is also unstable in the sense that, due to its discreteness (a variable is either included or not), a small change in the data can potentially change the selected set in a drastic way (Breiman, 1995, 1996).

An approach that has gained traction recently in machine learning is to measure variable importance in non-linear models such as XGBoost (Chen and Guestrin, 2016) using Shapley values (Lundberg and Lee, 2017), however Shapley values have also been criticised (Kumar et al., 2020).

A popular method for variable selection is the Least Absolute shrinkage and Selection Operator (LASSO) method (Tibshirani, 1996). This works by bounding the L1 norm of the $\boldsymbol{\beta}$ estimators above by a constant so that they cannot grow too large. This forces the coefficients to shrink to zero (Hastie et al., 2009). The LASSO has a number of extensions and generalisations, such as group-LASSO (Yuan and Lin, 2006; Meier et al., 2008), which allows for grouping of the variables so that every covariate in a group must be included or excluded. Another extension is the Elastic Net (Zou and Hastie, 2005), which combines LASSO with ridge regression (Hoerl and Kennard, 1970) to get a blend of shrinkage effects; variable sparsity from the LASSO, and collinearity mitigation from the ridge regression. There is also the adaptive LASSO (Zou, 2006), which weights the penalty term using the ordinary least squares coefficients leading to the so-called *oracle* property (Fan and Li, 2001). Needless to say, there are a lot of options from the LASSO family of variable selection methods. See Hastie et al. (2015) for a broader overview of these methods.

Regularisation methods have a direct Bayesian analogue via the prior distribution. When the Laplacian distribution (centered at zero) is used as the prior for the regression parameters $\boldsymbol{\beta}$, the mode of the posterior is equal to the frequentist LASSO estimator.

For this reason, this approach is often called the *Bayesian* LASSO (Park and Casella, 2008). In this case, the scale parameter of the Laplacian controls the strength of the constraint, and the posterior is readily tractable via Gibbs sampling since the Laplacian distribution can be represented as a scale-mixture of Gaussians. However, there are some drawbacks to the Bayesian LASSO. Castillo et al. (2015) showed that the scale of a Laplacian prior cannot be set such that it both encourages sparsity without over-shrinking the non-zero coefficients also. We could use the LASSO (or something similar) as a first-pass covariate filter based on the MAP, and re-run the model in a more 'standard' way on the retained covariate set, which has some credibility in frequentist analyses (Zhao et al., 2021; Leeb et al., 2015). But basing a selection rule off only one point of the posterior, the mode, and ignoring the rest of it arguably undermines the purpose of a posterior in the first place. Also, the uncertainty of the model choice is not taken into account when using this approach.

The prior can be utilised in other ways to encourage sparse results. Xu and Ghosh (2015) recommends the *spike-and-slab* prior; a mixture prior with one component set as an atom placed at zero (the 'spike') and the other usually a Gaussian or Laplacian prior (the 'slab'). This was utilised in the context of distributed lag models by Antonelli et al. (2021) and Mogliani and Simoni (2020). Another sparse prior is the horseshoe prior (Carvalho et al., 2010; Piironen and Vehtari, 2017), which places hierarchical half-Cauchy priors on the scale of the regression coefficients. The resulting distribution contains a sharp pole at zero, enforcing sparsity, but also has fat tails so that the non-zero coefficients are not too strongly constricted.

2.4.2 Reversible-Jump MCMC

Green (1995) explains how to augment a Metropolis-Hastings algorithm to perform MCMC across different candidate models of (potentially) differing parameter spaces, incorporated alongside the parameter updates themselves. This is known as Reversible-Jump MCMC (RJ-MCMC). RJ-MCMC can be used to decide between whole class of models, for example Hastie and Green (2012) go through an example of choosing between the Poisson distribution or Negative Binomial distribution for modelling count data using RJ-MCMC. However we will only be focusing on its application to variable selection.

To perform RJ-MCMC for variable selection we introduce a new parameter, a binary inclusion indicator $\gamma_i \in \{0, 1\}$ for all i corresponding to a covariate (except the intercept, which we assume is always included). Each γ_i represents the inclusion (when equal to 1) or exclusion (when equal to 0) of the i^{th} candidate covariate. As with any other parameter in a Bayesian model, these are assigned priors; the most obvious choice is a Bernoulli distribution, where the probability of success parameter should be set to reflect our prior belief that covariate i should be included. This parameter is included in Bayes theorem,

$$\pi(\boldsymbol{\gamma}, \boldsymbol{\beta}_{\boldsymbol{\gamma}}|\mathbf{y}) \propto \pi(\mathbf{y}|\boldsymbol{\gamma}, \boldsymbol{\beta}_{\boldsymbol{\gamma}})\pi(\boldsymbol{\gamma}, \boldsymbol{\beta}_{\boldsymbol{\gamma}}) \tag{2.10}$$

where $\boldsymbol{\beta}_{\boldsymbol{\gamma}}$ is $\boldsymbol{\beta}$ with only the components supported by $\boldsymbol{\gamma}$ retained. $\boldsymbol{\gamma}$ is updated using Metropolis-Hastings and as usual this means specifying a proposal distribution that generates potential updates and derive an acceptance probability such that the transition probabilities satisfy detailed balance. Detailed balance is the condition that

$$\pi_i P_{ij} = \pi_j P_{ji} \tag{2.11}$$

where π_i is the stationary probability of being in state i and P_{ij} denotes the i, jth elements of the transition matrix (note this is just for discrete Markov chains and that an analogous continuous version exists). It is required as it is how the correct acceptance probability for Metropolis-Hastings is computed (Gilks et al., 1995).

Of course a change in $\boldsymbol{\gamma}$ may necessitate the generation ('birth') and/or the removal ('death') of parameters. The acceptance probability must take into account this shift of dimensionality when births/deaths are proposed.

Let us illustrate with an example. Let new values of $\boldsymbol{\gamma}$, denoted $\boldsymbol{\gamma}^\star$, be proposed by selecting a single component at random (uniformly) and swap its value from zero to one or vice-versa. Let the proposed birth slope parameter $\beta_+^\star$ be generated from $\text{N}(0, \sigma^2)$, with σ^2 fixed and chosen by the user. We end up with two possibilities each iteration; a birth or death move. The acceptance probability of a birth move is

$$\alpha_+ = \frac{\pi(\mathbf{y}|\boldsymbol{\gamma}^\star, \boldsymbol{\beta}^\star)\pi(\boldsymbol{\gamma}^\star)\pi(\beta_+^\star)}{\pi(\mathbf{y}|\boldsymbol{\gamma}, \boldsymbol{\beta})\pi(\boldsymbol{\gamma})} f(\beta_+^\star|0, \sigma^2)^{-1}$$

where $f(\cdot|\mu, \sigma^2)$ is the pdf of the Gaussian distribution. Notice that the acceptance probability must also include the prior for the birth slope. Likewise, the acceptance probability for a death move is

$$\alpha_- = \frac{\pi(\mathbf{y}|\boldsymbol{\gamma}^\star, \boldsymbol{\beta}^\star)\pi(\boldsymbol{\gamma}^\star)}{\pi(\mathbf{y}|\boldsymbol{\gamma}, \boldsymbol{\beta})\pi(\boldsymbol{\gamma})\pi(\beta_-)} f(\beta_-|0, \sigma^2)$$

As noted in Hastie and Green (2012), the above framework is an over-complication for standard regression; since the death of a $\boldsymbol{\beta}$ component is equivalent to simply setting it to zero then it does not need to be framed as a trans-dimensional problem to begin with. A proposal distribution that randomly places/removes a component of $\boldsymbol{\beta}$ to/from an atom at zero achieves the same thing. But it is worth going through here as we will use this same logic in chapter 4 where we discuss a genuine trans-dimensional regression problem.

The appealing thing about this method is that it allows us to directly infer our posterior belief that a variable should be included, and it does so without changing how we regularise the non-zero elements of $\boldsymbol{\beta}$ itself, as is the case with Bayesian LASSO, spike-and-slab, and horseshoe priors. While it requires tuning a prior distribution for the variable indicators, it is a very easily interpreted prior; it is simply the prior degree of belief, between 0 and 1, that the corresponding variable is a true predictor, which even a non-statistical expert can understand (as opposed to relatively esoteric parameters like Laplacian distribution scale parameters). This makes it far easier to incorporate expert prior knowledge when available. On top of all this, as demonstrated by Holmes and Held (2006), RJ-MCMC-like inference can be implemented very elegantly into latent variable models. We will demonstrate the same for our model in section 4.2.

2.5 Compartmental Models

In chapter 5 we will discuss an approach of predicting COVID–19 cases using a compartmental model, leveraging data on social contacts between age groups (Prem et al.,

2017; Mossong et al., 2008). In this section we provide some basic background.

Compartmental models track how a population of agents progress over time through different interlinking stages (the *compartments*) of some larger state. These are very commonly used in epidemiology to model disease progression in a population, including coronavirus during the pandemic. Examples include Gleeson et al. (2022); Lemos-Paiao et al. (2020); He et al. (2020); Leontitsis et al. (2021); Dashtbali and Mirzaie (2021); Rădulescu et al. (2020), and many, many more. A basic example of a compartmental model is the *SIR* model which uses three compartments:

- $S(t)$, the number of susceptible individuals in the population at time t,
- $I(t)$, the number of infected in the population and
- $R(t)$, the number of individuals removed from in infection. This includes those who have either died or safely recovered from the disease. We assume here that once someone is removed they can no longer be infected again.

The total population $N = S(t) + I(t) + R(t)$ is assumed fixed and equal for all t.

We need to form parameterised assumptions about how the population flows from one compartment into another. For example, it stands to reason that in order for a susceptible person to catch the disease, they must come into contact with an infected person. The total number of possible susceptible/infected interactions, assuming every person in one compartment can interact with every person in another, is $S(t) \times I(t)$. If we assume that each interaction is equally likely to occur per unit time (the assumption of uniform mixing), then on average the number of new infections at time t is

$$\frac{S(t)I(t)}{N}\beta \tag{2.12}$$

where β is the average rate at which a susceptible/infected interaction leads to a new infection per unit of time. Let us further assume that people recover from the infection at an average rate of γ per time unit. These assumptions can be expressed as a system of Ordinary Differential Equations (ODEs):

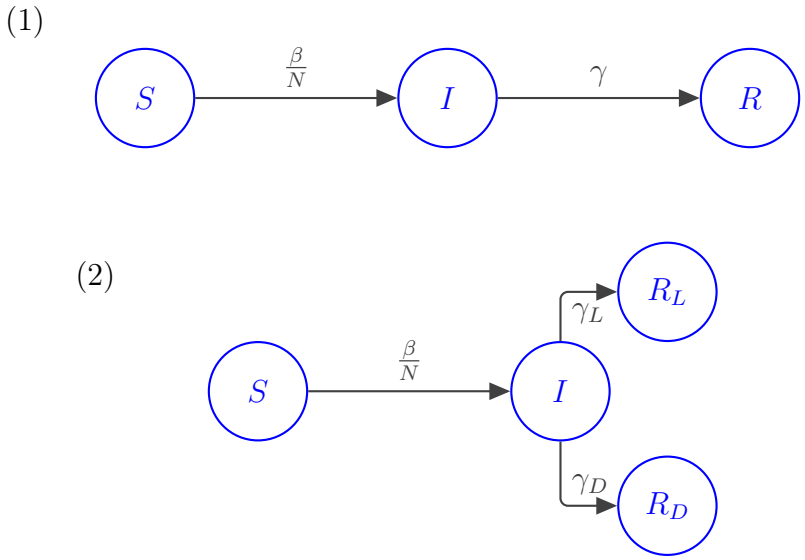


Figure 2.2: SIR model diagrams; (1) displays the straightforward case of one Removed compartment, and (2) illustrates the case where we split the Removed compartment into two, one representing the number of living and the other representing the number of dead post infection.

$$\frac{\mathrm{d}S(t)}{\mathrm{d}t} = -\frac{S(t)I(t)}{N}\beta \tag{2.13}$$

$$\frac{\mathrm{d}I(t)}{\mathrm{d}t} = \frac{S(t)I(t)}{N}\beta - \gamma I(t) \tag{2.14}$$

$$\frac{\mathrm{d}R(t)}{\mathrm{d}t} = \gamma I(t) \tag{2.15}$$

We can numerically solve the ODEs themselves if β and γ are known. To do this, the first step is to link one or more of the compartments to observable data; if this is not possible we can create a new compartment. For example, we can split the 'Removed' compartment into two: 'Removed, living' $R_L(t)$ and 'Removed, dead' $R_D(t)$. γ would then be split respectively into γ_L, γ_D. Figure 2.2 illustrates this. With a setup like this, the $R_D(t)$ compartment can be linked to observed deaths - let us call $d(t)$. The least squares estimate of β (assuming γ fixed) is then

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{T} \left(d(t_i) - R_D(t_i;\beta)\right)^2 \tag{2.16}$$

where the summation from 1 to T is across the units of time that the deaths are reported (daily, weekly, etc.). This requires solving a system of ODEs, which needs its

own layer of approximation.

The SIR model can be extended in any number of ways to make it more realistic. For example, in chapter 5 we will fit an age-structured Susceptible Exposed Infected Removed (SEIR) model that incorporates heterogeneous social mixing across age, relaxing the assumption that all age groups interact with each other in the same way. This will be used to quantify the effect of lockdowns in deterring the spread of COVID-19 in Dublin, Ireland.

There has already been a lot of exploration of age-structured SEIR models for disease incidence. A SEIR model with age-structuring is used by Teimouri (2020) for the London area incorporating contact tracing; their interest was detailing the impact of social mixing and contact tracing on the effective reproduction rate of the disease as opposed to model calibration. In a similar vein, Grimm et al. (2021) use assumed epidemiological parameters to simulate the impact of age-specific control measures and contact tracing impact with a focus on the impact of control measures on factors including hospitalisations and deaths. The impact of four control measures (school closure, social distancing, quarantine, and isolation) are simulated by Lee et al. (2021) to explore reproduction rates in South Korea using an age-structured SEIR model of disease spread. Maximum likelihood estimation is used to estimate contact scaling parameters, however no uncertainty in estimates or projections is presented. A two-cohort age-segmented model (age in years $\leq$65/>65) is proposed by Cuevas-Maraver et al. (2021) for Mexican incidence counts, suggesting that age specific control measures may have utility for public health policy decisions. The impact of three specific governmental interventions on case incidence is discussed by Kimathi et al. (2021) employing an age-structured SEIR model with predetermined model parameters. A SIRD model (SIR model with a death compartment) is fitted to data from Brazil by Canabarro et al. (2020) examining how different interventions affect different age groups. They expand the model by including a hospitalisation compartment to project when demand on Intensive Care Units could exceed supply under different interventions. A SEIRD model (SEIR model with a death compartment) is used by Moore et al. (2021) to project the number of deaths over the course of the vaccine rollout in the UK considering different lockdown scenarios. A Bayesian hierarchical model is used to estimate the effect of government interventions in Brauner et al. (2021), similar to previous work by

Flaxman et al. (2020) but harnessing data from multiple countries to disentangle the effects of different NPIs.

3 MIDAS Generalised Regression and Frequentist Inference

3.1 Irregularly Sampled Response Variables

An immediate issue we need to address is that the MIDAS model as expressed in section 2.2, equation (2.9), assumes the response vector y_t is sampled regularly (i.e., sampled with some exact periodic frequency). This isn't suitable for our purposes as our response data is sampled irregularly (i.e., sampled at any point in time, not periodically) but we can adapt the model as follows: let S_t be the time indices of the covariates x_t that fall within the time window of length τ going back from time t. Define $\Delta t_s = t - s$. We can alter the MIDAS as follows:

$$\begin{aligned} y_t &= \beta_0 + \beta_1 \sum_{s \in S_t} w(\Delta t_s; \boldsymbol{\theta}) x_s + \epsilon_t, \\ \epsilon_t &\sim \mathrm{N}(0, \sigma^2) \end{aligned} \tag{3.1}$$

so that the DLF acts on the differences between the lags rather than the lag indices. We call this the *Irregular Time Series MIDAS*, or IRTS-MIDAS. This collapses to the standard MIDAS in the case where the response is sampled regularly, i.e., when all Δt_s are constant. If that constant equal to 1, then we get exact equivalence with the standard MIDAS model; for anything other than 1, the interpretation of the model itself will not change as the same curve will still maximise the likelihood, but the values of $\boldsymbol{\theta}$ will be scaled to compensate the difference of scale in the time indices. For example, if the (constant) difference between each lag is d and we are using the Nealmon DLF, then θ_1/d and θ_2/d^2 will return the same curve for all values of d. Therefore if

exact agreement with the standard MIDAS is desired in the case of regularly sampled response data, this can be achieved by simply scaling the time axis so that the response has unit difference.

3.2 Extension to Multiple Covariates

The MIDAS can be extended to include arbitrarily many covariates;

$$y_t = \beta_0 + \beta_1 \sum_{s \in S_t^{(1)}} w_1(\Delta t_s; \boldsymbol{\theta}_1) x_s^{(1)} + \cdots + \beta_p \sum_{s \in S_t^{(p)}} w_p(\Delta t_s; \boldsymbol{\theta}_p) x_s^{(p)} + \epsilon_t, \qquad \epsilon_t \sim \mathrm{N}(0, \sigma^2). \tag{3.2}$$

The bracketed superscripts used throughout the above equation are covariate indices, for example $S_t^{(k)}$ is the time window for the k^{th} covariate of length τ_k. Note each covariate k has their own DLF $w_k(\cdot)$, associated DLF parameters $\boldsymbol{\theta}_k$, and time window τ_k.

Matrix Formulation of the MIDAS Model

The notation in (3.2) is unwieldy so it is worth discussing how to convert this to matrix notation. First, define the *weight matrix* $\mathbf{W}_k(\boldsymbol{\theta})$, where $\boldsymbol{\theta} = \cup_{i=1}^{p} \boldsymbol{\theta}_i$, where

$$\mathbf{W}_k(\boldsymbol{\theta}_k)_{ij} = \delta[s_j \in S_{t_i}^{(k)}] w_k(t_i - s_j; \boldsymbol{\theta}_k) \tag{3.3}$$

where $\delta[z]$ is the indicator function for condition z. s_j is the time index associated with the j^{th} value of $\mathbf{x}_k$, and t_i is the index of the i^{th} value of $\mathbf{y}$. In other words, the columns of $\mathbf{W}_k(\boldsymbol{\theta}_k)$ correspond to every observation of $\mathbf{x}_k$, and the rows correspond to every time window formed by $\mathbf{y}$. Each cell is the weight applied to each observation within the corresponding time window. The number of rows are equal to the number of $\mathbf{y}$ observations, and the number of columns are the number of $\mathbf{x}_k$ observations.

A weight matrix is constructed for every covariate. We can then create an overall weight matrix by simply concatenating them side-by-side so that the full weight matrix has the same number of rows as its component matrices, but the number of columns is the

number of every observation across every covariate:

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_1 \dots \mathbf{W}_p \end{bmatrix}$$

(we are hereafter dropping the explicit notation denoting $\boldsymbol{\theta}_k$ dependence). Note that for most practical choices of $\boldsymbol{\tau}$ (see section 3.2.3), this will be very sparse.

Now, we also want to construct compact vector notation for the covariates themselves. Let $\mathbf{x}_k$ be the column vector of all observed values of the k^{th} covariate. We construct a block matrix $\mathbf{X}$ as follows:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & & & \\ & \mathbf{x}_2 & & \\ & & \ddots & \\ & & & \mathbf{x}_p \end{bmatrix}$$

To be clear, each block is a vector of covariates and the blocks are arranged in a diagonal formation. The off-block diagonals are zero vectors so this will also be a sparse matrix.

With the above notation, we can now express (3.2) more succinctly:

$$\begin{aligned} \mathbf{y} &= \mathbf{W}\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \\ \boldsymbol{\epsilon} &\sim \mathrm{N}(\mathbf{0}, \sigma^2\mathbf{I}) \end{aligned} \tag{3.4}$$

To include an intercept we simply append a $\mathbf{1}$ vector to $\mathbf{WX}$. The sparsity of $\mathbf{W}$ and $\mathbf{X}$ makes the computation of their product efficient and scalable.

3.2.1 Frequentist Parameter Inference for MIDAS

Constrained DLMs and MIDAS appear to be the best of both worlds; they offer data-driven flexibility of model fit without an enormous number of parameters. But inference is harder; the parameters of unconstrained models such as equation (2.6) are linear functions of y_t and so can be easily inferred as standard linear models. The same applies for equation (3.4) if $\boldsymbol{\theta}$ is known, in which case we can estimate $\boldsymbol{\beta}$ via, for

example, least squares,

$$\hat{\boldsymbol{\beta}}_{\text{LS}} = (\tilde{\mathbf{X}}^\top \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}} \mathbf{y} \tag{3.5}$$

where $\tilde{\mathbf{X}} = \mathbf{W}\mathbf{X}$. Note that holding $\boldsymbol{\theta}$ constant also imples $\mathbf{W}$ is constant, so $\tilde{\mathbf{X}}$ can be treated as a standard design matrix that we would use in multiple linear regression. However in practice $\boldsymbol{\theta}$ will not be known. The maximum likelihood estimate of $\boldsymbol{\theta}$ will depend on the user's choice of corresponding DLF, but for most cases a closed form expression will not be possible to derive so numerical optimisation algorithms such as, say, the BFGS method or its extensions (Byrd et al., 1995), must be applied. Both $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ can be included in the optimiser, or alternatively we can update $\boldsymbol{\beta}$ using least squares conditioning on the current value in the optimiser, and vice-versa, until some termination criteria is met. Numerical optimisation of log-likelihoods are straightforward in `R` by leveraging the `optimx` function (Nash et al., 2011; Nash, 2014a).

When fitting the model, it may be prudent (or even necessary) to regularise the likelihood due to a potentially high degree of correlation between $\boldsymbol{\theta}$ parameters. There are cases where constraints should be imposed as they wouldn't make sense without them, for example θ_P in the Nealmon DLF, equation (2.8), must be negative to ensure that the weights decline to zero asymptotically.

3.2.2 Computational Considerations

The dimensions of $\mathbf{X}$ and $\mathbf{W}$ will expand rapidly the larger the dataset, and so some computational issues may arise. The biggest bottleneck to performance will be computing $\mathbf{W}$, as it will need to be re-constructed for every iteration of $\boldsymbol{\theta}$ inference; this means re-evaluation of the DLF across all the covariates and time windows for every numerical optimisation iteration. Parallel computing can alleviate this as the evaluation of every DLF is independent. Since the DLF computations are usually straightforward this is a task that might be best spread across a GPU. It's important to identify any component of the DLF that doesn't require re-evaluation every iteration, so that we can pre-process as much as possible to remove as much redundancy from the algorithms

as possible. Computing and storing all the required lag differences Δt_s is an example. As stated before, both $\mathbf{X}$ and $\mathbf{W}$ will be sparse so triplet representation (Buluç et al., 2009) will be effective for any matrix operations.

Another issue to consider is numerical instability which may occur when attempting to normalise the DLF. For example, the numerator of the Nealmon is exponential, which can easily explode in value. It's safer to evaluate the log instead; in this case the denominator becomes

$$\begin{aligned}\log\left[\sum_{s\in\mathcal{S}_t}\exp(z_s)\right] &= \log\left[\exp(z^\star)\sum_{s\in\mathcal{S}_t}\exp(z_s - z^\star)\right] \\ &= z^\star + \log\left[\sum_{s\in\mathcal{S}_t}\exp(z_s - z^\star)\right]\end{aligned}$$

where z_s is equal to $\sum_i^P \theta_i k_s^i$, and $z^\star$ is chosen to be the maximum value of all the z_s. The purpose of this is that none of the components of the denominator's sum ever exceeds $\exp(0) = 1$ and therefore numerical overflow is no longer a concern. This is known as the *LogSumExp* trick.

3.2.3 The Time Window

Up to this point we have not given much focus on how to select the time window parameter τ. In some cases this can be determined from context; for example, in their study of how a newborn's health is affected by their mother's air quality exposure during gestation, Mork et al. (2021) used a τ value of 37, roughly the length of time of gestation. But it will not always be obvious from context. τ needs to be at least large enough that the time window captures all the lags that meaningfully explain the response variance, though how large that is exactly is not clear. But while the minimum value of τ might not be known, its maximum is clearly the smallest lag length of the covariates that is available for all the response data. Larger values than this will result in some time windows containing less lags than others, which in turn will mean the weighting scheme from the DLF will be applied inconsistently. Ghysels et al. (2006) argues that simply setting τ to its largest value should be a safe choice since the lag

effects will naturally decline to zero in the long term. In a sense, this allows the model to decide its own time window length in an indirect manner. A potential drawback of this proposal is that larger values of τ will result in denser weight matrices $\mathbf{W}$, exacerbating the computational complexity.

We will investigate the impact of setting τ through simulations in section 3.3, and we will find that setting it too small will lead to badly biased results, though setting overly large has minimal negative impact.

3.2.4 Generalised MIDAS

Adapting the MIDAS framework to handle generalised response data proceeds the same as for GLMs discussed in section 2.1.2. Let us assume that the mean of y_t follows the general model

$$\mathbb{E}[y_t] = g^{-1}(\eta_t) \tag{3.6}$$

where η_t is taken from the components of

$$\boldsymbol{\eta} = \mathbf{W}\mathbf{X}\boldsymbol{\beta} \tag{3.7}$$

where $\mathbf{W}$, $\mathbf{X}$ and $\boldsymbol{\beta}$ are defined as in (3.4). Setting $g(\cdot)$ as the log odds corresponds to a logistic regression. As with standard GLMs, $\boldsymbol{\beta}$ can be inferred via Iteratively Re-weighted Least Squares (IRLS) (Green, 1984). Let $\tilde{\mathbf{X}} = \mathbf{W}\mathbf{X}$ as before and $\tilde{\mathbf{x}}_i$ denote the i^{th} row of $\tilde{\mathbf{X}}$. Define $\mathbf{p} = [p_1, p_2, \dots p_n]$, where $p_i = (1 + e^{-\hat{\boldsymbol{\beta}}^\top \tilde{\mathbf{x}}_i})^{-1}$ and n is the sample size of $\mathbf{y}$. Finally, let $\mathbf{M} = \text{diag}(p_i(1-p_i))$. Through IRLS, the k^{th} iteration of $\hat{\boldsymbol{\beta}}$ is

$$\hat{\boldsymbol{\beta}}_{k+1} = (\tilde{\mathbf{X}}^\top \mathbf{M}_k \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}}^\top (\mathbf{M}_k \tilde{\mathbf{X}} \hat{\boldsymbol{\beta}}_k + \mathbf{y} - \mathbf{p}_k) \tag{3.8}$$

which is the Newton-Raphson estimated maximum of the log-likelihood. To jointly infer with $\boldsymbol{\theta}$ we can apply something like the BFGS as we've suggested before in Section 3.2.1, but now each iteration also performs the IRLS method to find $\boldsymbol{\beta}$.

In our `R` implementation, which we discuss in more detail in appendix C, we adapted code from the `midasr` (Ghysels et al., 2016) package to build a function that infers the parameters jointly using the `optim` function by default, though can be easily set to use a number of other functions like `optimx` (Nash et al., 2011) if desired. We use the log-likelihood times minus two as the objective function, which for the binary regression model is simply

$$-2L = -2\sum_t y_t \ln(p_t) + (1 - y_t)\ln(1 - p_t). \tag{3.9}$$

This same idea can be applied to other forms of generalised linear regression. For example, if we instead had count data we could use the log link to perform Poisson regression (Schwartz, 2000). The corresponding objective function then would be

$$-2L = -2\sum_t y_t \boldsymbol{\beta}^\top \tilde{\mathbf{x}}_t - \exp\left(\boldsymbol{\beta}^\top \tilde{\mathbf{x}}_t\right) \tag{3.10}$$

but everything else would proceed as described above.

3.2.5 Appropriateness of Application to Irregularly Sampled Covariates

The time differential Δt that is used as the argument for the DLF may at first appear to imply that this method allows for irregularly sampled *covariates* as well as responses. This would open up new opportunities to explore more datasets. With the modern prevalence of mobile technology (Rehg et al., 2017), patients are able to submit self reports through health monitoring apps, such as surveys (Mulhern et al., 2015), self-administered tests such as home spirometry (Moor et al., 2018), among other things (Sama et al., 2014) which patients will carry out on an irregular basis. It would also be an elegant alternative against explicit imputation when using regular data with missing values.

However we do not recommend this; if the covariates are irregular, this opens up the very likely possibility of having a different number of observations within each time window, resulting in a inconsistently normalised DLF, and the DLF must be normalised

in order for $\boldsymbol{\beta}$ to be identifiable.

For example, imagine a time window with 3 points which are 0.1, 0.2 and 0.5 units away from the edge of the time window respectively. Let's say we are using the Nealmon of degree 2 as our DLF, with $\theta_1 = 0.5$ and $\theta_2 = -0.1$. Then the weights for each of the points above will be 0.31, 0.32 and 0.37 respectively. Now imagine another time window with an identical spread of points, except a fourth point 1 unit away from the edge of the time window. The weights assigned in this case will be 0.21, 0.22, 0.26 and 0.30 respectively. So although the first 3 points in the latter time window are in the same relative position as the 3 points in the former time window, they are given a different weight due to the inclusion of the fourth point. Hence the effect will be distributed in an inconsistent manner, and by extension the fitted DLF will have an inconsistent interpretation. Further, consider the more extreme example where only one value falls within the time window. No matter this point's relative position from the time window - whether it's close to one of the boundaries or somewhere in between - it will be assigned a weight of 1 due to normalisation, rendering the DLF meaningless as a whole. Changing the DLF parameters will have no impact on that point's contribution to the likelihood as any set of parameters will still return a weight of 1 for that observation.

To ensure the DLF is applied consistently (and thus has a consistent interpretation) you may only use a time window length such that every time window contains the same number of values. The fitted DLF curve will then be the same across all time windows. But when data is irregular this may not be possible, especially when sampling rates are heterogeneous.

The Gradient

Quasi-Newton numerical optimisation algorithms such as the BFGS method (Broyden, 1970) require evaluation of the gradient of the objective function. When one is not provided, usually numerical derivatives are used but these introduce an extra layer of estimation and computation, so if the true gradient is known it should be supplied. As already discussed in section 3.2.4, the log-likelihood is

$$L(\boldsymbol{\theta}, \boldsymbol{\beta}) = \sum_t y_t \ln(p_t) + (1 - y_t) \ln(1 - p_t)$$

where $p_t = (1 + \exp(-\eta_t))^{-1}$ as before and

$$\eta_t = \beta_0 + \sum_{i=1}^{p} \beta_i \sum_{s \in S_t^{(i)}} w_i(\Delta t_s, \boldsymbol{\theta}_i) x_s^{(i)} \tag{3.11}$$

Let us first focus on $\boldsymbol{\beta}$. By the chain rule, the m^{th} $\boldsymbol{\beta}$ component of the gradient is

$$\frac{\partial L}{\partial \beta_m} = \sum_t \frac{\partial L}{\partial p_t} \frac{\partial p_t}{\partial \eta_t} \frac{\partial \eta_t}{\partial \beta_m} \tag{3.12}$$

where

$$\begin{aligned}
\frac{\partial L}{\partial p_t} &= \frac{y_t}{p_t} - \frac{1 - y_t}{1 - p_t}, \\
\frac{\partial p_t}{\partial \eta_t} &= (1 + \exp(-\eta_t))^{-2} \exp(-\eta_t), \\
\frac{\partial \eta_t}{\partial \beta_m} &= \begin{cases} \sum_{s \in S_t} w_m(\Delta t_s, \boldsymbol{\theta}_m) x_s^{(m)} & \text{if } m \neq 0, \\ 1 & \text{if } m = 0. \end{cases}
\end{aligned}$$

The gradient components with respect to $\boldsymbol{\theta}$ are similar:

$$\frac{\partial L}{\partial \theta_k} = \sum_t \frac{\partial L}{\partial p_t} \frac{\partial p_t}{\partial \eta_t} \frac{\partial \eta_t}{\partial \theta_k} \tag{3.13}$$

The first two factors in (3.13) are the same as the first two in (3.12). The final factor is

$$\frac{\partial \eta_t}{\partial \theta_k} = \beta_m \sum_{s \in S_t} x_s^{(m)} \frac{\partial}{\partial \theta_k} w_m(\Delta t_s, \boldsymbol{\theta}_m) \tag{3.14}$$

where $\theta_k \in \boldsymbol{\theta}_m$ and $\boldsymbol{\theta}_m$ denotes the set of $\boldsymbol{\theta}$ that are used in the DLF corresponding to covariate m. In some cases it may be prudent to assign the same DLF and associated

parameters to more than one covariate if we suspect that lag effects are distributed in the same way. In this case we can simply sum over (3.14) for all the relevant covariates to compute $\partial\eta_t/\partial\theta_k$.

The derivative of $w_m(\Delta t_s, \boldsymbol{\theta}_m)$ will depend on the user's choice of DLF. It will be the case for some choices that it is intractable. Even if numerical derivatives are required at this point, it is still much more preferable over numerically deriving the entire gradient. In the case where the Nealmon is used, the derivative is tractable and works out to be

$$\frac{\partial w_m(\Delta t_s, \boldsymbol{\theta}_j)}{\partial \theta_k} = \frac{\exp\left(\sum_{i=1}^{P} \theta_i \Delta t_s^i\right)\left[\Delta t_s^M \sum_{n\in S_t} \exp\left(\sum_{i=1}^{P} \theta_i \Delta t_n^i\right) - \sum_{n\in S_t} \Delta t_n^M \exp\left(\sum_{i=1}^{P} \theta_i \Delta t_n^i\right)\right]}{\left[\sum_{n\in S_t} \exp\left(\sum_{i=1}^{P} \theta_i \Delta t_n^i\right)\right]^2} \tag{3.15}$$

where $M \in \{1, \dots P\}$ is the polynomial order that θ_k is a coefficient of in equation (2.8).

3.3 Simulation Study

We test the method described in this chapter with a simulation study. We will generate data under a number of different parameter settings, which we will discuss below, and sample sizes n (where $n = 100, 500, 1000$). All experiments will use 2 randomly generated time series covariates to create the response, with two very different simulation DLF profiles to observe their differences. 1,000 such datasets are generated for each parameter setting and sample size configuration, which are fed into the IRTS-MIDAS inference algorithm. Our objective is to examine how biased the model estimates are from the true generating parameter values of these datasets.

3.3.1 Data Generation and Simulation Details

Since the response data is irregular we must first generate its temporal indices. To do this we randomly sample n values from the Exponential distribution. By default the rate parameter will be set to 1/15, but we will vary this value for some of the simulations (see below). A weight matrix is constructed using the two-degree Nealmon

DLF with DLF parameters $\{14, -1\}$ for one covariate and $\{0.5, -0.1\}$ for the other, using a time window of length 15 (by default - this is also subject to change in some of the experiments). These DLFs are illustrated in figure 2.1: one is highly concentrated on a point relatively far from the edge of the time window, the other is more dispersed but centered closer to the edge, giving us two very different kinds of distributed lag relationships to test. The regression coefficients (including intercept) are $\{0, 1.5, -1.5\}$. We use the logit link function to generate the binary responses.

Covariates are independent autoregression time series of order 1, sampled randomly from the following distribution;

$$\begin{aligned} x_t &= 0.5x_{t-1} + 2\cos\left(\frac{2\pi}{\omega}t + \phi\right) + \epsilon_t \\ \epsilon_t &\sim \mathrm{N}(0, 1) \end{aligned} \tag{3.16}$$

where ω, the frequency parameter, controls the seasonality of the covariate term and ϕ, the shift parameter, shifts the cosine wave along the x-axis. The seasonality ensures the covariate varies sufficiently over time so that the variation of the effect won't be aggregated out by the DLF. Two covariates are created for each of the simulations; both with a periodicity of 30, but one with a shift of 2.5 and the other a shift of 7. This makes it so they both exhibit the same seasonality but their peaks are placed on different lags, inducing a small amount of collinearity to make the setting more realistic. Besides this, we will create a number of different settings explained below:

1. **Effect of response sampling rate:** We will run an experiment with different response rates from the exponential distribution mean parameter: 1/7 and 1/30.

2. **Effect of time window:** The time window we will use to create the data will be 15 units wide. We want to examine the effect of misspecifying the time window in the model, so we will use a shorter window of 7 units wide and a larger window of 25 units wide.

3. **Effect of covariate noise:** For one set of experiments, we will use a standard deviation of 3 instead of 1 in equation (3.16).

We test each simulation for three different number of response samples: $n = 100, 500$

and 1000. The models are fit to the simulated data using the BFGS algorithm. This entire procedure is repeated 1,000 times over for each experiment.

The simulation study was run on a Dell Latitude 5400 on 6 parallel cores. For each experiment, the $n = 100$ simulations took roughly 8 minutes each, the $n = 500$ simulations took roughly 30 - 35 minutes each, and the $n = 1000$ simulations took roughly 1.5 hours each.

3.3.2 Simulation Results

The mean and standard error for the $\boldsymbol{\beta}$ coefficients and estimated centers of the DLFs are given in table 3.1. The results show some bias for every method, especially for $n = 100$, though seems to improve as the sample size increases.

The most difficult component to fit, going by the bias, was the second DLF. This was the DLF that was more severely dispersed across the time window so it is understandable that the model struggles to fit it. The experiment with extra covariate noise did particularly well here compared to the rest, likely since the extra variation is informative for the estimator. Regardless, the corresponding covariate β_2 estimate did not seem too badly affected by this. This seems to imply that even if the method fails to understand how the effect is distributed, it is still reliable at estimating the overall effect itself.

The worst performing experiment by far was where we used a small time window of 7 to fit the model. All parameters, apart from the intercept, show a huge amount of bias under this scenario. On the other hand, the larger time window had results roughly as good for most parameters as the other experiments. The only exception is again the position of the second DLF; it seems that allowing a larger time window made it even harder to locate the heavily dispersed DLF, though again the estimate for the overall effect size $\hat{\beta}_2$ appears unaffected. This lends credence to our advice in section 3.2.3 that it is probably much safer to overestimate τ than to underestimate it.

The general conclusion we can draw from this simulation study is that the method is reasonably accurate on average for large enough sample sizes, and we seem to certainly require more than $n = 100$ according to these results. It seems harder to locate the center of highly dispersed DLFs, but the effect sizes remain identifiable. From this we

can conclude that while the model is good at finding effect sizes of predictors, it may struggle to identify how exactly that effect is distributed across the lags. Whether or not this is an important issue will depend on the problem; for example it would be a serious flaw if attempting to find the so called 'critical window' of exposure vulnerability (Wilson et al., 2017b). If however, we are only interested in identifying potential predictors then it is less of a concern. Finally, these results seem to suggest that using a value of τ that is too small can be fatal for estimation.

Frequentist Simulation Results															
Statistic	β_0			β_1			β_2			DLF 1			DLF 2		
n	100	500	1000	100	500	1000	100	500	1000	100	500	1000	100	500	1000
True Value	**0**	**0**	**0**	**1.5**	**1.5**	**1.5**	**-1.5**	**-1.5**	**-1.5**	**7**	**7**	**7**	**2.5**	**2.5**	**2.5**
Rate 1/7	0 (0.01)	0 (0)	0 (0)	1.35 (0.03)	1.46 (0.01)	1.5 (0.01)	-1.41 (0.03)	-1.46 (0.01)	-1.5 (0.01)	18.71 (13.44)	6.84 (0.03)	6.99 (0)	2.55 (0.56)	1.71 (0.49)	1.13 (1.19)
Rate 1/30	0 (0.01)	0 (0)	0 (0)	1.35 (0.03)	1.47 (0.01)	1.49 (0.01)	-1.41 (0.03)	-1.47 (0.01)	-1.49 (0.01)	5.18 (0.13)	6.87 (0.03)	6.92 (0.04)	6.79 (4.3)	16.09 (9.51)	1.53 (0.57)
Time Window 7	-0.01 (0.01)	0 (0)	0 (0)	0.59 (0.02)	0.36 (0.01)	0.31 (0)	-0.57 (0.02)	-0.37 (0.01)	-0.33 (0.01)	2 (2.2)	-0.11 (0.05)	-0.08 (0.02)	0.99 (0.15)	-0.13 (0.2)	0 (0.08)
Time Window 25	-0.01 (0.01)	0 (0)	0 (0)	1.4 (0.03)	1.46 (0.01)	1.49 (0.01)	-1.47 (0.03)	-1.46 (0.01)	-1.49 (0.01)	5.22 (0.29)	6.91 (0.03)	7 (0)	4.47 (1.67)	5.37 (2.95)	1.84 (0.26)
Std. Deviation 3	0 (0.01)	0 (0)	0 (0)	1.35 (0.03)	1.48 (0.01)	1.51 (0)	-1.46 (0.02)	-1.49 (0.01)	-1.52 (0.01)	5.43 (0.1)	6.87 (0.03)	6.99 (0.01)	2.72 (0.68)	2.16 (0.19)	2.3 (0.12)

Table 3.1: Mean values of the $\hat{\boldsymbol{\beta}}$ estimates and DLFs, with corresponding standard error in parentheses underneath, rounded to the second decimal digit.

4 Bayesian Inference, Quantile Regression and Variable Selection

In this chapter we further hone in on our specific needs to analyse the vasculitis flare data. Besides the distributed lag effects, we have two other major concerns: a highly imbalanced response variable and a sizeable number of covariates that we wish to filter out of the model. On top of this, the relatively large standard errors we saw for some of the $\boldsymbol{\theta}$ estimators in table 3.1 suggest that they may benefit from some degree of regularisation. One way to do this is to add a penalty term to (3.9), for example

$$O = -2\sum_t y_t \ln(p_t) + (1-y_t)\ln(1-p_t) - c||\boldsymbol{\theta}||_2^2 \tag{4.1}$$

for some $c > 0$. This particular method is an example of *ridge regression* (Hoerl and Kennard, 1970) which is useful for overcoming issues associated with multicolinearity. Penalised regression methods such as LASSO (Tibshirani, 1996) and Elastic Net (Zou and Hastie, 2005) can be used for variable selection, which is another desirable feature we want to implement. However these methods come with the drawback that it is very difficult to derive uncertainty estimates of penalised regression coefficients (see for example Chatterjee and Lahiri (2010)). Zhao et al. (2021) and Leeb et al. (2015) suggest, for example, fitting a penalised model, and then fitting a standard non-penalised version to the data with only the selected variables discarded, from which standard errors can be calculated. While they offer theoretical and empirical justification, this method essentially assumes that the penalised inference is able to filter variables with 100% confidence since the refit model is implicitly conditioning on their exclusion. It also can't tell us how certain we can be the excluded variables are truly exclusions; it

is a pure binary in/out delineation. There is more that can be said about frequentist regularisation uncertainty estimates; see for example Chatterjee and Lahiri (2011); Lee et al. (2016); Tibshirani et al. (2016), among others, but here we instead pivot to a Bayesian approach as it is able to combine variable selection and imbalanced response inference within the distributed lag framework very elegantly, as we will show in this chapter.

4.1 Bayesian Model Specification

4.1.1 Binary Quantile Regression with Distributed Lag Parameters

The major issue with imbalanced binary response is that it can cause the model to essentially ignore the minority cases. In section 2.3 we spoke about how regressions utilising skewed link functions have shown promise in resolving this issue. Quantile regression is one such option. We briefly went through how latent variable representation has been used for efficient Bayesian inference of binary regressions in section 2.1.2 and in this section we will go through in more detail how to apply the same idea for binary quantile regression (Benoit and den Poel, 2012, 2017). We introduce a random variable $\mathbf{z}$ whose position is linearly dependent on the regression coefficients, whose sign determines exactly the value of the binary response,

$$\begin{aligned} z_i &= \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i \\ y_i &= I(z_i \geq 0) \\ \epsilon_i &\sim \text{ALD}(0, \sigma = 1, q) \end{aligned} \tag{4.2}$$

where the pdf of ALD distribution is (Yu and Zhang, 2005),

$$f_{\text{ALD}}(x|\mu, \sigma, q) = \frac{q(1-q)}{\sigma} \exp\left(-\frac{(x-\mu)}{\sigma}\left(q - I(x \leq \mu)\right)\right). \tag{4.3}$$

where $\mu \in \mathbb{R}$ is the location parameter, $\sigma \in \mathbb{R}^+$ is the scale, and $q \in (0, 1)$ is the skew

parameter, which corresponds to the desired quantile of interest. When $q = 0.5$ this collapses to the standard (symmetric) Laplace distribution. σ needs to be fixed in the residual term in equation (4.2) since $I(\mathbf{x}_i \top \boldsymbol{\beta}) = I(\sigma \mathbf{x}_i \top \boldsymbol{\beta})$ for all possible values of σ, making it non-identifiable when it's allowed to vary.

As mentioned, the skew parameter q corresponds to the desired quantile. Notice that as q changes, the likelihood is skewed so that the location parameter μ is equal to the q^{th} quantile. The importance of this is that it allows us to target specific quantiles of the response distribution. This means that for imbalanced response problems, we can set the value of q to be the proportion of binary response values equal to 0, i.e., the quantile that distinguishes the 'events' (=1) from the 'non-events' (=0). Kordas (2006) discusses more virtues of quantile based binary regression.

What we have described so far only applies for standard contemporaneous regression. But we can smoothly extend its application to distributed lag models by simply using $\mathbf{WX}$ as the design matrix, where $\mathbf{W}$ and $\mathbf{X}$ are as described in section 3.2. Of course, by doing this we are introducing the distributed lag parameters $\boldsymbol{\theta}$ (within the $\mathbf{W}$ matrix) that also requires inference.

4.1.2 Model Inference

To ease the burden of notation, from here on we will simply use $\mathbf{X}$ and $\mathbf{x}_i$ to denote the distributed lag design matrix and its rows respectively, with the understanding that this still includes the weight matrix $\mathbf{W}$ (and hence the distributed lag parameters $\boldsymbol{\theta}$ as part of it). We infer the parameters using Gibbs Sampling, since the full conditional distributions for most of the parameters are easily tractable (Benoit and den Poel, 2017). We start with the latent variable z_i: Kozumi and Kobayashi (2011) showed that the ALD can be equivalently expressed as a location-scale mixture of Gaussian distributions. i.e., the standard ALD variable ϵ_i can be expressed as

$$\epsilon_i = \psi \nu_i + \omega \sqrt{\nu_i} u_i$$

where u_i is a standard normal variable, ν_i is a standard exponential, and ψ and ω are deterministic functions of the skew parameter q,

$$\psi = \frac{1-2q}{q(1-q)}, \qquad \omega^2 = \frac{2}{q(1-q)}.$$

So when conditioning on ν_i, ϵ_i becomes Gaussian,

$$\epsilon_i|\nu_i \sim \mathrm{N}(\psi\nu_i, \omega^2\nu_i)$$

and therefore, combined with (4.2), we find that

$$z_i|y_i, \boldsymbol{\beta}, \boldsymbol{\theta}, \nu_i \sim \begin{cases} \mathrm{N}(\mathbf{x}_i^\top\boldsymbol{\beta} + \psi\nu_i, \omega^2\nu_i) & \text{restricted to the positive axis when } y_i = 1, \\ \mathrm{N}(\mathbf{x}_i^\top\boldsymbol{\beta} + \psi\nu_i, \omega^2\nu_i) & \text{restricted to the negative axis when } y_i = 0. \end{cases} \tag{4.4}$$

The full conditional distribution of the additional latent variable ν_i is

$$\nu_i|z_i, \boldsymbol{\beta}, \boldsymbol{\theta} \sim \mathrm{GIG}(1/2, \chi_i^2, \delta_i^2) \tag{4.5}$$

with

$$\chi_i^2 = \frac{(z_i - \mathbf{x}_i^\top\boldsymbol{\beta})^2}{\omega^2}, \qquad \delta_i^2 = 2 + \frac{\psi^2}{\omega^2}$$

So to summarise, the full conditional distributions of the latent variables z_i and ν_i are truncated normal and GIG respectively. We can sample from these as they are, or alternatively we can jointly update z_i and ν_i to improve Monte Carlo efficiency as described by Holmes and Held (2006) for logistic regression. The joint distribution can be factorised as

$$\pi(z_i, \nu_i|y_i, \boldsymbol{\beta}, \boldsymbol{\theta}) = \pi(\nu_i|y_i, z_i, \boldsymbol{\beta}, \boldsymbol{\theta})\pi(z_i|y_i, \boldsymbol{\beta}, \boldsymbol{\theta})$$

The first component is simply the full conditional of ν_i as above in equation (4.5), unchanged. The second component is z_i with ν_i marginalised out, returning it to a truncated ALD distribution,

$$z_i|y_i, \boldsymbol{\beta}, \boldsymbol{\theta} \sim \begin{cases} \text{ALD}(\mathbf{x}_i^\top \boldsymbol{\beta}, \sigma = 1, q) & \text{restricted to the positive axis when } y_i = 1, \\ \text{ALD}(\mathbf{x}_i^\top \boldsymbol{\beta}, \sigma = 1, q) & \text{restricted to the negative axis when } y_i = 0. \end{cases}$$

Since both the CDF and inverse CDF are readily tractable, the truncated ALD can be easily sampled from using the inversion method. Benoit and den Poel (2017) offer an alternative sampling technique that utilises the fact that the tails of the ALD can be treated as exponential distributions.

Thanks to the latent variables, the full conditional distribution of $\boldsymbol{\beta}$ is simply

$$\boldsymbol{\beta}|\mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\nu} \sim \text{N}(\mathbf{B}, \mathbf{V}) \tag{4.6}$$

where

$$\begin{aligned} \mathbf{V} &= (\mathbf{v}^{-1} + \mathbf{X}^\top \boldsymbol{\Omega}^{-1} \mathbf{X})^{-1} \\ \mathbf{B} &= \mathbf{V}(\mathbf{v}^{-1}\mathbf{b} + \mathbf{X}^{\text{T}} \boldsymbol{\Omega}^{-1}(\mathbf{z} - \psi \boldsymbol{\nu})) \\ \boldsymbol{\Omega} &= \text{diag}(\omega^2 \boldsymbol{\nu}), \end{aligned} \tag{4.7}$$

and $\text{diag}(\cdot)$ is the function $\mathbb{R}^n \to \mathbb{R}^{n \times n}$ that creates a diagonal matrix from the elements of the vector input.

The last group of parameters to consider are the DLF parameters $\boldsymbol{\theta}$. Let $\boldsymbol{\theta}_k$ denote the set of $\boldsymbol{\theta}$ that correspond to the DLF of covariate k, and let $\boldsymbol{\theta}_{(-k)}$ denote all DLF parameters *except* $\boldsymbol{\theta}_k$. The full conditional distribution of each $\boldsymbol{\theta}_k$ is (recall that $\boldsymbol{\theta}_{(-k)}$ is part of the design matrix $\mathbf{X}$),

$$\pi\big(\boldsymbol{\theta}_k|\mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\beta}, \boldsymbol{\theta}_{(-k)}\big) \propto \pi\big(\boldsymbol{\theta}_k\big) \exp\left(-\frac{1}{2}\Big(\boldsymbol{\beta}^\top \mathbf{X}^\top \boldsymbol{\Omega}^{-1} \mathbf{X} \boldsymbol{\beta} - 2(\mathbf{z} - \psi \boldsymbol{\nu})^\top \boldsymbol{\Omega}^{-1} \mathbf{X} \boldsymbol{\beta}\Big)\right). \tag{4.8}$$

We need to choose a prior for $\boldsymbol{\theta}_k$. The range of appropriate priors for each individual θ value will largely depend on what was chosen as the DLF. Different choices will not only

mean a different number of DLF parameters, but also the range of possible values they are restricted to (if anywhere). For example a beta DLF, which is simply a function that uses the kernel of the beta distribution normalised across all the lags in the time window (Ghysels et al., 2016), uses two DLF parameters that must both be positive. On the other hand, the Nealmon has as many DLF parameters as the user wants (corresponding to the degree of the Almon polynomial), and these parameters can take any value, with the exception of the parameter corresponding to the highest order coefficient of the polynomial, which must be negative to ensure that the lag weights decline to zero asymptotically. Therefore it is impossible to suggest a generic prior thought would be suitable for all DLF parameters, so from here we will restrict our discussion to the two-degree Nealmon DLF. This has two parameters, $\boldsymbol{\theta}_k = \{\theta_{k1}, \theta_{k2}\}$, where $\theta_{k1} \in \mathbb{R}$ and $\theta_{k2} \in \mathbb{R}^-$. This is the DLF we will be using for our flare data analysis. Notice that this can be re-expressed as (up to proportionality with respect to Δt):

$$\exp\left\{\frac{2\theta_2}{2}\left(\Delta t + \frac{\theta_1}{2\theta_2}\right)^2\right\} \tag{4.9}$$

i.e., the two-degree Nealmon has the same form as the PDF of the Gaussian distribution with mean $-\theta_{k1}/2\theta_{k2}$ and standard deviation $(-2\theta_{k2})^{-1}$, affording a meaningful interpretation of the $\boldsymbol{\theta}_k$ parameters to build a prior around. We will use

$$\theta_{k1} \sim \mathrm{N}(\mu_0, \sigma_0^2) \tag{4.10}$$

$$-\theta_{k2} \sim \mathrm{Exp}(\lambda_0) \tag{4.11}$$

with $\mu_0 = 15$ and $\sigma_0 = 5$ for the θ_{k1} prior, and $\lambda_0 = 1$ for the θ_{k2} prior. The mean of the resulting joint prior is $\{15, -1\}$, which corresponds to the majority of the DLF weight being placed on the $15/2 = 6.5$th lag, roughly corresponding to a week when the time index is daily. But the prior is very diffuse in terms of what the DLF center will take; for example if θ_1 is 23 (roughly corresponding to the 95th quantile of its prior) and θ_2 is -0.05 (roughly corresponding to the 5th quantile of its prior) the mean of the DLF will be centered at $23/0.1 = 230$. The takeaway point here is that our prior

is set so that the resulting DLF is centered close to the time window but is diffuse enough to not penalise large shifts too heavily. Our choice here reflects our a priori uncertainty around the effect of air quality and vasculitis flare propensity due to the lack of prior knowledge to draw from. As time goes on and more research and attention is (hopefully) given to this research area, future analyses may be able to benefit from more informative priors.

Regardless of what prior we choose, (4.8) does not resemble any standard distribution that we are familiar with, so we must rely on a Metropolis-within-Gibbs update. This requires us to choose a proposal distribution $g_\theta(\cdot)$, for example the two-dimensional normal random walk proposal:

$$g_\theta(\theta_{k1}^\star, \zeta^\star | \theta_{k1}, \zeta) = \mathrm{N}\left((\theta_{k1}, \zeta)^\top, \boldsymbol{\Sigma}_k\right) \tag{4.12}$$

where $\theta_{k2} = -\exp(\zeta)$, a transformation to ensure θ_{k2} remains negative, and $\boldsymbol{\Sigma}_k$ is the Gaussian covariance matrix which controls the average step size for each proposal. The acceptance probability is then

$$\alpha_\theta = \min\left(1, \frac{\pi(\boldsymbol{\theta}_k^\star | \boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\nu}, ...)}{\pi(\boldsymbol{\theta}_k | \boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\nu}, ...)} \frac{\exp(\zeta^\star)}{\exp(\zeta)}\right) \tag{4.13}$$

where k is a covariate index. The ratio term $\frac{\exp(\zeta^\star)}{\exp(\zeta)}$ is the Jacobian due to the θ_{k2} transformation.

Choosing an appropriate value for $\boldsymbol{\Sigma}_k$ is challenging. This is the well known 'Goldilocks' dilemma of MCMC based inference (see for example the first chapter of Gilks et al. (1995)); setting a value of $\boldsymbol{\Sigma}_k$ such that the proposed values are too extreme will result very few accepted draws, and by extension an inefficient exploration of the posterior. On the other hand, proposing values too close to the current values may result in many acceptances, but only because the proposals will suggest tiny incremental steps which will yield a very high degree of auto-correlation in the Markov chain and thus a lower effective sample size (see for example section 11.5 of Gelman et al. (2013)).

Instead of choosing a fixed value, we will use an *adaptive* proposal (Haario et al., 2001). The idea of an adaptive proposal is to adjust the value of $\boldsymbol{\Sigma}_k$ as the chain progresses in

such a way that we achieve an 'optimal' (in some sense) acceptance rate. The method is applied here as follows: let $\boldsymbol{\theta}_k^i$ be the i^{th} iterative of $\boldsymbol{\theta}_k$ in the MCMC chain. The adaptive covariance matrix $\boldsymbol{\Sigma}_k^i$ computed for iteration i is

$$\boldsymbol{\Sigma}_k^i = s_d \text{cov}(\boldsymbol{\theta}_k^1, \cdots, \boldsymbol{\theta}_k^{i-1}) + s_d \epsilon \mathbf{I}_d \tag{4.14}$$

where $\text{cov}(\cdot)$ is the empirical covariance function, ϵ is some arbitrary small positive constant, d is the number of dimensions we are proposing for (in our case, $d = 2$), $\mathbf{I}_d$ is the $d \times d$ identity matrix, and $s_d = 2.38^2/d$. This achieves the optimal acceptance rate as defined by Gelman et al. (1996). A few important remarks about equation (4.14):

First, it should be noted that, because of the explicit dependence on the full history of the chain, this does not actually posses the Markov property. However, Haario et al. (2001) proved that despite this caveat, the ergodic distribution of the adaptive chain described above is still the exact posterior distribution, which is the only real concern. This is partially due to the fact that dependence on the history of the chain diminishes as the iterations increase, so that it essentially acts as a Markov chain asymptotically.

Second, the small positive constant term ϵ is a technical requirement in order to satisfy the conditions for true posterior ergodicity. Haario et al. (2001) claim that simply setting ϵ to zero may not be present much problems in practice based on their own observations, but we will use a small non-zero term for safety.

Third, the full empirical covariance becomes expensive (both in memory and speed) to calculate the longer the chain runs. For this reason it is strongly suggested to use the recursive formula for empirical covariance:

$$\boldsymbol{\Sigma}_k^{i+1} = \frac{i-1}{i}\boldsymbol{\Sigma}_k^i + \frac{s_d}{i}\left(i\bar{\boldsymbol{\theta}}_k^{i-1}\bar{\boldsymbol{\theta}}_k^{i-1^\top} - (i+1)\bar{\boldsymbol{\theta}}_k^i\bar{\boldsymbol{\theta}}_k^{i^\top} + \boldsymbol{\theta}_k^i\boldsymbol{\theta}_k^{i^\top} + \epsilon\mathbf{I}_d\right)$$

where $\bar{\boldsymbol{\theta}}_k^i$ denotes the mean of $\left(\boldsymbol{\theta}_k^1, \cdots \boldsymbol{\theta}_k^i\right)$ which also has a recursive formula,

$$\bar{\boldsymbol{\theta}}_k^i = \frac{i-1}{i}\bar{\boldsymbol{\theta}}_k^{i-1} + \frac{1}{i}\boldsymbol{\theta}_k^i.$$

Fourth, the algorithm requires a user-specified starting value $\boldsymbol{\Sigma}_k^1$. When lacking any real prior knowledge about the parameters (which is the case here) this becomes an arbitrary choice, though it is not that important as it is updated every iteration to an optimal value; if the proposal of $\boldsymbol{\theta}$ is rejected/accepted too often it will be corrected to attain the optimal acceptance rate. We use the following:

$$\boldsymbol{\Sigma}_k^1 = \begin{pmatrix} 6 & 0.5 \\ 0.5 & 0.05 \end{pmatrix}$$

the top leftmost cell is the variance of the θ_1 parameter, which we set to 6. Since θ_1 is the numerator of the location we want to propose large jumps to give it the opportunity to find peaks in the DLF. The variance for ζ is smaller as it controls the denominator so choose to start with we make smaller more careful proposals to avoid the DLF location from exploding in value. We use non-zero values in the off-diagonal to induce correlation in the proposals. Again, we emphasise that this covariance will be adjusted over the MCMC iterations to attain optimal acceptance rates so if our starting value is off it will be corrected as the chain progresses.

There are many other options we could have gone with besides the above adaptive random walk approach, for example the Metropolis-Adjusted Langevin Algorithm (Roberts and Tweedie, 1996), which would utilise the gradient computed in section 3.2.5 (though this may also benefit from adaptive methods as it requires a set tuning parameter). This possibility is discussed more in chapter 6.

4.2 Variable Selection

In this section we implement Reversible-Jump MCMC (Green, 1995) which we gave a basic overview of in section 2.4.2. As a reminder, we introduce binary parameters $\gamma_k \in \{0, 1\}$ that determine whether or not covariate k is retained in the model. We can then update $\boldsymbol{\gamma}$ using Metropolis-Hastings updates, making sure the acceptance probability takes into account the birth or death of associated parameters. This is a perfectly adequate setup for this situation, however we can somewhat simplify the process using a method proposed by Holmes and Held (2006). For the moment, we

will focus on adaptation only for the standard (contemporaneous, non distributed lag) quantile binary regression model. We will show how to extend this when incorporating distributed lags in section 4.3.

The method works by updating $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ jointly and factorizing as follows;

$$\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}|\mathbf{z}, \boldsymbol{\nu}) = \pi(\boldsymbol{\gamma}|\mathbf{z}, \nu)\pi(\boldsymbol{\beta}|\boldsymbol{\gamma}, \mathbf{z}, \boldsymbol{\nu})$$

$\boldsymbol{\beta}$ is updated as described before, but using only the columns of the design matrix supported by $\boldsymbol{\gamma}$. As for the $\boldsymbol{\gamma}$ term we still update it using Metropolis-Hastings using the following proposal,

$$g_\gamma(\boldsymbol{\beta}^\star, \boldsymbol{\gamma}^\star|\mathbf{z}, \boldsymbol{\nu}) = \pi(\boldsymbol{\beta}^\star|\boldsymbol{\gamma}^\star, \mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}^\star|\boldsymbol{\gamma}) \tag{4.15}$$

The unbounded acceptance rate is therefore

$$\begin{aligned}
r &= \frac{\pi(\boldsymbol{\beta}^\star, \boldsymbol{\gamma}^\star|\mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\beta}, \boldsymbol{\gamma})}{\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}|\mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\beta}^\star, \boldsymbol{\gamma}^\star)} && (4.16)\\
&= \frac{\pi(\boldsymbol{\beta}^\star, \boldsymbol{\gamma}^\star|\mathbf{z}, \boldsymbol{\nu})\pi(\boldsymbol{\beta}|\boldsymbol{\gamma}, \mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}|\boldsymbol{\gamma}^\star)}{\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}|\mathbf{z}, \boldsymbol{\nu})\pi(\boldsymbol{\beta}^\star|\boldsymbol{\gamma}^\star, \mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}^\star|\boldsymbol{\gamma})} && (4.17)\\
&= \frac{\pi(\boldsymbol{\beta}^\star|\boldsymbol{\gamma}^\star, \mathbf{z}, \boldsymbol{\nu})\pi(\boldsymbol{\beta}|\boldsymbol{\gamma}, \mathbf{z}, \boldsymbol{\nu})\pi(\boldsymbol{\gamma}^\star|\mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}|\boldsymbol{\gamma}^\star)}{\pi(\boldsymbol{\beta}|\boldsymbol{\gamma}, \mathbf{z}, \boldsymbol{\nu})\pi(\boldsymbol{\beta}^\star|\boldsymbol{\gamma}^\star, \mathbf{z}, \boldsymbol{\nu})\pi(\boldsymbol{\gamma}|\mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}^\star|\boldsymbol{\gamma})} && (4.18)\\
&= \frac{\pi(\boldsymbol{\gamma}^\star|\mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}|\boldsymbol{\gamma}^\star)}{\pi(\boldsymbol{\gamma}|\mathbf{z}, \boldsymbol{\nu})g_\gamma(\boldsymbol{\gamma}^\star|\boldsymbol{\gamma})} && (4.19)
\end{aligned}$$

Notice that $\boldsymbol{\beta}$ has been entirely cancelled out - in other words, we can update $\boldsymbol{\gamma}$ without explicitly including the dimension transitioning regression parameter. This makes intuitive sense since 'removing' a regression coefficient is equivalent to simply fixing it at zero. Since the dimensions of $\mathbf{z}$ and $\boldsymbol{\lambda}$ are unaffected by $\boldsymbol{\gamma}$, this means there are actually no dimensional transitions when updating $\boldsymbol{\gamma}$.

Using Bayes Theorem and re-arranging we find

$$\pi(\boldsymbol{\gamma}|\mathbf{z}, \boldsymbol{\nu}) \propto \frac{\pi(\boldsymbol{\gamma}|\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\nu})}{\pi(\boldsymbol{\beta}|\boldsymbol{\gamma}, \mathbf{z}, \boldsymbol{\nu})}$$

which works out to be

$$\pi(\boldsymbol{\gamma}|\mathbf{z},\boldsymbol{\nu}) \propto |\mathbf{v}_{\boldsymbol{\gamma}}|^{-1/2}|\mathbf{V}_{\boldsymbol{\gamma}}|^{1/2}\exp\left\{\frac{1}{2}\left(\mathbf{B}_{\boldsymbol{\gamma}}^{\top}\mathbf{V}_{\boldsymbol{\gamma}}^{-1}\mathbf{B}_{\boldsymbol{\gamma}} - \mathbf{b}_{\boldsymbol{\gamma}}^{\top}\mathbf{v}_{\boldsymbol{\gamma}}^{-1}\mathbf{b}_{\boldsymbol{\gamma}}\right)\right\}\pi(\boldsymbol{\gamma})$$

where $\pi(\boldsymbol{\gamma})$ is the prior distribution of $\boldsymbol{\gamma}$, $\mathbf{b}$ and $\mathbf{v}$ are the mean and covariance for the prior of $\boldsymbol{\beta}$ and $\mathbf{B}$ and $\mathbf{V}$ are the mean and covariance of the posterior of $\boldsymbol{\beta}$, as given in equations 4.7. The usage of $\boldsymbol{\gamma}$ as a subscript for the matrices/vectors is to denote the fact that we only use the columns/elements corresponding to the covariates chosen by $\boldsymbol{\gamma}$. The resulting M-H acceptance ratio α_γ is

$$\alpha_\gamma = \min\left(1, \frac{|\mathbf{v}_{\boldsymbol{\gamma}^\star}|^{-1/2}|\mathbf{V}_{\boldsymbol{\gamma}^\star}|^{1/2}\exp\left\{\frac{1}{2}\left(\mathbf{B}_{\boldsymbol{\gamma}^\star}^{\top}\mathbf{V}_{\boldsymbol{\gamma}^\star}^{-1}\mathbf{B}_{\boldsymbol{\gamma}^\star} - \mathbf{b}_{\boldsymbol{\gamma}^\star}^{\top}\mathbf{v}_{\boldsymbol{\gamma}^\star}^{-1}\mathbf{b}_{\boldsymbol{\gamma}^\star}\right)\right\} g_\gamma(\boldsymbol{\gamma})\pi(\boldsymbol{\gamma}^\star)}{|\mathbf{v}_{\boldsymbol{\gamma}}|^{-1/2}|\mathbf{V}_{\boldsymbol{\gamma}}|^{1/2}\exp\left\{\frac{1}{2}\left(\mathbf{B}_{\boldsymbol{\gamma}}^{\top}\mathbf{V}_{\boldsymbol{\gamma}}^{-1}\mathbf{B}_{\boldsymbol{\gamma}} - \mathbf{b}_{\boldsymbol{\gamma}}^{\top}\mathbf{v}_{\boldsymbol{\gamma}}^{-1}\mathbf{b}_{\boldsymbol{\gamma}}\right)\right\} g_\gamma(\boldsymbol{\gamma}^\star)\pi(\boldsymbol{\gamma})}\right) \tag{4.20}$$

Our method of proposing new $\boldsymbol{\gamma}$ values, which we will use for all further analysis, is to simply select a covariate with uniform randomness and swap its corresponding γ value. Since this results in a symmetric proposal, the $g_\gamma(\cdot)$ terms in equation (4.20) will cancel out.

4.2.1 Variable Indicator Prior

Since $\boldsymbol{\gamma}$ is a binary vector, the Bernoulli distribution is the most appropriate distribution for the prior. We must specify the Bernoulli probability of success parameter p which in this case is interpreted as our prior degree of belief that the covariate is a true predictor. A common choice is 0.5, as it is unbiased and the most uninformative (in the sense that it results in the largest prior variance), but it translates to assuming a priori that 50% of the covariates meaningfully explain the variance of the response, which seems unlikely to us. Our alternative solution is to use a hierarchical prior for $\boldsymbol{\gamma}$, by setting a hyperprior on p,

$$p \sim \text{Beta}(a, b) \tag{4.21}$$

since p only takes values between 0 and 1. We will discuss our choice of hyperparameters

a and b in a moment.

The full conditional distribution of p (which is only dependent on $\boldsymbol{\gamma}$) is then

$$p|\boldsymbol{\gamma} \sim \text{Beta}\left(a + \sum_i \gamma_i,\ b + \sum_i (1 - \gamma_i)\right).$$

The advantage here is that we essentially let the data decide the most appropriate prior, based on how many inclusions are currently in the model. The values of a and b are chosen to best represent our prior belief in the overall number of parameters in the model. In the absence of expert opinion, this becomes a matter of personal judgement. To help us decide on appropriate values, we need to consider how many variables will be under consideration. Excluding the intercept, our dataset for the flare modelling analysis will have 18 covariates. With this in mind, let us consider the situation where we set a and b both equal to 1. This is conceptually appealing as a starting point for non-experts since the resulting hyperprior is uniform in (0, 1) meaning that any configuration of variables is equally likely on average a priori. But this may induce a peak in the posterior in regions where every variable is rejected (or included) in the model as the full-conditional mode will be situated at 0 (or 1). As a result, the Markov chain may get stuck in these regions in the event that every variable is included or excluded (via initialisation, for instance). For that reason we believe it is safer to set a and b equal to 3. This means that, for 18 variables, the full-conditional mode will never fall below 0.1 or above 0.9, working as a sort of safety buffer in the event the algorithm swings too far one way or the other during burn-in, but still allows enough room for the chain to reach the boundary in the event that the bulk of the posterior is truly situated there. Figure 4.1 illustrates this point. In section 4.5 we will test this line of thought through simulation experiments.

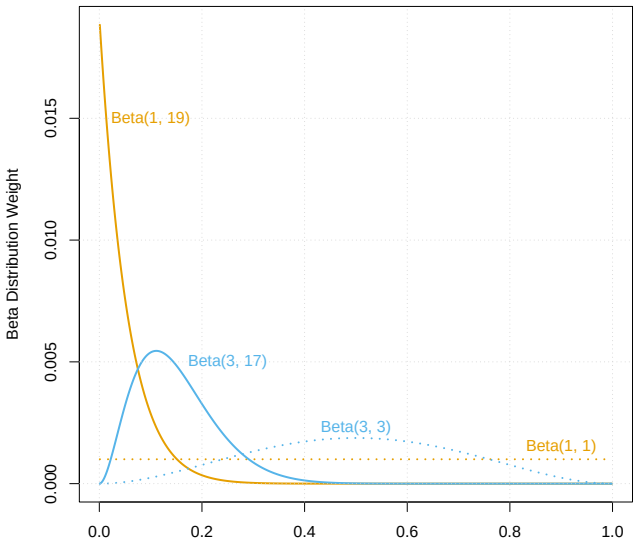


Figure 4.1: Two different choice of hyperpriors and how they affect the full conditional distribution of the prior inclusion parameter. The dashed lines correspond to hyperpriors, and solid lines show the resulting full conditionals when there are 18 covariates and all their corresponding indicators are set to zero. The (1, 1) hyperprior results in a very sharp peak at 0 in the full conditional, which may result in the Markov chain getting stuck. The (3, 3) hyperprior on the other hand never peaks too sharply near the boundary.

4.3 Variable Selection With DLF Parameters

When incorporating distributed lags into the above variable selection method, trans-dimensionality starts to become a concern since the dimensionality of the DLF parameters $\boldsymbol{\theta}$ depends on $\boldsymbol{\gamma}$. Whenever a new value of $\boldsymbol{\gamma}$ is proposed, we must take into account the birth or death of the relevant $\boldsymbol{\theta}$ variables when calculating the acceptance probability. Although we are back to using RJ-MCMC, the above exposition for standard quantile regression is still be useful here, since we can still jointly update $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$.

If the proposed update intends to birth a new variable then we must propose a new $\boldsymbol{\theta}$ value as well. Say we are proposing to birth covariate k (and thus newly introduce the corresponding $\boldsymbol{\theta}_k$). The proposal distribution in this case, which we call $g_{k^+}(\cdot)$, is

$$g_{k^+}(\boldsymbol{\beta}^\star, \boldsymbol{\theta}_k^\star, \boldsymbol{\gamma}^\star | \mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\theta}_{-k}) = \pi(\boldsymbol{\beta}^\star | \boldsymbol{\gamma}^\star, \mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\theta}_k^\star, \boldsymbol{\theta}_{-k}) g_\gamma(\boldsymbol{\gamma}^\star | \boldsymbol{\gamma}) g_{\theta^+}(\boldsymbol{\theta}_k^\star) \tag{4.22}$$

where $g_{\theta^+}(\cdot)$ is the proposal distribution for the $\boldsymbol{\theta}_k$ parameters that are to be birthed (which is different than the proposal $g_\theta(\cdot)$ used in equation (4.12)). An appropriate proposal will of course depend on the choice of DLF. As before, we restrict our attention to the two-degree Nealmon DLF. Each $\boldsymbol{\theta}_k$ is then a vector of length two, $\{\theta_1, \theta_2\}$. For $\boldsymbol{\theta}$ birth proposals, we use a normal distribution centered at 0 with standard deviation σ. $\theta_1^\star$ can be proposed directly from this, but since $\theta_2^\star$ must be negative, we instead sample a random variable $\zeta^\star = \log(-\theta_2)$ and transform, same as for the $\boldsymbol{\theta}$ updates in section 4.1.2. The values of σ we use are 5 for θ_1 and 2 for ζ.

The acceptance probability is

$$\alpha_{k^+} = \min\left(1, r \frac{\pi(\theta_1^\star)\pi(\theta_2^\star)}{g_{\theta^+}(\theta_1^\star) g_{\theta^+}(\zeta^\star)} \exp(\zeta^\star)\right) \tag{4.23}$$

where r is referring to equation (4.19). The $\exp(\zeta^\star)$ factor is the Jacobian to correct the probability measure due to transformation to θ_2.

If the proposed update intends to *remove* covariate k instead ($\gamma_k^\star = 0$), the proposal becomes

$$q_{k-}(\boldsymbol{\beta}^{\star}, \boldsymbol{\gamma}^{\star}|\mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\theta}) = \pi(\boldsymbol{\beta}^{\star}|\boldsymbol{\gamma}^{\star}, \mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\theta}) g_{\gamma}(\boldsymbol{\gamma}^{\star}|\boldsymbol{\gamma}). \tag{4.24}$$

The corresponding acceptance probability α_{k-} must take into account that we are potentially removing a component of $\boldsymbol{\theta}$, so

$$\alpha_{k-} = \min\left(1, r\frac{g_{\theta^+}(\theta_1) g_{\theta^+}(\zeta)}{\pi(\theta_1)\pi(\theta_2)} \frac{1}{\exp(\zeta)}\right). \tag{4.25}$$

After sampling from the posterior using RJ-MCMC, the proportion of draws where γ_i is equal to 1 represents the posterior probability that the corresponding variable should be retained. A proportion close to the implies that we can accept or reject the possibility of the corresponding covariate of being predictive with high confidence, at least according to our data. A proportion closer to 50% suggests that the data does not strongly support inclusion or exclusion.

With this, we have fully specified the Bayesian Quantile MIDAS model. A step-by-step algorithm of the method is given in appendix A.

4.4 Posterior Predictive Distribution

The model samples from

$$\pi(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\gamma}, p|\mathbf{y}) \propto \pi(\mathbf{y}|\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\gamma}, p)\pi(\boldsymbol{\beta}|\boldsymbol{\gamma})\pi(\boldsymbol{\theta}|\boldsymbol{\gamma})\pi(\boldsymbol{\gamma}|p)\pi(p)$$

by employing latent variables. As a part of model validation as well as out of sample prediction, we will compute a Monte Carlo estimate of the posterior predictive distribution (PPD) using the MCMC samples. The PPD for class $\mathbf{y}_{\text{new}}$, given observed predictors $\mathbf{X}_{\text{new}}$ is

$$\pi(\mathbf{y}_{\text{new}}|\mathbf{X}_{\text{new}}) = \sum_{\boldsymbol{\gamma}\in\Gamma} \int_{\mathcal{D}(\boldsymbol{\theta})} \int_{\mathcal{D}(\boldsymbol{\beta})} \pi(\mathbf{y}_{\text{new}}|\mathbf{X}_{\text{new}}, \boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\gamma}) \pi(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\gamma}|\mathcal{D}) d\boldsymbol{\beta} d\boldsymbol{\theta} \tag{4.26}$$

where Γ denotes the space of all potential models encoded by $\boldsymbol{\gamma}$, and $\mathcal{D}(\cdot)$ denotes the domain of the given parameter. Indexing the MCMC samples retained after burn in and thinning as $(\boldsymbol{\beta}^1, \boldsymbol{\theta}^1, \boldsymbol{\gamma}^1), \ldots, (\boldsymbol{\beta}^N, \boldsymbol{\theta}^N, \boldsymbol{\gamma}^N)$, the Monte Carlo estimate is simply:

$$\pi(\mathbf{y}_{\text{new}}|\mathbf{X}_{\text{new}}) \approx \frac{1}{N}\sum_{i}^{N} \pi(\mathbf{y}_{\text{new}}|\mathbf{X}_{\text{new}}, \boldsymbol{\beta}^i, \boldsymbol{\theta}^i, \boldsymbol{\gamma}^i). \tag{4.27}$$

The likelihood for the binary ALD model is

$$\pi(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\gamma}) = \prod_{i=1}^{n} F_{\text{ALD}}(\mathbf{x}_i^\top \boldsymbol{\gamma}\boldsymbol{\beta}|0, 1, q)^{y_i}(1 - F_{\text{ALD}}(\mathbf{x}_i^\top \boldsymbol{\gamma}\boldsymbol{\beta}|0, 1, q))^{1-y_i} \tag{4.28}$$

where $\mathbf{x}_i\boldsymbol{\gamma}$ is the i^{th} row of $\mathbf{WX}$ using only the columns supported by $\boldsymbol{\gamma}$, and F_{ALD} is the cumulative distribution function of the ALD:

$$F_{ALD}(x|\mu, \sigma, q) = \begin{cases} q \exp\left(-(q-1)\frac{x-\mu}{\sigma}\right) & \text{if } x > \mu \\ 1 - (1-q)\exp\left(-q\frac{x-\mu}{\sigma}\right) & \text{otherwise.} \end{cases} \tag{4.29}$$

4.5 Simulation Study

In this section we run a simulation study for the method described in this chapter. This method is a lot more computationally intensive than the frequentist one so we will not be able to run as many simulations we had in chapter 3. Although many characteristics of this study will differ from that previous simulation study (such as the fact that we will be generating many more covariates per dataset), there are some aspects that are the same or similar, so as a result there will be some repetition in our exposition. Perhaps the main difference here is the purpose of this study: our primary concern here will be to examine how effective the method is at identifying the correct covariate set used to generate the data. This is important as the ability to select true predictors is the main motivating problem of the vasculitis flare analysis we will talk about in section 4.6.

Data Generation and Simulation Details

We generate 1000 irregular response observations. To do this, we first create its time indices by sampling 1000 exponential random variables with a rate parameter of 1/15 (i.e., we expect on average one observation for every 15 units of time). The time indices are their cumulative sum. With this we use a Nealmon DLF of degree 2 and a time window 30 units wide to create the weight matrix $\mathbf{W}$.

We create 18 covariates; this is the same number as our dataset in the flare analysis in section 4.6. The covariates are independent autoregression time series of order one, generated using the `mlVAR` package (Epskamp et al., 2021),

$$\begin{aligned} x_t &= 0.5x_{t-1} + 2\cos\left(\frac{2\pi}{\omega}t + \phi\right) + \epsilon_t \\ \epsilon_t &\sim \mathrm{N}(0,1) \end{aligned} \tag{4.30}$$

This is the same method we used to generate covariates for the simulation study in chapter 3. As a reminder, the cosine term adds seasonality. ω and ϕ control the frequency and shift of the wave respectively. We will discuss the values we use for these parameters further below. We generate as many observations as we need for all the time windows to be fully populated. This will be a different number for each simulation due to the irregularity of the response indices.

With the covariates and weight matrix, we use the CDF of the ALD as the link function to create the binary response. The skew parameter q is set to control the degree of imbalance in the simulated response. We partition the data into training and test sets using an 80/20 split. The predictive performance will be evaluated by computing the AUC of the PPD we derived in section 4.4.

We will perform a number of experiments for a number of different parameter settings. The main purpose of this simulation experiment is to examine how effectively the RJ-MCMC algorithm selects the correct variables. Unless otherwise stated, the following parameters are used to generate the simulated data:

- For the main parameter of interest $\boldsymbol{\gamma}$, 4 of the 18 covariates are set to 1, and the remainder are set to zero. In other words, the majority of the dataset is pure

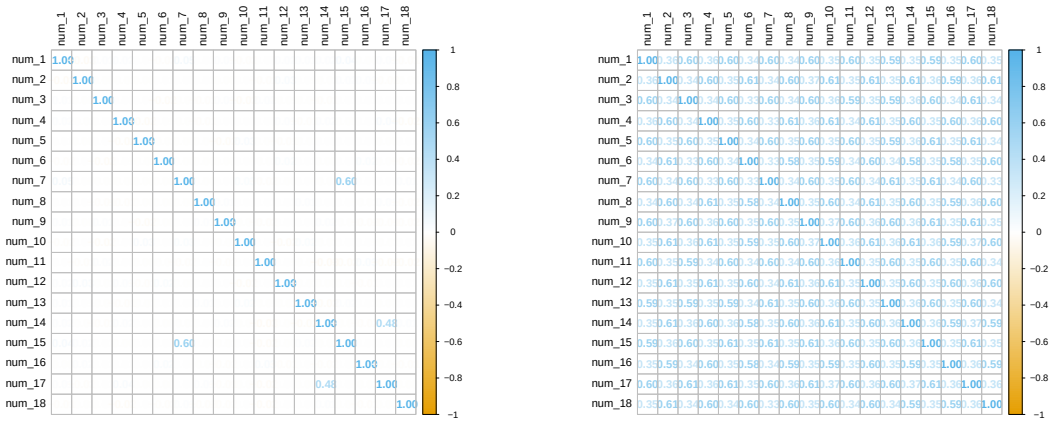
noise.

- The non-zero elements of $\boldsymbol{\beta}$ are set to $\{1.5, -1.5, 1, -1\}$, so we have two strong effects and two more of middling strength.
- The DLF parameters we use for the two degree Nealmon are $\{14, -1\}$ for the positive effect covariates and $\{0.5, -0.1\}$ for the negative effect covariates. These curves were illustrated in figure 2.1. The first resulting DLF is heavily concentrated on the 7th lag unit, the other is much more dispersed and centered at the 2.5th lag unit, allowing us to test for two different kinds of DLF.
- We use a time window length of 15.

There are specific inferential characteristics we want to test. These are:

1. **Effect of the hierarchical prior for the variable indicator:** We want to see how the model performs under different settings of the Beta hyperprior for the prior probability of inclusion p. We will use a Beta(1, 1) and Beta(3, 3).
2. **Effect of misspecified time window:** We wish to see how the model performs when it is set too narrow at 7 units wide, and also too wide at 25 units wide.
3. **Effect of misspecified DLF:** We will use a Beta distribution for simulating the dataset, and the Nealmon for fitting the model.
4. **Effect of Covariate Noise:** We will run a simulation where the residual term in equation (4.30) has a standard deviation of 3 instead of 1.

The above experiments are run under four different settings for different levels of response imbalance and collinearity in the covariates, to test the RJ-MCMC method under these conditions. To control imbalance, we simply set the ALD skew parameter to $q = 0.5$ for balanced simulations and 0.2 for imbalanced. Controlling the level of multicollinearity is done through the ω (which controls the frequency/seasonality of the wave) and ϕ (which controls the shift) parameters in equation (4.30). Time series with similar frequency and shift parameters will be very similar and thus will exhibit a large degree of correlation. With this in mind, to induce a mostly independent covariate set, the frequency and shift parameters are chosen randomly. To induce high collinearity, we use $\omega = 30$ for all covariates and alternate $\phi = 2.5$ and 7 for each variable. So



(a) Random frequency and shift parameters. (b) Fixed frequency and shift parameters.

Figure 4.2: Example correlation matrices for both versions of covariate simulation. (a) displays a mostly independent covariate set, while (b) displays a large degree of multicollinearity.

all covariate time series will have the same frequency, and the same shift as 8 of the others. Typical correlation matrices from these settings are given in figure 4.2. All in all, we run $6 \times 4 = 24$ simulation experiments.

For each dataset randomly generated, the model will be fit 30 times to assess Monte Carlo consistency. Each MCMC algorithm is run for 80,000 iterations. The first half of the iterations are discarded as burn in, and only every 10^{th} value is retained thereafter, for a total of 4,000 draws from the posterior distribution.

We emphasise again that for this simulation study our main interest lies in how accurate the variable selection routine is, so those are the only results we will display. However many other plots and diagnostics for the other parameters can be found in https://github.com/DanDempsey/DD_Thesis_Files/tree/master/BayesQMIDAS/package_code/Simulations/BayesQMIDAS_Sims/Output.

4.5.1 Simulation Results

The proportion of selected variables are presented in figures 4.3-4.6 in order of increasing difficulty. The blue boxes denote the non-zero predictors that were used to generate the response data, whereas orange denotes the covariates that had zero effect on the data generation that we hope RJ-MCMC filters out. The intercepts, which are always included with 100% probability, are not displayed on these graphs.

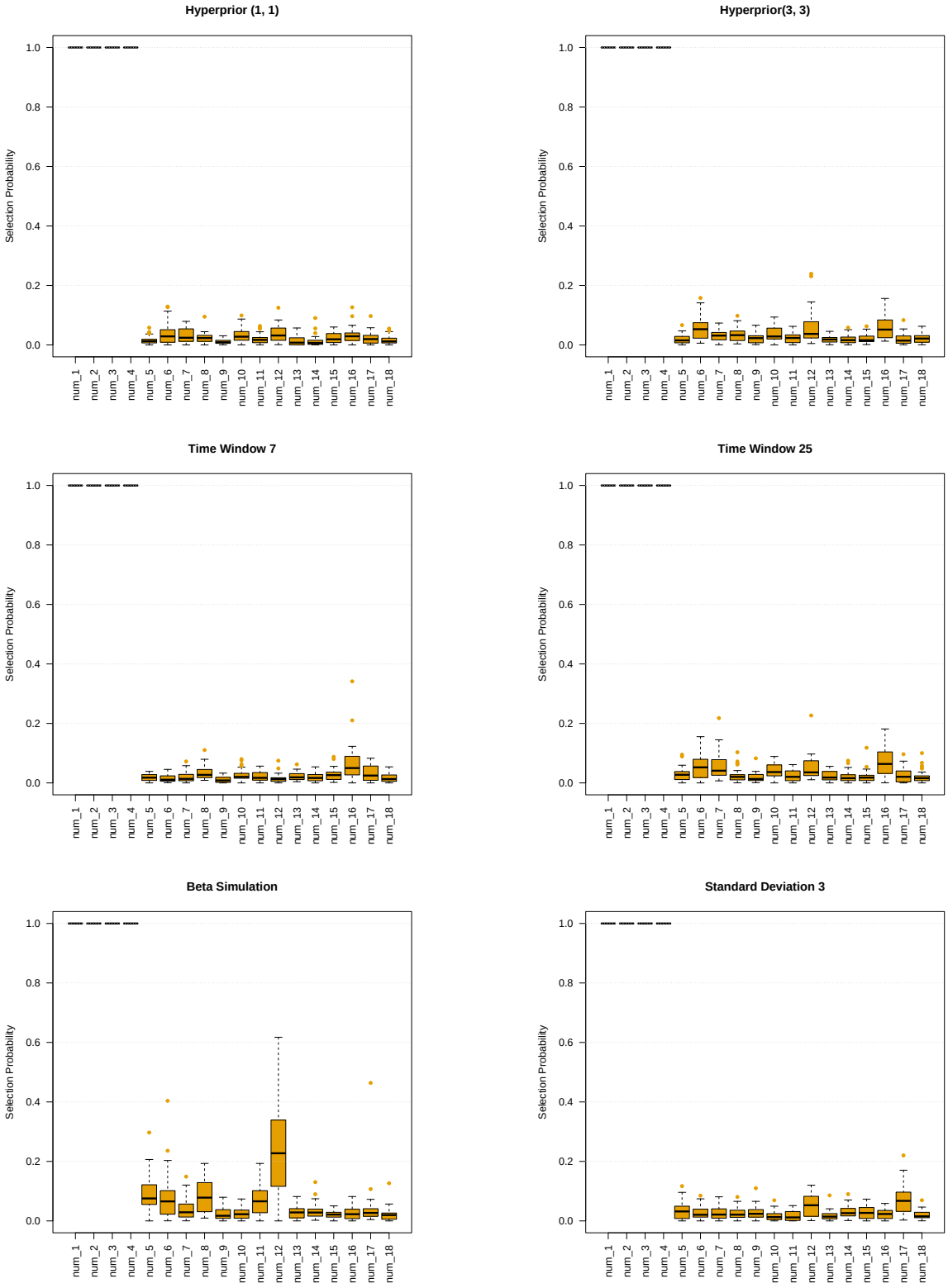


Figure 4.3: Variable inclusion for all the simulated datasets across 30 runs each for **balanced simulations** with **low collinearity**.

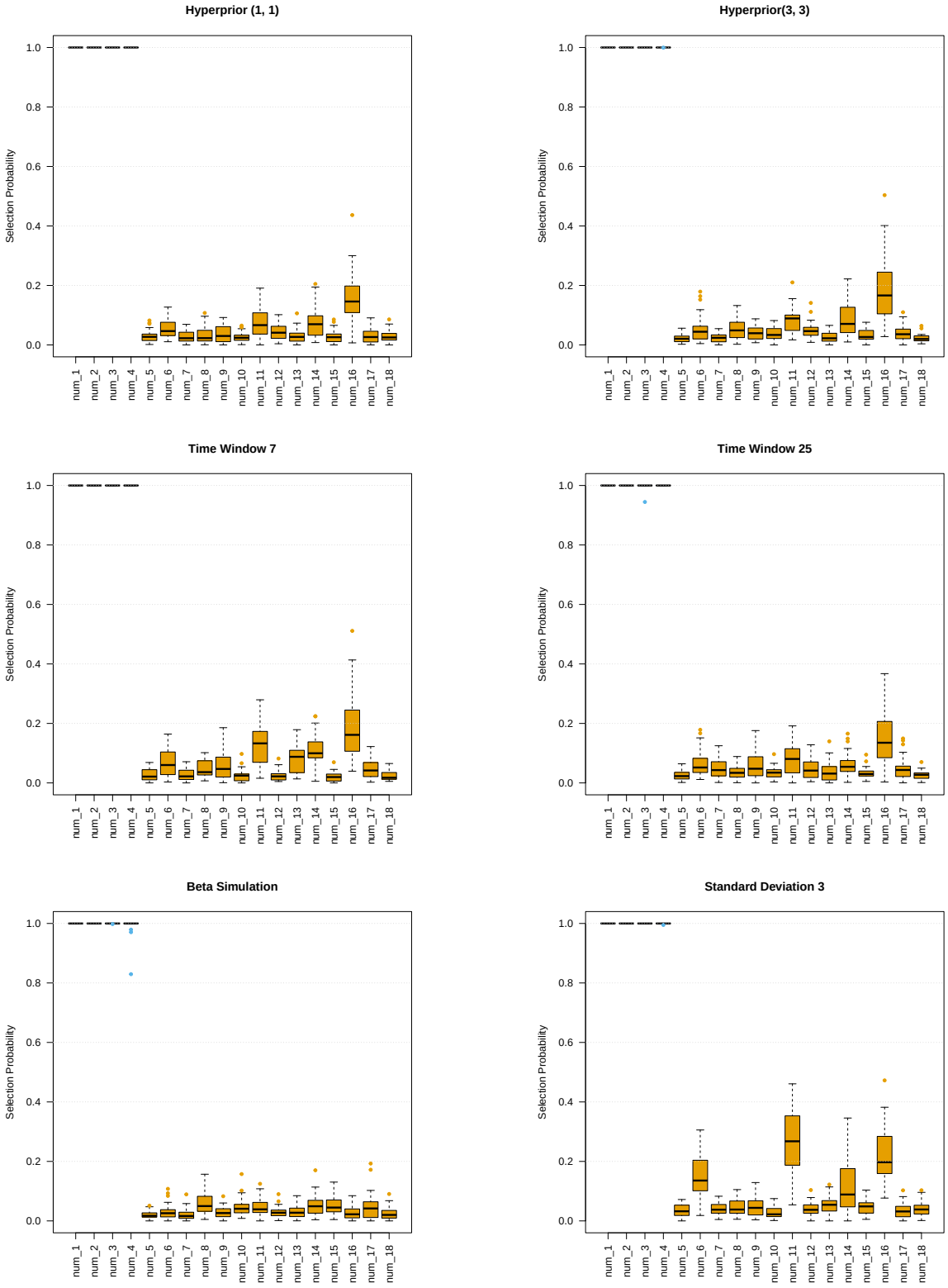


Figure 4.4: Variable inclusion for all the simulated datasets across 30 runs each for **imbalanced simulations** with **low collinearity**.

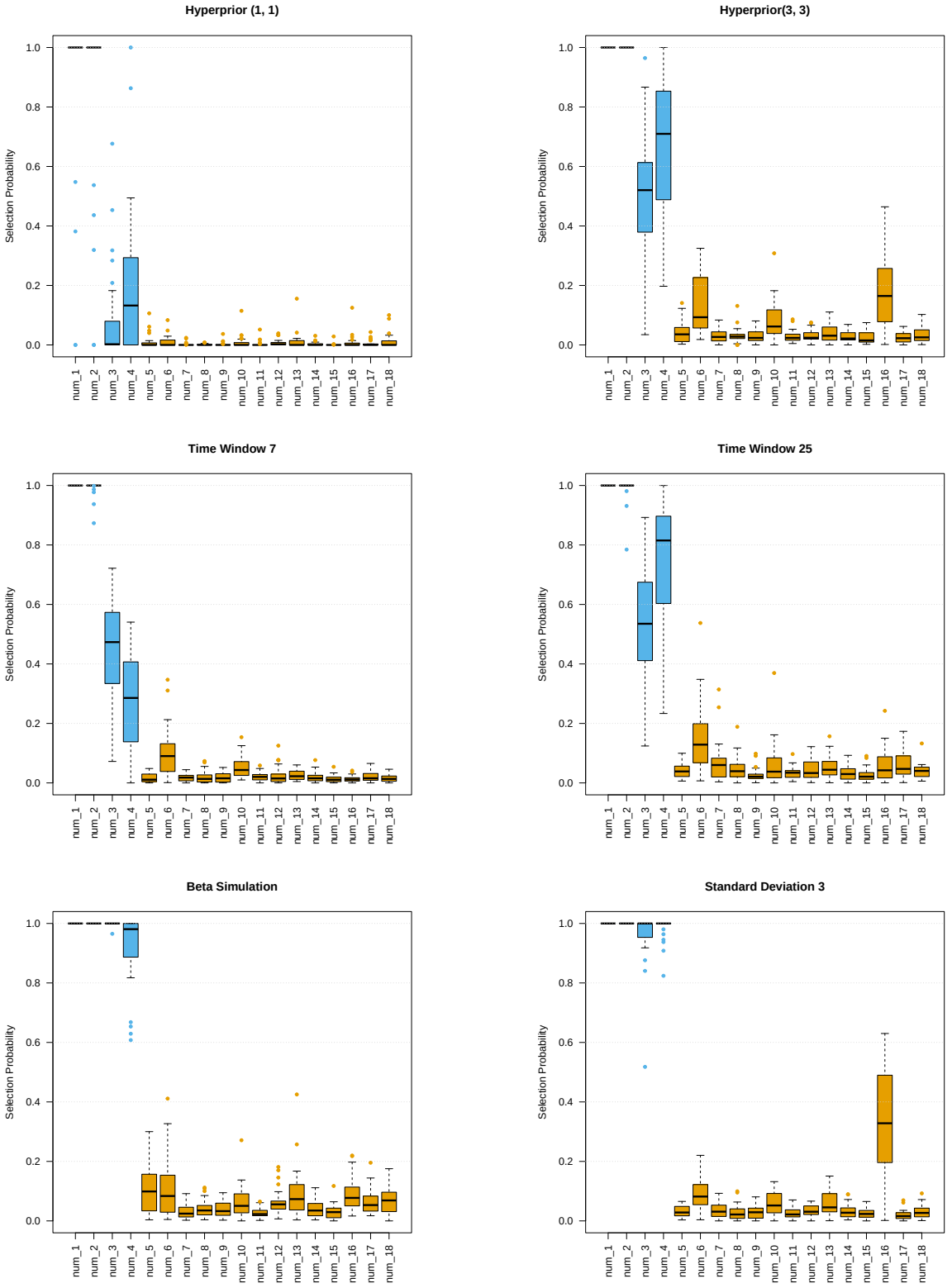


Figure 4.5: Variable inclusion for all the simulated datasets across 30 runs each for **balanced simulations** with **high collinearity**.

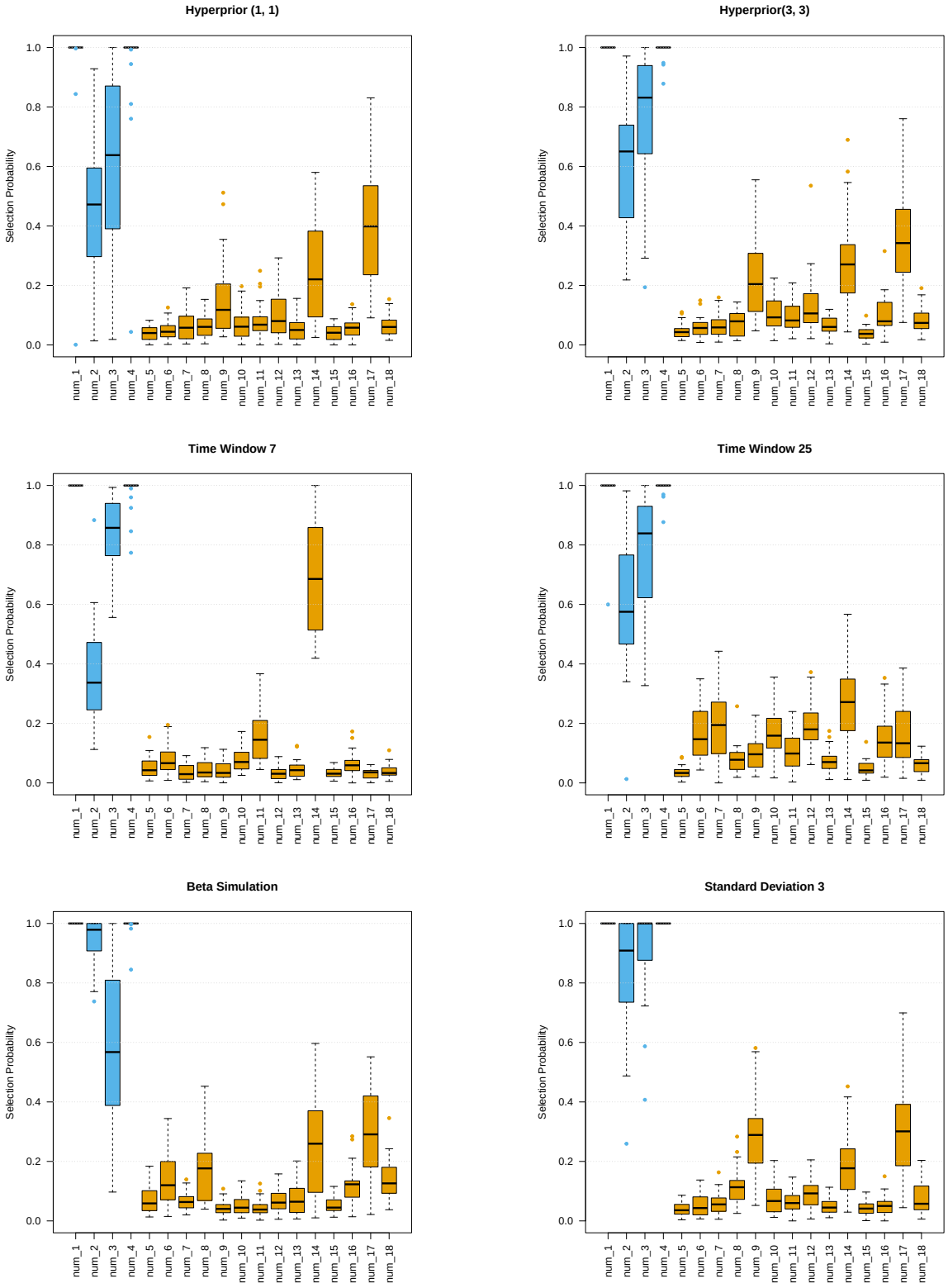


Figure 4.6: Variable inclusion for all the simulated datasets across 30 runs each for **imbalanced simulations** with **high collinearity**.

Figures 4.3 and 4.4, that show the results of the low collinearity simulations, are all generally quite good regardless of the experiment. The true predictive variables are almost always included across all the runs of all the experiments, and the non-predictive variables are generally close to the 0 boundary, though not quite as tightly. In almost all cases the algorithm more effectively filtered out non-predictive data when the response distribution was balanced, in the sense that the posterior samples are closer to 0 on average. The only exception being the simulation that used the Beta distribution as the simulation DLF, where one of the incorrect variables has a higher than usual acceptance rate. In contrast, when the data was imbalanced, one of the correct variables was selected less often than normal. However in both cases, the results are still largely positive.

The high collinearity experiments, seen in figures 4.5 and 4.6 have more mixed performance. Under these circumstances, the model seems to struggle a lot more at identifying the correct covariate set, which is to be expected when the predictor time series are so statistically similar to the non-predictors. Regardless, the results are still mostly good; for the balanced experiments, the true predictors are chosen on average far more often than the false predictors. An interesting contrast in the balanced, high collinearity experiments is between the Hyperprior(1, 1) experiment and Hyperprior(3, 3) experiment. For Hyperprior(1, 1), the weaker predictors are filtered out much more often than they should, but the non-predictors are even closer to the boundary than for every other experiment. The reason for this is intuitive; as more variables are rejected, the prior probability of inclusion will on average fall closer to the 0 boundary due to the Beta hyperprior shifting the bulk of its distribution closer to 0. This will occur at a faster rate for the (1, 1) setting than for (3, 3), as seen in figure 4.1. This suggests that if sparsity is more desirable than identifying the correct covariate set then a Beta(1, 1) hyperprior might be a good choice, but otherwise should be avoided.

Again we see that making the time window too small has negative impact on performance, especially for the high collinearity imbalanced version of the experiment. One of the correct predictors is selected less than 50% of the time on average and one of the false predictors is selected far more often than it should. Whereas the larger time window of length 25 shows decent performance throughout.

Experiment	Training AUC (%)	Test AUC (%)
Hyperprior (1, 1)	81	83
Hyperprior(3, 3)	80	83
Time Window 7	80	81
Time Window 25	80	82
Beta Simulation	81	78
Standard Deviation 3	79	79

Table 4.1: Mean AUC values for the **balanced simulations** with **low collinearity**.

Experiment	Training AUC (%)	Test AUC (%)
Hyperprior (1, 1)	80	79
Hyperprior(3, 3)	80	79
Time Window 7	79	79
Time Window 25	79	79
Beta Simulation	78	82
Standard Deviation 3	79	75

Table 4.2: Mean AUC values for the **imbalanced simulations** with **low collinearity**.

Higher covariate variability tends to perform better than average at rightly selecting the correct predictors, but also picks up more false predictors than on average. In this sense, the increased variation both helps and hinders; more variability means the model can more easily pick up on informative predictors but also allows more opportunity for noise to be picked up as a signal.

The AUCs across all runs are given in tables 4.1-4.4. We see a high level of agreement between the training set and test set AUCs (in some cases the test sets slightly outperform training), suggesting that the model is not overfitting, regardless of the setting. The AUCs are generally very high for the low multicollinearity experiments, sitting in the low 80's / high 70's, but drop roughly 5-10% for the high multicollinearity experiments.

We conclude from these experiments that the method works very well for balanced

Experiment	Training AUC (%)	Test AUC (%)
Hyperprior (1, 1)	67	70
Hyperprior(3, 3)	70	73
Time Window 7	68	70
Time Window 25	70	73
Beta Simulation	74	78
Standard Deviation 3	75	78

Table 4.3: Mean AUC values for the **balanced simulations** with **high collinearity**.

Experiment	Training AUC (%)	Test AUC (%)
Hyperprior (1, 1)	72	66
Hyperprior(3, 3)	73	66
Time Window 7	71	62
Time Window 25	74	66
Beta Simulation	75	68
Standard Deviation 3	75	71

Table 4.4: Mean AUC values for the **imbalanced simulations** with **high collinearity**.

and imbalanced data when the data is mostly independent. Results are more mixed in the presence of high collinearity, especially for imbalanced data, but still largely promising.

4.6 Flare Data Analysis

We now return to the research problem motivating the development of the above methods; to determine if there are any measures of air quality that are predictive of ANCA vasculitis flare propensity.

4.6.1 Data

The data is stitched together from two sources; the first we will speak about are clinical records from the AVERT Resource Description Framework (RDF) database. The patients in this dataset are members of Trinity College's AVERT project; people suffering from vasculitis who have consented to sharing their electronic clinical records for the purposes of vasculitis research. Among other things, the AVERT database contains data on hospital visits. Whether or not the patient was suffering a relapse is a point of contention; due to the lack of specific symptoms early on in disease progression (Karangizi and Harper, 2018) the attending clinician can only make an informed guess (this is logged in the clinical records). Thankfully expert nephrologists in the AVERT group were able to retroactively investigate patient records to surmise a more accurate impression of whether or not the patient was truly experiencing a flare at the time. Their degree of belief that a flare event had occurred was split into four different categories:

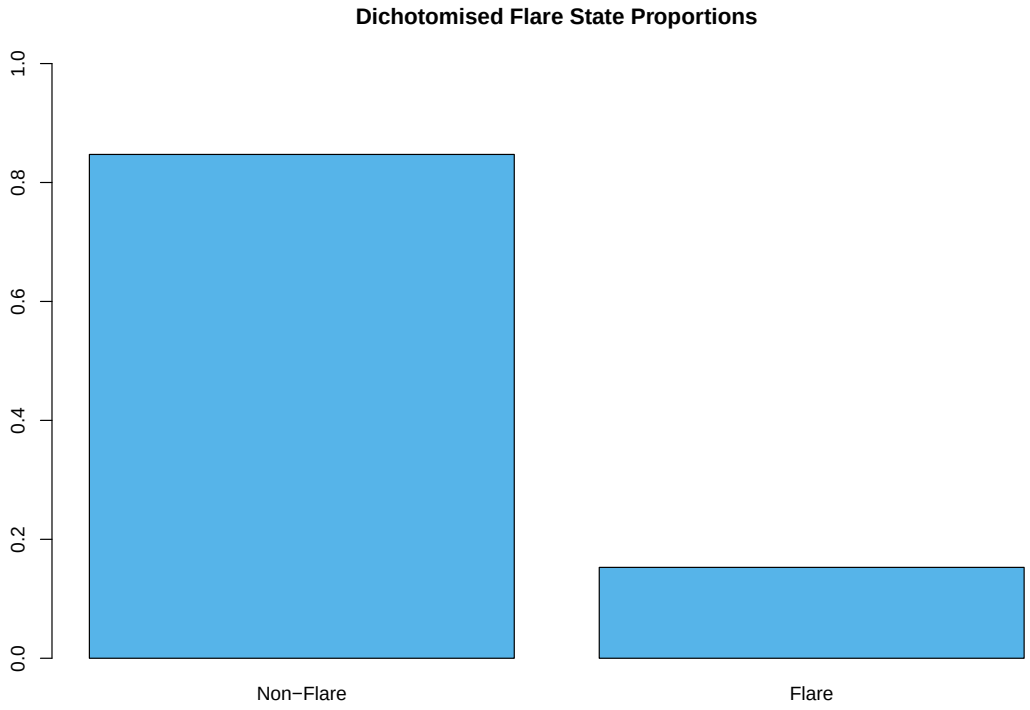


Figure 4.7: Distribution of flare states in our data. Note the comparatively much larger proportion of non-flares.

- No possibility,
- possible but not probable,
- probable,
- definite.

We refer to this as the *adjudicated probability* of flare. For the purposes of our analysis (and at the recommendation of the experts) our response variable is a dichotomised version of the adjudicated probability; 0 for No possibility, 1 otherwise. The distribution of flares/non-flares is given in figure 4.7, where we can see a large degree of imbalance.

One issue here is that the dates of these hospital visits are not truly relevant; what we really need is the date that the corresponding flare began, and it is very unlikely that flares manifested the very day of the hospital visits. Through a focus group with patients, we were able to directly ask them how far apart they usually felt was the time between first experiencing symptoms of a flare and when they actually made the

hospital visit; most responses put the number around 7 to 14 days. Based on this, we backdate the hospital visit dates by 10 days to approximate the date of the actual event. We will test the sensitivity of this decision in appendix B.

We also have information on each patient's location which is used to link to the the appropriate region for the environmental data, where the 'regions' are roughly split by the counties of Ireland. For a small number of patients, we have smartphone location (Beukenhorst et al., 2017) provided with their permission using the PatientMPower app (homepage: `https://info.patientmpower.com/`), an app that helps patients of chronic diseases manage their illness. For patients who don't have app data we use their home address instead if available. If neither of those are available we use the address of their hospital. A clear limitation of this is that non-smartphone location data might be misleading in characterising the patient's actual region of exposure. For example a person who lives in Wicklow but works in Dublin will be exposed to much more different air quality than someone who lives *and* works in Wicklow. Unfortunately, due to a low level of engagement with the app, the majority of the data is based on either home or hospital location.

Our weather and pollution variables are retrieved from the AVERT RDF database, originally sourced from the Copernicus Atmosphere Monitoring Service, or CAMS (homepage `https://atmosphere.copernicus.eu/`) (Inness et al., 2019; Hersbach et al., 2020). The SPARQL queries necessary to extract the correct spatio-temporal snapshots for each patient were constructed via the use of a simplified GUI built by the AVERT group. The data is mean aggregated so that it falls on a daily time scale.

The only patients we included in the study are those who were determined to have at least one flare event between 11th of April 2016 and 29th of February 2020, so that we have almost 4 years of daily data. The reason for starting on the 11th of April is that the AVERT RDF has pollution data beginning the 1st March 2016, and we need at least a month extra to accommodate the fact that we will use a time window of length 30 days, and another 10 days to backshift the hospital visit date to approximate the flare date. The final date is 29th of February 2020 to avoid the study period overlapping with the COVID–19 pandemic as this would introduce a level of confounding that our

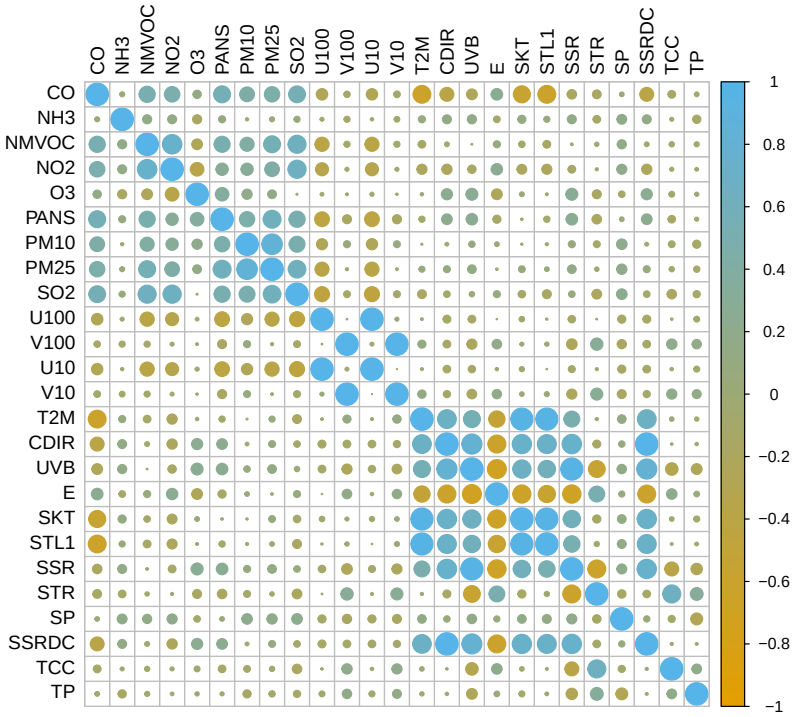


Figure 4.8: Correlation matrix of environmental dataset.

model is not equipped to handle.

There were some missing values in the data. Most notably, every pollution variable was missing on the 30th of November 2016. We resolved this by linearly interpolating the gap. Three patients only had associated pollution or weather data up until mid-2017, and so had to be removed from the study. A weather variable, Sea Surface Temperature, was missing a large proportion of its values so it was also discarded. After this, there were just a small number of missing values remaining; the variable with the next highest number of missing values was carbon monoxide, and it was only missing twelve (non-consecutive) days out of the 4 year period. All of these remaining missing values were linearly interpolated.

Another concern is the high degree of multicollinearity in the dataset as illustrated in figure 4.8. We removed the following variables so that there would not be more than 80% collinearity in the dataset: U10, V10, T2M, SSRDC, SSR, STL1 and PM10.

Our final dataset contains 87 patients, each with 4 years worth of daily data and 18 location specific environmental explanatory variables. Between the 87 patients, there

were 746 clinical visits, 114 ($\approx 15\%$) of which were retroactively determined to coincide with flare events.

4.6.2 Model Settings and Results

We used a two-degree Almon function as the DLF with a time window of 30 days. The quantile was set to 1 minus the proportion of flares in our data, roughly 0.85. The prior on $\boldsymbol{\beta}$ was Gaussian with mean vector $\mathbf{0}$ and covariance matrix with only diagonal entries equal to 100. As explained in section 4.1.2, the prior on the first component of $\boldsymbol{\theta}$ was Gaussian with mean 15 and standard deviation 5, and the prior of the second component was exponential with rate parameter 1. The hyperprior of the $\boldsymbol{\gamma}$ prior probability of inclusion p was set to a Beta(3, 3) distribution.

To mitigate the risk of initialisation sensitivity, we ran the model four times with different starting values for $\boldsymbol{\gamma}$:

- the intercept only model,
- the saturated model,
- two models where each variable was randomly included with probability 0.5.

In the intercept only model, the starting value for the intercept was 0, and in all other cases the starting value for $\boldsymbol{\beta}$ were randomly sampled from a standard Gaussian. As for the DLF parameters, the first component started at zero, the second started at -1 in all cases. Each run lasted for a million overall iterations. The first half of these iterations were discarded and only every 10th iteration thereafter was retained, for a total of 50,000 approximately independent draws from the posterior.

Each model was fit in parallel using 4 cores on a Dell Latitude 5400 laptop. The total time to complete was over 8 hours.

Results

The posterior of $\boldsymbol{\gamma}$ across all runs are given in Figure 4.9, revealing a very low rate of variable inclusion across all runs (note the x-axis only extends as far as 6%). It's important to emphasise here that only one of the four runs was initialised in this region of the posterior, yet they all converged to it.

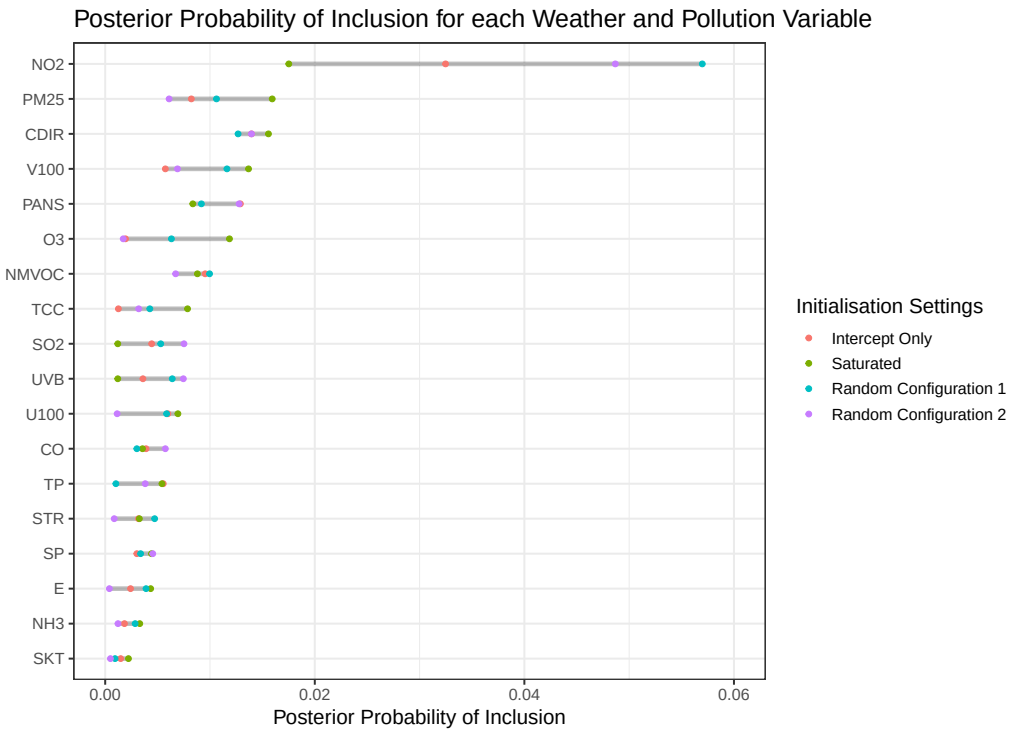


Figure 4.9: Variable selection results after 4 runs with different starting points. Take notice of the scale of the x-axis; across all runs, no variable is selected more than 6% of the time.

The variable selected most often was Nitrogen Dioxide (N02). Across all runs, it was selected more often than the next most highly selected variable, PM25. No other variable ever achieved more than 2% inclusion across all runs, with most never going even above 1%. In appendix B we show the results of re-running this analysis using two other possible backshift values to approximate the flare onset date. One uses no backshift at all, and just assume that the clinic visit dates are the onset dates. The other backshifts the data by 20 days (as a reminder, the analysis above uses 10 days). The resulting selection rates are even lower in both cases; none of the covariates have a higher than 3% inclusion rate. Interestingly N02 remains the most commonly selected variable, though not by as high of a margin.

4.6.3 Concluding Remarks

Given the results above, we conclude that the environmental variables that were under consideration were not predictive of the flare events. Nitrogen Dioxide showed the most potential judging by the fact that it was consistently the most selected variable here and in appendix B. Despite this, its overall acceptance was very low, so we consider it

very unlikely that it is a true predictor of flare onset. This does not mean that we have proven that adverse environmental exposure has no effect on flare propensity, only that a signal could not be found in this specific set of data.

To put this finding into proper context, we reiterate some of the problems with the data. To start with, the given clinical visit date does not tell us the exact time of flare occurrence. The environmental data is based on heavily processed satellite data. Ground based measurements from monitoring stations would theoretically produce more accurate data, but there are not enough of those in Ireland to get good data coverage. Even still, it may be the case that the satellite data is not granular enough to adequately capture location-based variability, especially around large and environmentally diverse counties like Dublin. There is also a lot of uncertainty around the patients' locations - for most of them we only have their home address and we do not know how often they travel away from home or how far, masking their true air quality exposure profile.

There's also limitations with the model. The method requires a lot of tuning via the prior distributions, and unfortunately we know of no experts we can consult for elicitation on this specific research problem. While we have tried our best to use sensible priors, they are ultimately a matter of our own personal judgement, and we are not experts on this matter. Besides that, given more time, we would have liked to include random effect parameters to possibly account for patient/spatial heterogeneity, and perhaps explore other options of sampling from the full-conditionals of the DLF parameters. We go into more details about these ideas in chapter 6.

That is not to say that we think our model 'failed' in any sense. We believe we have developed a very useful and solid inferential foundation to build upon. Bear in mind many of the issues discussed above may be resolved in a matter of time; as research in this area grows, future models will presumably be able to set more informed priors. Over time, we would also expect satellite reanalysis to get more accurate, and perhaps at some point Ireland will have more facilities to collect more accurate environmental data. In the meantime we can continue building on the work we have, and perhaps apply it to other problems where more reliable data is already available.

5 Estimation of Lockdown Effect During COVID–19 Pandemic using Age-Structured SEIR Model

5.1 Introduction

In section 2.5 we spoke about compartmental models. In this chapter we propose and calibrate a flexible compartmental model within the Susceptible Exposed Infected Recovered (SEIR) class, in order to quantify the effect of different forms of non-pharmaceutical interventions (NPIs) on the spread of Coronavirus. The model is age structured to account for the differences in social mixing behaviours and risk profiles. The social mixing component of the analysis is modelled using age group to age group contact rates, allowing for assessment of the long run impact brought about by lockdowns which implicitly target specific age groups.

Self-plagiarism and contribution disclosure: this work was part of a joint research project with the University of Limerick, resulting in a published manuscript (Jaouimaa et al., 2021), authored by a number of people besides myself, that this chapter borrows heavily from. In particular I want to acknowledge that Fatima-Zahra Jaouimaa contributed as much to the published article as I did. To make clear my contribution of what's shown in this chapter, I was the primary analyst involved in doing the literature review for table 5.2, doing the calculations to derive the next-generation matrix in section 5.3, and developing the `Shiny` app discussed in section 5.6. The model fit and bootstrapping analysis was primarily implemented by Fatima, though I offered support there also. Otherwise the text, graphs and figures presented here were an equal

effort among all authors. Note the published manuscript includes more information, such as an economic costing analysis of NPIs and a sensitivity analysis of the model fit under differing contact matrices, but I have omitted them here as I personally had little involvement with those aspects of the project.

5.1.1 Background

NPIs such as lockdowns (or restriction of movement) were vital in managing the spread of COVID–19 before the development and rollout of the vaccines. However NPIs have undoubtedly left a harsh mark on society as the societal and economic impacts have become clear. While older individuals are observed to be gravely threatened by the risk of infection, younger people have been particularly impacted by deteriorating mental health during this time (Kwong et al., 2020) in addition to reduced economic prospects (Darmody et al., 2020) resulting in a difficult balancing for policy-makers. Strict lockdown measures are necessary for public safety and to prevent health systems from becoming overwhelmed. However, periods of strict measures need to be punctuated by temporary easing of restrictions whenever possible to ease the impact to the population's mental and economic health.

National 'maps' and 'road-plans' for emerging from COVID–19 that were proposed in the first quarter of 2020 by national Governments have been tweaked and revised world-wide; the time elapsed since March 2020 has been characterised by an ebb and flow of various forms of restriction of movement, both within and between nations. Certain measures or guidelines to citizens may target specific age groups. For example, guidance has often urged extra protection for the elderly. There has been much debate about the risks posed by keeping schools for children open and as a result there has been variation in school closures globally, making it essential to quantify the potential impact of new or changing measures that target age cohorts differently.

In chapter 2 we mentioned many examples of how SEIR models were used in COVID research. The dynamics of the age-structured SEIR model we propose here have a number of advantages over these competing approaches. We account for the impact of movement restrictions on population mixing by scaling age-structured contact matrices, as with Prem et al. (2020), however our scaling parameters are calibrated using

the time series of observed Irish incidence counts as opposed to best-guess estimates. Where the parameters governing dynamics of models cannot be estimated due to data sparsity, we use results published in the COVID–19 literature on infection dynamics as well as expert opinion from the Irish Epidemiological Modelling Advisory Group (IEMAG); note that IEMAG developed an initial SEIR model (IEMAG, 2020; Gleeson et al., 2022) that we extend through the introduction of age-structuring and incorporation of the contact patterns, thus relaxing the assumption of homogeneous mixing across population age groups. Otherwise, this assumption would imply that the force of infection is the same for all ages and may lead to the misrepresentation of disease dynamics for populations with heterogeneous population mixing and non-random contact patterns as a result. The force of infection in our extended model reflects the age-related degree of mixing both within and among different age-groups which is a more realistic transmission hypothesis. We use a parametric bootstrap to estimate uncertainties in learned parameters, in addition to providing uncertainty intervals for incidence projections.

5.2 Data

The available data for calibration of social contacts consists of daily case counts and the specific lockdown measures implemented within Ireland from February 29th 2020 to January 31st 2021. While we present an analysis specific to an Irish context, we believe the proposed approach is adaptable to other locales, wherein region specific macro-level behaviours can be calibrated. Furthermore, the framework we present can be easily adapted to incorporate more in depth population mixing knowledge from contact-tracing initiatives as well as allowing for estimation of all unknown model parameters.

We restrict our data sources to those that are typically freely publicly available, allowing for ease of implementation in other regions. The available data in an Irish context consists of daily incidence counts and the dates of changes of lockdown restrictions. Estimated contact matrices for age structured population mixing are sourced from literature. We defer discussion of the mechanistic parameters sourced from the COVID–19 literature to Section 5.3.

5.2.1 Daily incidence counts

The Irish Health Surveillance Protection Centre (https://data.gov.ie/) provide anonymised daily COVID–19 incidences. We use the data from the period of February 29th 2020 to January 31st 2021 for model calibration. The daily COVID–19 count incidence is shown in Fig. 5.1. Age structured case count data is not publicly available in Ireland, and hence we use aggregate case counts at the population level. We use projected population data for 2019 provided by Irish Central Statistics Office (CSO) (https://data.cso.ie/table/PEB07) to estimate the age-structured population breakdown by county to estimate Dublin's population. Given the constraints on public movement, we make the assumption that the 2019 projections are representative of the population since the beginning of the pandemic.

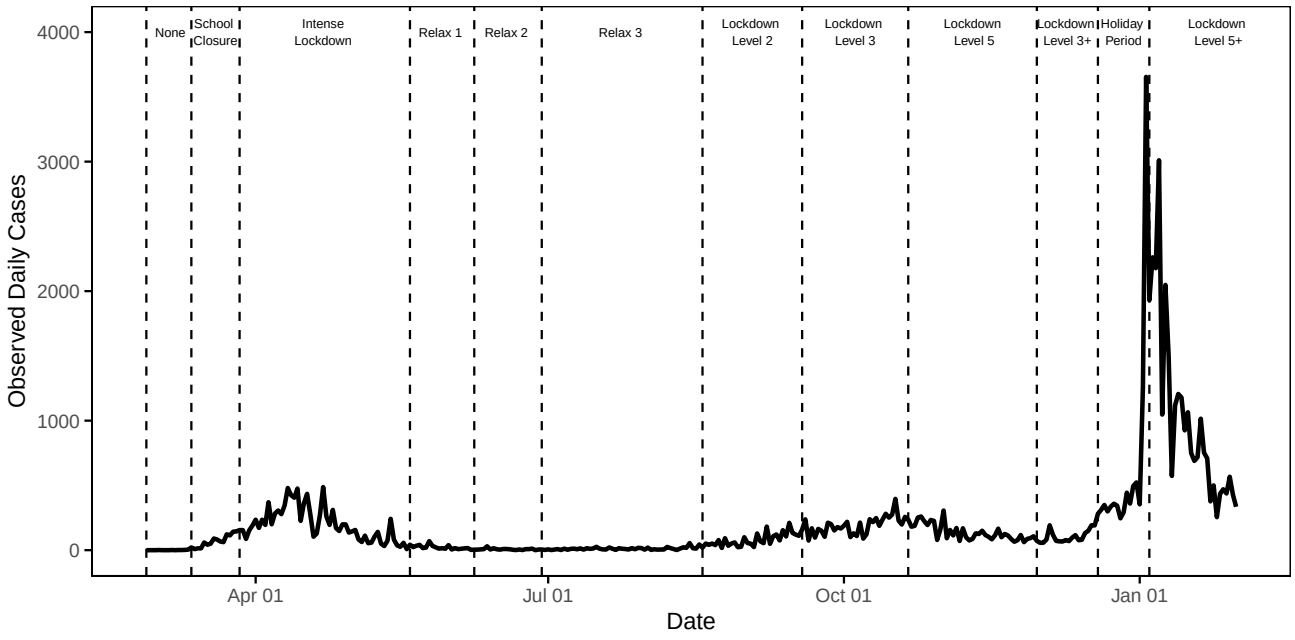


Figure 5.1: COVID–19 daily case incidence with corresponding lockdown levels in each period between February 2020 and January 2021. Descriptions of individual lockdown levels are presented in S1 Table.

5.2.2 Form of lockdown restrictions

Fig. 5.1 shows the timeline and duration of varying degrees of restriction measures (vertical dashed lines) implemented in Ireland from March 2020. In line with IEMAG (2020), we define the 28th February 2020 as "day zero" of the Irish epidemic. As with many other countries, the Irish government introduced a strict lockdown in the early

	Level 1	Level 2	Level 3	Level 4	Level 5
House visits	10 (3 households)	6 (2-3 households)	1 household	0	0
Gatherings	50 outdoor	6 indoor 15 outdoor	0	0	0
Weddings	100	50	25	6	6
Indoor events	100	50	0	0	0
Sporting events	100 indoor 200 outdoor	50 indoor 100 outdoor	0	0	0
Food venues	Open	6 (3 households)	15 outdoor	15 outdoor	0
Pubs	Open	6 (3 households)	15 outdoor	15 outdoor	0
Public transport capacity	100%	50%	50%	25%	25%

Table 5.1: Summary overview of the restrictions impacting on public gatherings for each of the five lockdown levels in Ireland. The numbers comprise the limits on individuals allowed to gather together in each social setting unless otherwise specified as a household limit. Details on other restrictions, such as on private travel, have been omitted for brevity.

stages of the pandemic which lasted until May 2020. This lockdown was followed by a gradual easing of restrictions throughout the summer until case numbers began to rise in early autumn, when harsher restrictions were reintroduced and another phase of a strict lockdown was announced for late October. Restrictions were eased over the month of December but a subsequent wave of cases forced the implementation of a further strict lockdown immediately after the December holiday period.

The nature of restrictions on public mobility in Ireland, announced by the Irish government in April 2020, follow five levels. Level one is the least restrictive with this increasing to most restrictive at level five. In level one, food venues and bars remain open, gatherings of up to fifty people are permitted outdoors and sporting events can take place with restrictions on numbers. Level five corresponds to a near total blanket close on all activities. As the public health situation evolved during 2020, small adjustments were made to these levels with slight easing of targeted restrictions (for example, reopening of schools or childcare) within more severe lockdowns. An overview of the five level lockdowns is provided in Table 5.1 with in-depth detail available at gov.ie/en/campaigns/resilience-recovery-2020-2021-plan-for-living-with-covid-19

We denote the time intervals of lockdown measures using $\mathcal{I}_k = (r_k, r_{k+1}]$, where r_k is the time of the beginning of the kth regime for $k = 1, \ldots, N$ where $N = 12$, with the first regime corresponding to no intervention from 29th February to 11th March 2020. Thus, we define $r_1 = 0$ corresponding to 29th February 2020 and $r_{N+1} = 336$ corresponding to 31th January 2021.

5.2.3 Age structuring and social mixing

COVID–19 is an airborne virus, hence consideration of close social mixing in the population is essential to capturing the observed patterns of infection. Furthermore, the strong association of morbidity patterns with the elderly, and more recently younger persons (Taylor, 2021), suggest consideration of age structured social mixing will be a key component of future projections (Prem et al., 2020; Cuevas-Maraver et al., 2021). Age-structured social mixing is typically captured in SEIR models through the use of age group to age group contact matrices. Although such matrices cannot capture the granular complexity of individual human interactions, they provide a reasonable approximation that can be incorporated into mathematical models for infectious diseases as demonstrated by Mossong et al. (2008), and within this article. We follow Prem et al. (2020) and stratify the population into five-year bands from age 0 up to age 75, with one category for all individuals aged 75 and above, giving $A = 16$ age groups.

Contact tracing has been a prominent factor in disease suppression in Asian countries to date. However, such data is not available in an Irish context, and we are unaware of any large-scale survey or study on age-structured social mixing patterns in Ireland. However, Fumanelli et al. (2012) and Prem et al. (2017) provide a methodology for deriving contact patterns by leveraging mixing patterns studied in other European countries. Our analysis relies on contact matrices given by Prem et al. (2017) who projected age and location specific contact matrices in 16 age bands for 152 countries including Ireland. These are constructed from the POLYMOD study (Mossong et al., 2008), which incorporated large-scale demographic household surveys (from the UN population division) and school and labour force participation rates. The estimated Irish contact interactions are shown in Fig. 5.2. For the purpose of our work, we sum together expected contacts in the home, work, school and other locations to give an overall matrix of expected contacts ('All' in Fig. 5.2). We assume that the Irish contact

matrix applies to just Dublin.

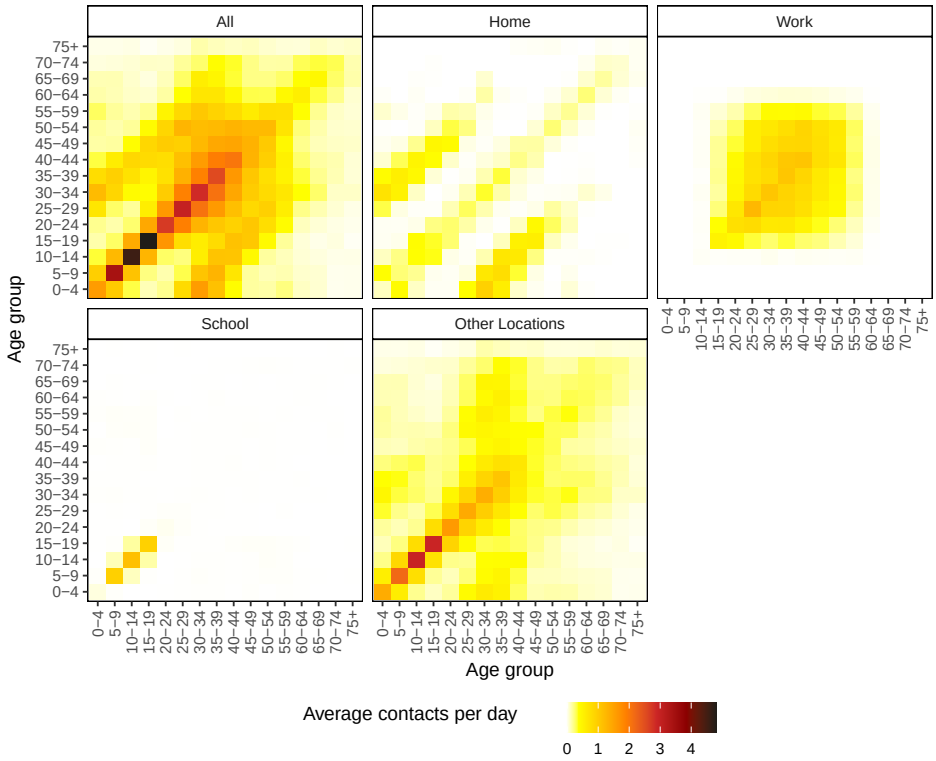


Figure 5.2: Estimated social contact matrices for Irish population mixing at 5 year intervals (Prem et al., 2017)

Government interventions to suppress virus spread result in changes in population mixing. Therefore, to reflect these changes, we introduce a free parameter, θ_k, which scales the aggregate expected contact matrix for the time interval $\mathcal{I}_k = (r_k, r_{k+1}]$ corresponding to the kth of N lockdown regimes. The aforementioned age-structured contact matrices are formed by entries $c_{i\,j}$ representing the average number of daily contacts between an individual in age category i with an individual in age category j, where $i, j = 1, \ldots, A$. Then, at time $t \in \mathcal{I}_k$ (i.e., during the kth lockdown), the scaled contact matrix is

$$\begin{pmatrix} \theta_k c_{1\,1} & \ldots & \theta_k c_{1\,A} \\ \vdots & \ldots & \vdots \\ \theta_k c_{A\,1} & \ldots & \theta_k c_{A\,A} \end{pmatrix} = \theta_k \mathbf{C}, \qquad k = 1, \ldots, N.$$

In Section 5.3 we outline the estimation of the scaling parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_N)$ for each lockdown period using observed incidences in Dublin. This allows an estimation of macro-level behavioural changes in socialising brought about by specific

measures.

5.3 SEIR model specification

Irish population modelling of COVID–19 during the crisis has been carried out by the Irish Epidemiological Modelling and Advisory Service (IEMAG) (IEMAG, 2020). They present a model for the Irish population where, at any point in time, an individual is assumed to be in one of a number of distinct model compartments that describe COVID–19 status. Movement between compartments over time is based on the current understanding of the epidemiology of COVID–19, as evidenced by the extensive literature review and evidence synthesis conducted by Griffin et al. (2020); McAloon et al. (2020); Byrne et al. (2020).

We evolve this model to consider age-structured differences in population mixing. We assume closed age classes, such that population N_i of age class i is the sum of susceptible (S_i), exposed (E_i), infected and removed (R_i) compartments for that age class. There is no movement between age classes. Infected cases fall into a number of compartments: asymptomatic (I_i^{AS}), pre-symptomatic (I_i^{PS}), symptomatic and self-isolating without testing (I_i^{SI}), symptomatic and awaiting test results (I_i^{ST}), symptomatic and isolating after receiving positive test results (I_i^{PI}) and symptomatic but not tested or isolating (I_i^{SN}). The closed age class assumption implies that

$$N_i = S_i + E_i + I_i^{\mathrm{AS}} + I_i^{\mathrm{PS}} + I_i^{\mathrm{SI}} + I_i^{\mathrm{ST}} + I_i^{\mathrm{PI}} + I_i^{\mathrm{SN}} + R_i, \qquad i = 1, \ldots, A.$$

Exposed individuals are those incubating the disease but not yet infectious. Asymptomatic individuals are infectious but do not exhibit symptoms. Pre-symptomatic individuals are infectious but have yet to show symptoms. As pre-symptomatic individuals' symptoms develop, they will move to one of the infectious or symptomatic compartments, either self isolating and following government guidance around testing, or neither getting tested nor isolating when symptomatic, i.e., ignoring symptoms. Following infection, individuals move to the removed class (R_i), which accounts for cases who recover and those who die.

We write the system of ordinary differential equations (ODEs) describing the SEIR

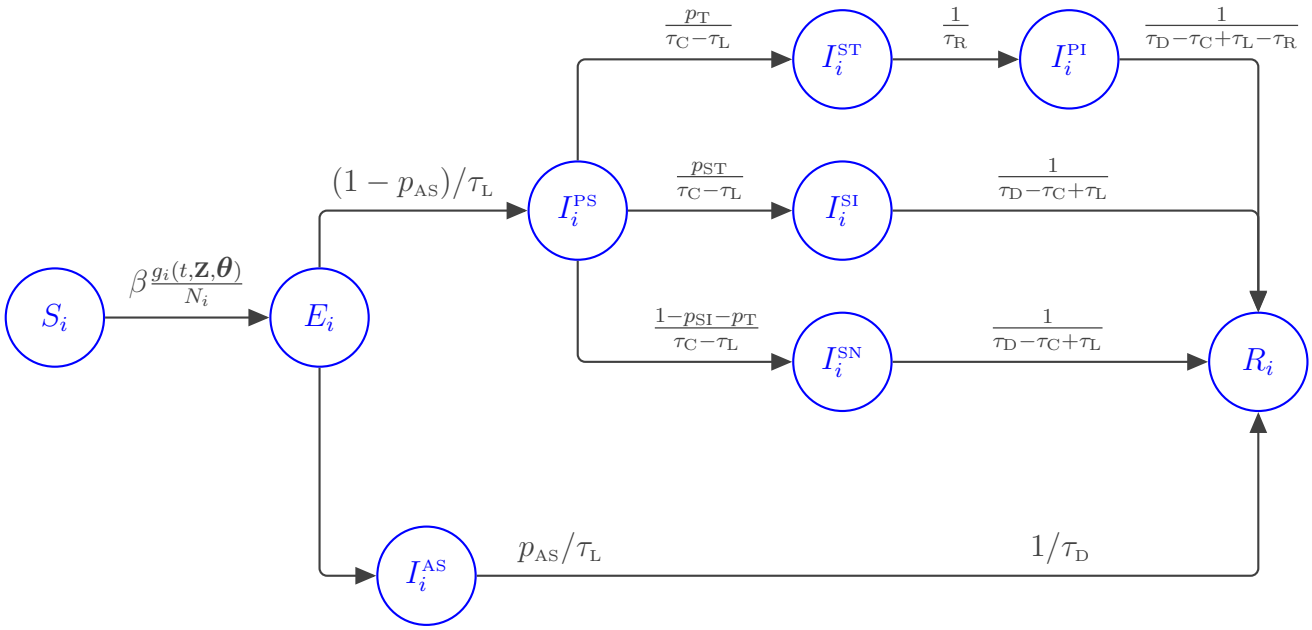


Figure 5.3: Diagram representing interactions in the age-structured SEIR system of ODEs with rate of movement between classes indicated. The compartments are susceptible (S), exposed (E), infected and removed (R), asymptomatic (I^{AS}), pre-symptomatic (I^{PS}), symptomatic and self-isolating without testing (I^{SI}), symptomatic and awaiting test results (I^{ST}), symptomatic and isolating after receiving positive test results (I^{PI}) and symptomatic but not tested or isolating (I^{SN}). A full description is given in Eq (5.1).

model for age class $i = 1, \ldots, A$:

$$\frac{\mathrm{d}S_i}{\mathrm{d}\,t} = -\beta\,\frac{g_i(t,\mathbf{z},\boldsymbol{\theta})}{N_i}\,S_i \qquad \frac{\mathrm{d}E_i}{\mathrm{d}\,t} = \beta\,\frac{g_i(t,\mathbf{z},\boldsymbol{\theta})}{N_i}\,S_i - \frac{E_i}{\tau_{\mathrm{L}}}$$

$$\frac{\mathrm{d}I_i^{\mathrm{AS}}}{\mathrm{d}\,t} = p_{\mathrm{AS}}\,\frac{E_i}{\tau_{\mathrm{L}}} - \frac{I_i^{\mathrm{AS}}}{\tau_{\mathrm{D}}} \qquad \frac{\mathrm{d}I_i^{\mathrm{PS}}}{\mathrm{d}\,t} = (1-p_{\mathrm{AS}})\,\frac{E_i}{\tau_{\mathrm{L}}} - \frac{I_i^{\mathrm{PS}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}}$$

$$\frac{\mathrm{d}I_i^{\mathrm{SI}}}{\mathrm{d}\,t} = p_{\mathrm{SI}}\,\frac{I_i^{\mathrm{PS}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} - \frac{I_i^{\mathrm{SI}}}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}} \qquad \frac{\mathrm{d}I_i^{\mathrm{ST}}}{\mathrm{d}\,t} = p_{\mathrm{T}}\,\frac{I_i^{\mathrm{PS}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} - \frac{I_i^{\mathrm{ST}}}{\tau_{\mathrm{R}}}$$

$$\frac{\mathrm{d}I_i^{\mathrm{SN}}}{\mathrm{d}\,t} = (1-p_{\mathrm{SI}}-p_{\mathrm{T}})\,\frac{I_i^{\mathrm{PS}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} - \frac{I_i^{\mathrm{SN}}}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}} \qquad \frac{\mathrm{d}I_i^{\mathrm{PI}}}{\mathrm{d}\,t} = \frac{I_i^{\mathrm{ST}}}{\tau_{\mathrm{R}}} - \frac{I_i^{\mathrm{PI}}}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}-\tau_{\mathrm{R}}}$$

$$\frac{\mathrm{d}R_i}{\mathrm{d}\,t} = \frac{I_i^{\mathrm{AS}}}{\tau_{\mathrm{D}}} + \frac{I_i^{\mathrm{SI}}}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}} + \frac{I_i^{\mathrm{PI}}}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}-\tau_{\mathrm{R}}} + \frac{I_i^{\mathrm{SN}}}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}} \tag{5.1}$$

where the function $g_i(t, \mathbf{z}, \boldsymbol{\theta})$ in the mass relation for being exposed when susceptible is

$$g_i(t, \mathbf{z}, \boldsymbol{\theta}) = \sum_{k=1}^{N} \mathbb{I}\,(\, t \in \mathcal{I}_k\,) \sum_{j=1}^{A} \theta_k\, c_{ij}\, \left[\, \alpha\, I_j^{\mathrm{AS}} + I_j^{\mathrm{PS}} + \kappa\, I_j^{\mathrm{SI}} + I_j^{\mathrm{ST}} + \kappa\, I_j^{\mathrm{PI}} + I_j^{\mathrm{SN}} \,\right]$$

where $\mathbf{z}$ denotes the entire state vector

$$\mathbf{z} = (\, S_1, E_1, \ldots, S_2, E_2, \ldots, S_A, E_A, \ldots, R_A\,),$$

and $\mathbb{I}\,(\, t \in \mathcal{I}_k\,)$ is an indicator function which equals one when $t \in \mathcal{I}_k$ and is zero otherwise. Susceptible individuals in age class i are exposed to the virus through contacts with infected individuals in all age classes and this is described through the function $g_i(t, \mathbf{z}, \boldsymbol{\theta})$. The level of exposure is modulated by the scaled average number of daily contacts with each age class, with a scaling factor for each lockdown regime. The list of parameter value settings used in our model is given in Table 5.2. The value of β we use is based on R_0, described in the following section. $\theta_k, k = 1, \ldots, N$ are treated as unknown and are estimated using observed case indices. This will be discussed further in Section 5.4.

Next Generation Matrix and Derivation of the Force of Infection Parameter

One of the required SEIR model parameters is the multiplicative *force of infection* β. The value of β is chosen based on a specified value of R_0, the baseline reproduction rate. The word 'baseline' here highlights that this is the expected reproduction rate in Dublin assuming a fully susceptible population (i.e., $S_i = N_i$, $i = 1, \ldots, A$), under no intervention. R_0 can be expressed as the largest absolute eigenvalue of the *next generation matrix*, a matrix that encodes the spread of the disease whose form is determined by the model. Diekmann et al. (1990) showed that (subject to light conditions) its dominant eigenvalue of the *next-generation operator* can be interpreted as "the typical number of secondary cases", or R_0. For *discrete* state models such as ours, Section 2.2 of Heffernan et al. (2005) provides a practical explanation of how to construct the next generation matrix.

Parameter	Description	Value	Reference
τ_{C}	Average incubation period	5.8	McAloon et al. (2020)
τ_{P}	Average pre-symptomatic period	2	Byrne et al. (2020)
τ_{L}	Average latent period. This is computed as τ_{C} minus τ_{P}	3.8	
τ_{D}^{C}	Average infectious period for symptomatic patients	13.4	Byrne et al. (2020)
τ_{D}^{SC}	Average infectious period for asymptomatic patients	6	Byrne et al. (2020)
τ_{D}	Average infectious period. Weighted average of the symptomatic and asymptomatic periods (weighted by prevalence)	13.5	
R_0	Basic reproductive number	3.4	Náraigh and Byrne (2020)
α	Factor reduction of transmission from asymptomatic cases	0.55	Evoy et al. (2020) (The mean of given interv
κ	Factor reduction of transmission from isolating cases	0.05	IEMAG (2020)
p_{AS}	Proportion of asymptomatic infections	0.20	Buitrago-Garcia et al. (2020)
p_{T}	Proportion symptomatic who get tested	0.8	IEMAG (2020)
p_{SI}	Proportion symptomatic who self-isolate	0.1	IEMAG (2020)
τ_{R}	Expected time between first symptoms and test result	7	IEMAG (2020)

Table 5.2: List of parameters used directly in or sourced to for the specification of the SEIR model.

Let $\tilde{\mathbf{z}} = (\tilde{z}_1, \ldots, \tilde{z}_p)$ be the vector of compartment sizes for compartments from which *infected* individuals either enter or leave. In our case, this is *all* the compartments except for susceptible and removed since leaving the susceptible compartment means you have only been *exposed* (not yet infected), and entering the removed compartment implies you are no longer infected. Now introduce $f_i(\tilde{\mathbf{z}})$ as the rate of *new* infections that enter compartment i, let $v_i^+(\tilde{\mathbf{z}})$ be the rate of individuals arriving into compartment i who are *not* newly infected, let $v_i^-(\tilde{\mathbf{z}})$ be the rate of individuals leaving compartment i, and finally let $v_i(\tilde{\mathbf{z}}) = v_i^-(\tilde{\mathbf{z}}) - v_i^+(\tilde{\mathbf{z}})$. With this notation, every system of equations described in (5.1) can be expressed as $f_i(\tilde{\mathbf{z}}) - v_i(\tilde{\mathbf{z}})$.

The next generation matrix is constructed from the matrices of partial derivatives of f_i and v_i,

$$F_{ij} = \frac{\partial f_i}{\partial \tilde{z}_j}(\tilde{\mathbf{z}}_0), \qquad V_{ij} = \frac{\partial v_i}{\partial \tilde{z}_j}(\tilde{\mathbf{z}}_0)$$

evaluated at the disease-free equilibrium, $\tilde{\mathbf{z}}_0$, i.e., the point at which no infection is present. In our application, since we are imposing a constant population N, the disease free equilibrium simply means that $S_i = N_i, i = 1, \ldots, A$ and all other compartments equal 0. The next generation matrix $\mathbf{Q}$ is equal to

$$\mathbf{Q} = \mathbf{F}\,\mathbf{V}^{-1}$$

In our application, the $\mathbf{F}$ matrix can be expressed as a block matrix, where each block corresponds to an age group

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & . & . & . & \mathbf{F}_{1A} \\ \mathbf{F}_{21} & \mathbf{F}_{22} & . & . & . & \mathbf{F}_{2A} \\ . & . & . & & & . \\ . & . & & . & & . \\ . & . & & & . & . \\ \mathbf{F}_{A1} & \mathbf{F}_{A2} & . & . & . & \mathbf{F}_{AA} \end{bmatrix}$$

where each block is given by

$$\mathbf{F}_{mn} = \begin{bmatrix} 0 & \alpha B_{mn} & B_{mn} & \kappa B_{mn} & B_{mn} & \kappa B_{mn} & B_{mn} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Note that only the first row of each block is non-zero and B_{mn} is defined as follows

$$B_{mn} = \beta c_{mn} \frac{N_m}{N_n}.$$

Beware that the subscripts m and n here correspond to the *block* indices, not its cell position in the matrix. Assuming there is no movement between the age groups the $\mathbf{V}$ matrix is expressed as a block diagonal matrix where each block corresponds to an age group.

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & 0 & . & . & . & 0 \\ 0 & \mathbf{V}_{22} & . & . & . & 0 \\ . & . & . & & & . \\ . & . & & . & & . \\ . & . & & & . & . \\ 0 & 0 & . & . & . & \mathbf{V}_{AA} \end{bmatrix}$$

Each block is given by

$$\mathbf{V}_{ii} = \begin{bmatrix} \frac{1}{\tau_{\mathrm{L}}} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{p_{\mathrm{AS}}}{\tau_{\mathrm{L}}} & \frac{1}{\tau_{\mathrm{D}}} & 0 & 0 & 0 & 0 & 0 \\ -\frac{1-p_{\mathrm{AS}}}{\tau_{\mathrm{L}}} & 0 & \frac{1}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{p_{\mathrm{SI}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} & \frac{1}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} & 0 & 0 & 0 \\ 0 & 0 & -\frac{p_{\mathrm{T}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} & 0 & \frac{1}{\tau_{\mathrm{R}}} & 0 & 0 \\ 0 & 0 & -\frac{1-p_{\mathrm{SI}}-p_{\mathrm{T}}}{\tau_{\mathrm{C}}-\tau_{\mathrm{L}}} & 0 & 0 & \frac{1}{\tau_{\mathrm{C}}-\tau_{\mathrm{D}}+\tau_{\mathrm{L}}} & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau_{\mathrm{R}}} & 0 & \frac{1}{\tau_{\mathrm{D}}-\tau_{\mathrm{C}}+\tau_{\mathrm{L}}-\tau_{\mathrm{R}}} \end{bmatrix}$$

Since each block represents an age group, different parameters for different age can be easily incorporated by simply altering the suitable block.

The largest absolute eigenvalue value of $\mathbf{Q}$ is R_0. Since β is easily factored out of the $\mathbf{F}$ matrix, the eigenvalue can be expressed as a product of β and the maximum eigenvalue of $\widehat{\mathbf{F}}\,\mathbf{V}^{-1}$, where $\mathbf{F} = \beta\,\widehat{\mathbf{F}}$. This means that if R_0 is determined and β is desired, the expression can easily be re-arranged:

$$\beta = R_0/\xi$$

where ξ is the maximum eigenvalue of $\widehat{\mathbf{F}}\,\mathbf{V}^{-1}$. The values of R_0 and β are only calculated once as they are baseline values; shifts in infection dynamics away from the baseline are captured by $\boldsymbol{\theta}$. Our choice for R_0 is 3.4, based on Náraigh and Byrne (2020). This leads to $\beta = 0.031$ when not under intervention.

5.4 Model Fitting

Specification of the model in Section 5.3 uses parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_N)$ to rescale, for each of the $N = 12$ lockdown intervention policies, what would have been the assumed average contacts between individuals in the various age classes under normal circumstances (i.e., prior to the pandemic). Estimation of these parameters is of interest in predicting behaviour during lockdown, and hence for forecasting the benefit of specific interventions. We first describe estimation of $\boldsymbol{\theta}$ and then a parametric bootstrap method (Pawitan, 2001) to provide uncertainty, which can be propagated

through model forecasts.

In order to link the model with observed data we monitor the cumulative number of cases up to time t for each age class. It is only cases exiting the $I_i^{\mathrm{ST}}, i = 1, \ldots, A$ compartment that can be linked to observed incidence counts in the general population. We can think of a variable, counting infected cases as they exit compartment I_i^{ST} before going into I_i^{PI}. For scaling $\boldsymbol{\theta}$, age class i and time t, denote this by $X_i(t; \boldsymbol{\theta})$. This can be related to the other model compartments through

$$\frac{\mathrm{d}X_i}{\mathrm{d}\,t} = \frac{I_i^{\mathrm{ST}}}{\tau_{\mathrm{R}}}, \qquad i = 1, \ldots, A. \tag{5.2}$$

To compare outputs from the SEIR model with observed data, we use this count aggregated over age classes:

$$X(t; \boldsymbol{\theta}) = \sum_{i=1}^{A} X_i(t; \boldsymbol{\theta}) \tag{5.3}$$

which gives total cumulative case counts to time t. Evaluating this at $t_d = hd$, $d = 1, \ldots, n$ where h generates a time discretisation corresponding to consecutive days, we can then compare $X(t_d; \boldsymbol{\theta})$ to observed cumulative cases at day d. We denote the observed cumulative counts at day d by x_d.

5.4.1 Estimation of regime specific contact scaling parameters

To estimate $\boldsymbol{\theta}$, we minimize the squared error loss, i.e., the residual sum of squares,

$$RSS(\boldsymbol{\theta}) = \sum_{d=1}^{n} \left(x_d - X(t_d; \boldsymbol{\theta})\right)^2$$

on cumulative case counts. Minimization is carried out using the default Nelder-Mead alogrithm (Nelder and Mead, 1965) provided in the `R` package `optimx` (Nash et al., 2011). Note that each step of this algorithm, corresponding to a proposed $\boldsymbol{\theta}$ vector, requires the calculation of $X(t; \boldsymbol{\theta})$ to evaluate the suitability of $\boldsymbol{\theta}$ through $RSS(\boldsymbol{\theta})$. In order to obtain $X(t; \boldsymbol{\theta})$, the system of ODEs given in (5.1) are numerically solved using the `R` package `deSolve` (Soetaert et al., 2010). Specifically, we have found the `lsoda` function within this package to be particularly flexible, providing automatic selection

of stiff or non-stiff methods; see Nash et al. (2011) for details. When solving the system of ODEs for a candidate $\boldsymbol{\theta}$, we take $I_i^{\text{PS}} = 1/A$, $E_i = 15/A$ and hence $S_i = N_i - 16/A$ as the initial values for $i = 1, \ldots, A$, as per IEMAG (2020). The initial values for the remaining compartments are set to 0. Multiple random initialisations of $\boldsymbol{\theta}$ are used to improve robustness of the overall algorithm with respect to the issue of convergence to local minima. When generating these initial vectors, we assume that the effect of lockdown measures is to reduce social mixing below pre-pandemic levels, and, therefore, use a $U(0,1)$ draw to initialise each parameter, i.e., $\boldsymbol{\theta}_k^0 \sim U(0,1), k = 1, \ldots, N$. A summary of our estimation procedure is given in Algorithm 1.

Algorithm 1 Estimation of $\boldsymbol{\theta}$

procedure ESTIMATION$(M, x_d, d = 1, \ldots, n)$
  for $m = 1, \ldots, M$ **do**
    $\boldsymbol{\theta}_k^{0(m)} \sim \text{Unif}(0,1), \qquad k = 1, \ldots, N$
    $RSS(\boldsymbol{\theta}) := \sum_{d=1}^{n} (x_d - X(t_d; \boldsymbol{\theta}))^2$ with $X(t_d; \boldsymbol{\theta})$ given by `lsoda`
    $\widehat{\boldsymbol{\theta}}^{(m)} = \text{argmin}_\theta \, RSS(\boldsymbol{\theta})$ using `optimx` initialised at $\boldsymbol{\theta}^{0(m)}$
    $RSS^{(m)} = RSS(\widehat{\boldsymbol{\theta}}^{(m)})$
  end forreturn $\widehat{\boldsymbol{\theta}} = \{\widehat{\boldsymbol{\theta}}^{(m)} \mid RSS^{(m)} = \min(RSS^{(1)}, \ldots, RSS^{(M)})\}$
end procedure

Since we have to solve the ODEs (5.1) numerically using `lsoda` of each iteration within the `optimx` optimisation, the above procedure is computationally intensive. On average it takes approximately 28 minutes to run the optimisation for each random initialisation on an Intel Core i5-8250U CPU with 4 cores. We use $C = 300$ random starts. To improve the computational feasibility we have each start was run in parallel on an EC2 instance hosted by Amazon Web Services with 32 cores and 64 GB memory.

5.4.2 Propagating uncertainty in contact scaling parameters

We explore uncertainty in the estimation of $\boldsymbol{\theta}$ and investigate how this propagates into the reproductive rate. In order to quantify uncertainty we follow Chowell (2017) by using a *parametric bootstrap* approach. The parametric bootstrap differs from its more commonly known non-parametric analogue (Efron, 1979) by making use of an assumed generative parametric model for daily case counts based on the observed case counts, instead of the sample themselves. This version is much more suitable for generating

time series replicates.

The model is used to re-generate B synthetic instances of the daily new case series; each of these instances is used to re-estimate the vector $\boldsymbol{\theta}$. The resulting empirical distribution of the re-estimated vectors can be used as an approximation to the sampling distribution of $\widehat{\boldsymbol{\theta}}$ (the estimate based on the original cumulative case counts).

The estimate $\widehat{\boldsymbol{\theta}}$ is found using the observed daily cumulative counts as described in Section 5.4.1. Given this estimate, the expected daily case count $\widehat{\mu}_d$ for day d can be predicted using

$$\widehat{\mu}_d = X(t_d; \widehat{\boldsymbol{\theta}}) - X(t_{d-1}; \widehat{\boldsymbol{\theta}}), \qquad d \geq 1$$

where $t_0 := 0$ and $X(0; \boldsymbol{\theta}) := 0$. We assume a negative binomial distribution (Chowell, 2017) as a generative model for daily case counts $Y_d \sim \text{NegBin}(\widehat{\mu}_d, \rho)$ with expected value $\widehat{\mu}_d$ and dispersion parameter ρ:

$$\Pr(Y_d = y) = \frac{\Gamma(\rho + y)}{y!\,\Gamma(\rho)} \left(\frac{\widehat{\mu}_d}{\widehat{\mu}_d + \rho}\right)^y \left(1 + \frac{\widehat{\mu}_d}{\rho}\right)^{-\rho} \tag{5.4}$$

where we have parameterised the negative binomial distribution through its expected value and dispersion. The value of ρ used for generating bootstrap Y_d series is the maximum likelihood estimate $\hat{\rho}$ based on the observed daily cases.

For each of $b = 1, \ldots, B$ bootstrap replications, we generate daily counts $y_d^{(b)}$ and convert to cumulative counts $x_d^{(b)}$, $d = 1, \ldots, n$. Then the method of Section 5.4.1 is applied to the $x_d^{(b)}$ to produce a bootstrap estimate $\widehat{\boldsymbol{\theta}}^{(b)}$. Collectively, the B estimates $\widehat{\boldsymbol{\theta}}^{(b)}$ provide an approximation to the sampling distribution of $\widehat{\boldsymbol{\theta}}$. The steps are summarized in Algorithm 2.

Algorithm 2 Parametric bootstrapping

procedure BOOTSTRAP($B, \widehat{\rho}, \widehat{\mu}_d, d = 1, \ldots, n$)
  for $b = 1, \ldots, B$ **do**
    $x_0^{(b)} := 0$
    for $d = 1, \ldots, n$ **do**
      $Y_d \sim \text{NegBin}(\widehat{\mu}_d, \widehat{\rho})$
      $x_d^{(b)} = x_{d-1}^{(b)} + y_d$
    end for
    $\widehat{\boldsymbol{\theta}}^{(b)}$ obtained from Algorithm 1
  end forreturn Bootstrap sample $\{\widehat{\boldsymbol{\theta}}^{(b)}, 1, \ldots, B\}$
end procedure

5.5 Results

In this section we present the results of fitting the age-structured SEIR model to Irish data using the methodology described in Section 5.4, and also discuss some economic findings.

5.5.1 Fitted SEIR model

Fig. 5.4(a) shows the model fit to the daily cumulative cases data (i.e., $X(t_d; \widehat{\boldsymbol{\theta}})$ and x_d respectively), while Fig. 5.4(b) shows a plot of the model fit to the daily new cases data (i.e., $X(t_d; \widehat{\boldsymbol{\theta}}) - X(t_{d-1}; \widehat{\boldsymbol{\theta}})$ and $x_d - x_{d-1}$ respectively). Both figures illustrate that the model provides a good fit to the observed data albeit with slight deviations in the early days of the epidemic and later around the December holiday and New Year period. We observe from the model fit to daily new cases that these periods were characterised by larger variability in the daily recorded number of new cases. The presence of outliers may be as a result of data reporting issues, especially over the December holiday period. For example, in Ireland a testing backlog developed over this period with test results from multiple days being subsequently batched together. An advantage of our age-structured model is the availability of a breakdown of cases by age-group and in fact all compartments. The mixing patterns for each age group determine its rate of infection which in turn determines the number of observed cases for each age group. Fig. 5.5 demonstrates the differences in model-based case numbers across selected age groups (see also S2 and S3 Figures). We can see that the case numbers are higher for the middle-aged group (35 – 39) than for the younger (5 – 9)

or older (65 – 69) groups, and note from Fig. 5.2 that the number of contacts is higher for the middle-aged group.

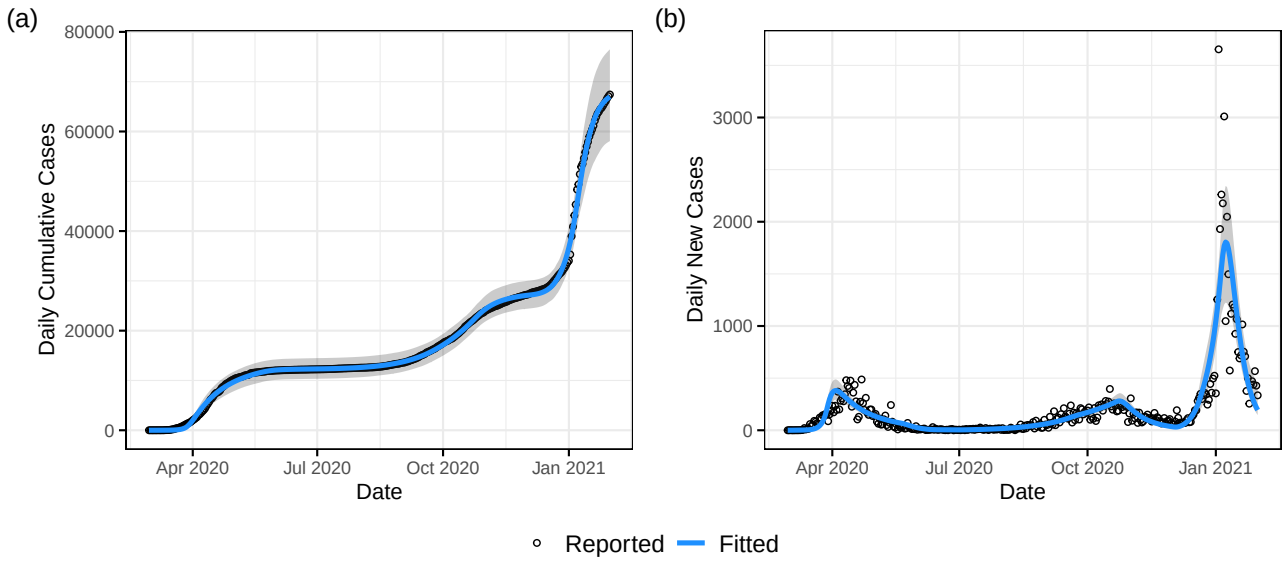


Figure 5.4: Model fit to daily recorded cases with bootstrapped 95% uncertainty bounds.

Fig. 5.6 displays the estimated scaling parameters $\boldsymbol{\theta}$ with bootstrapped 2.5% and 97.5% percentiles for each government policy observed over the period of study. The associated numeric values are given in S5 Table.

The scaling parameter estimate for the no-intervention period, $\widehat{\theta}_1 \approx 1.27$ (95% confidence interval (CI) 0.90 – 1.94), indicates that the social contact patterns based on the POLYMOD study may be slightly under-estimating the current Irish contact patterns. Perhaps somewhat surprisingly, the parameter $\widehat{\theta}_2$, corresponding to an initial school closure period, is greater than one (95% CI 1.78 – 2.72). However, this might be explained by the fact that it corresponds to a short period of time where no other measures had yet been introduced (apart from pub closures later in the period), but with an imminent government announcement of a strict nationwide lockdown expected – in line with what had been observed in other countries already by this stage in the global pandemic. During this period there was frenzied panic buying and stock piling of goods, increased travel across the country, and possibly increased social gatherings prior to movement restrictions. The remaining scaling parameters behave as expected based on the level of restrictions in place at that time: the higher the levels of restrictions, the smaller the scaling parameter, corresponding to reduced social mixing. The

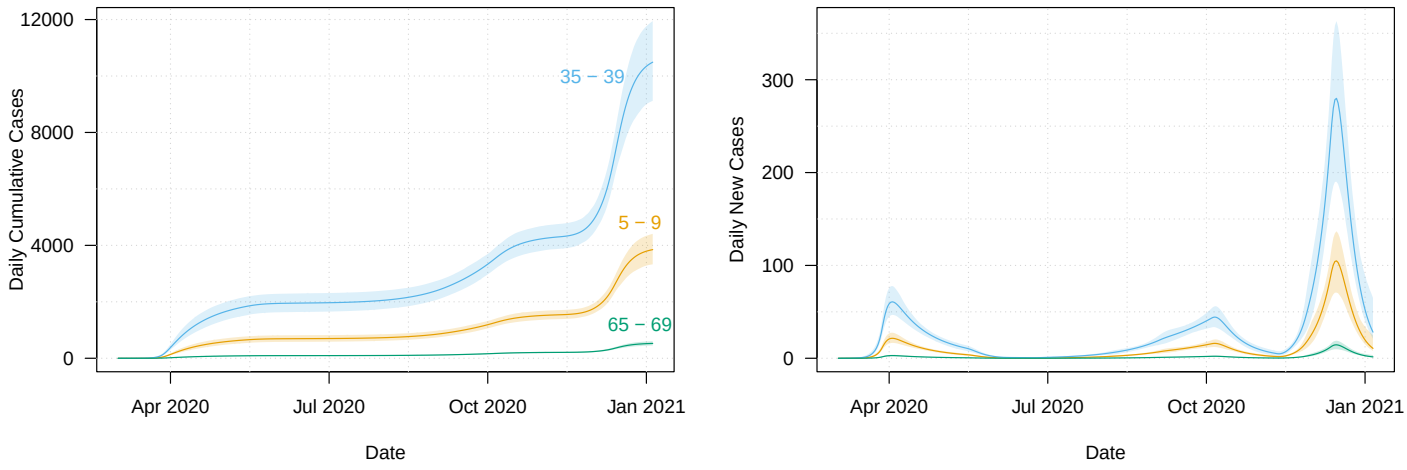


Figure 5.5: Case numbers for selected age classes obtained from the forward simulations of the SEIR with 95% intervals from bootstrapping. The age classes are 5 – 9 (orange), 35 – 39 (blue), and 65 – 69 (green).

confidence intervals just prior to and including the holiday period indicate that social mixing returned to a near normal level at that time where a dramatic spike in the case numbers was also observed. This was followed immediately by a heavy lockdown and consequent drop in case numbers; indeed, the scaling parameter for this final period has the smallest value of all.

Over the period of study, note that we have witnessed two lockdown Level 3 periods (September/October and December 2020). However, although in theory both periods were designated as "Lockdown Level 3" by the Irish government, we have applied two separate scaling parameters for these two periods as the December lockdown included some relaxations compared to a full Level 3 lockdown. Specifically, non-essential retail and services were open once again and indoor service in restaurants and cafes was also permitted. This was done to facilitate people's social needs around the holiday period, and indeed we see that the estimated parameter value for lockdown Level 3+ (December) is much larger than that for lockdown Level 3 (September/October). Again because of modifications in the execution, we have separate scaling parameters for the lockdown Level 5 in October/November and what we call the lockdown Level 5+ January 2021; the latter was stricter following the large rise in case numbers during the holiday period, and this is reflected in the small scaling value for this period as previously mentioned. Since the lockdown levels commenced with (what we label as)

a Level 2 in August 2020, we have not observed a Level 1 or 4 lockdown. Prior to August 2020, the lockdowns and relaxations were more ad-hoc and do not fit into any particular governmental lockdown level.

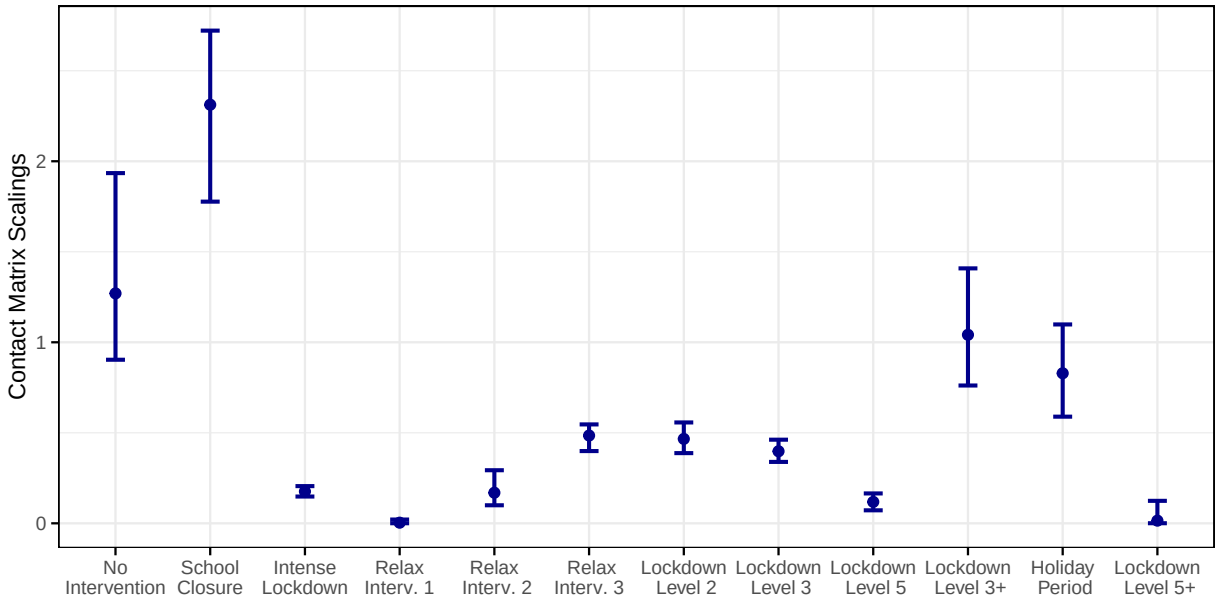


Figure 5.6: Estimates of the contact matrix scaling parameters for each lockdown period with 95% uncertainty intervals obtained through bootstrapping.

An important epidemiological metric is the *effective reproductive number*, $R(t)$, the expected number of secondary infections at time t. It differs from R_0, the *baseline* reproduction number, in that it changes over time and takes into account that the whole population will not be fully susceptible. A common estimate of $R(t)$ is the product of R_0 and the total proportion susceptible in the population at time t (see for example Section 2.2 of Nishiura and Chowell (2009)). In the context of our model, the baseline reproductive number is $R_0\theta_k$ (rather than just R_0) to account for the rate at which individuals interact with each other; recall from Section 5.2.3 that θ_k is the scaling parameter corresponding to the time interval $\mathcal{I}_k = (r_k, r_{k+1}]$. So our estimate of the effective reproductive number is

$$R(t) = R_0\theta_k\widetilde{S}(t) \tag{5.5}$$

where $\widetilde{S}(t) = \sum_{i=1}^{A} S_i(t) / \sum_{i=1}^{A} N_i(t)$ is the proportion of susceptible individuals in the the entire population at time t. Fig. 5.7 displays the estimate of $R(t)$ based on our fitted model. We see here that prior to the first heavy lockdown in April, $R(t)$

was initially very large. This dropped below one following that first lockdown, but gradually increased again over the summer period when relaxations were introduced. It was brought back under control with successive Level 3 and Level 5 lockdowns, but markedly increased over the run-up to the December holiday period; $R(t)$ was then driven towards zero with the lockdown Level 5+.

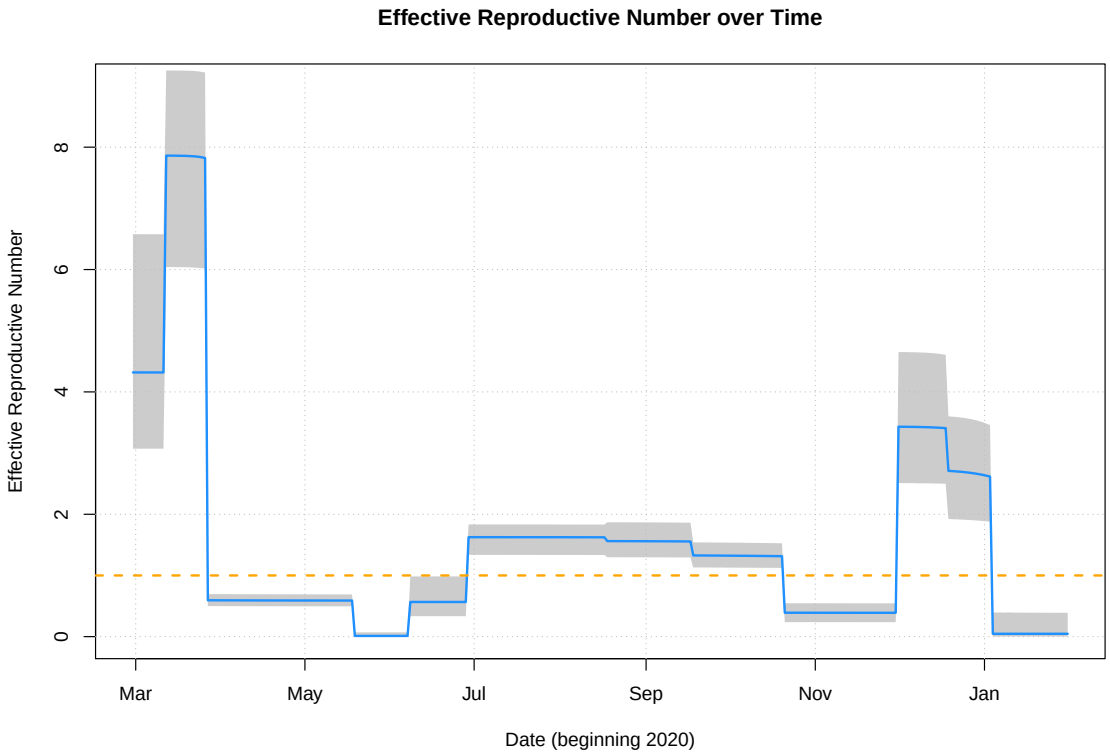


Figure 5.7: The effective reproduction number $R(t)$. The solid line was calculated from the best fit parameters, and the uncertainty intervals were drawn by computing $R(t)$ for each bootstrap replicate and selecting the 2.5% and 97.5% quantiles at each time point.

5.6 Shiny App Forecasting

Compartmental models are often used to create projections of future growth of compartments under certain conditions. One useful outcome of our research is that with we can incorporate the impact of lockdown measures in such projections to compare the potential impact of policy decisions. As a proof-of-concept, we built a supplementary app to create these predictions complete with bootstrapped confidence intervals. This app was built in `R` version 4.0.3 using the `shiny` package (Chang et al., 2022), heavily utilising the `shinydashboard` package (Chang et al., 2018). The in-

teractive output graphics were built using the `R` version of `plotly` (Sievert et al., 2020). As with the inference procedure, constructing these projections requires numerical solutions of a system of ODEs, done through `deSolve` package (Soetaert et al., 2010). The app also utilises the `Matrix` (Bates et al., 2022), `tidyverse` (Wickham et al., 2019) and `doParallel` (Corporation and Weston, 2022) packages. The code is available at `https://github.com/DanDempsey/DD_Thesis_Files/tree/master/SEIR/ForecastApp`. The app can be run in rstudio by opening the `global.R` file and clicking the 'Run App' button on the script panel. Of course, the user will have to install all the necessary packages before it will work.

The app, as is normal for `shiny`, is the union of a `ui.R` file, where the user interface settings are defined, a `server.R` file that creates the desired output based on the user settings, and a `global.R` file that loads in the required files and data and sets the needed parameters. When setting the working directory inside the `ForecastApp` folder, the app can be activated by typing `shiny::runApp()` into the `R` console. Alternatively, if using the Rstudio IDE, the usercan simply click on the 'Run App' button above the script panel.

Once running, the user will see two tabs on the left-hand panel. One of which, labelled 'Info' simply contains basic background information. The main 'Forecast Settings' tab is where the forecasts are created. Starting from the 1st of February the user can create an 8 week forecast, with a choice of different of different lockdown effects for each subsequent two week period.

There are a number of input widgets available to tune the forecast. The orange box on the right, shown in figure 5.8, allows us to choose what compartment to display, the start date of the forecast (up to 1st February 2021) and the included age groups.

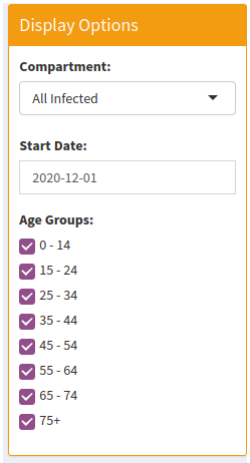


Figure 5.8: The Display Options widget.

Note that the compartment and age groups can be toggled *after* the forecast is created without needing to re-run the ODE solver.

The widget that controls the desired lockdown levels are located on the center of the page, underneath the main panel, shown in figure 5.9. These will also display the estimated cost per day of the selected lockdown levels, whose estimates are based on Jaouimaa et al. (2021).

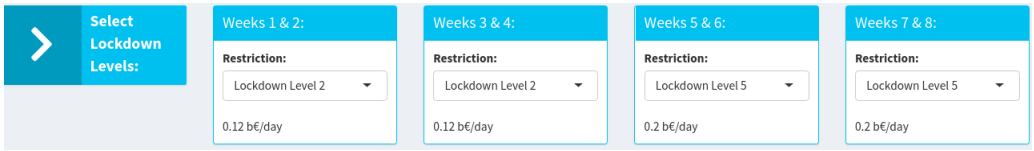


Figure 5.9: The lockdown specification widget.

The lockdown levels range from level 1 to 5, with the intent to replicate the Irish government's system. The effect of lockdown levels 2, 3 and 5 on the contact matrices are taken from our analysis in the above sections, but levels 1 and 4 were never actually implemented, so their effect is simply linearly interpolated.

Once the desired inputs have been set, the user can create the forecast by clicking the 'Create Forecast' button, located on the left beside the lockdown widgets. The forecast is created by numerically solving the ODEs for the specified time period as well as for the 1000 parametric bootstrap samples. These calculations are spread across all but one of the machine's CPUs to accommodate the computational cost. After it has finished running an image will appear on the main panel. Example output is shown in figure

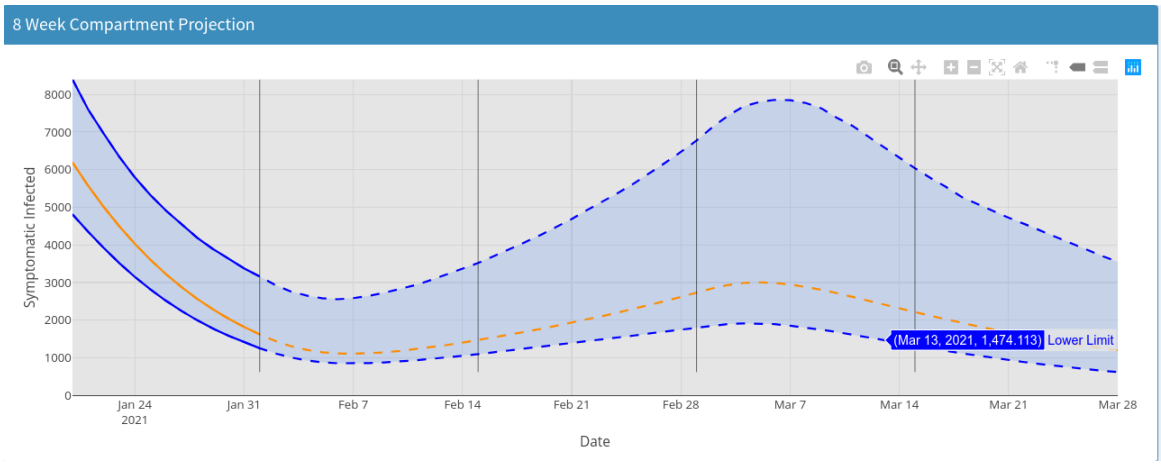


Figure 5.10: Example output from the main panel.

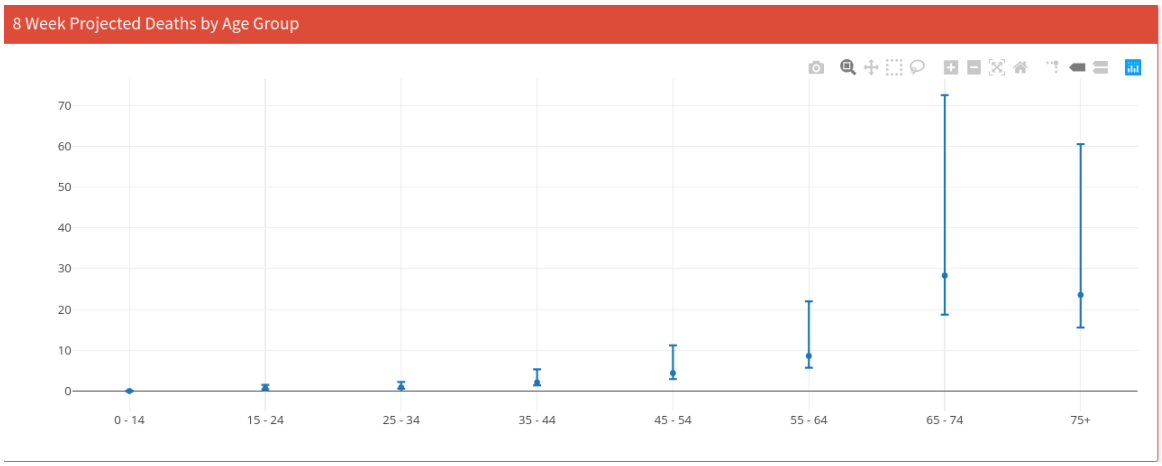


Figure 5.11: Example output from the main deaths and costs panels.

5.10. The orange line is the forecast based on the real data and the blue region denotes an empirical 95% confidence interval based on the bootstrap replicates. As a reminder, the user can change the displayed compartment and age groups and the output will change immediately without the need of clicking the 'Create Forecast' button again, but re-computation is necessary if the user wants to change the lockdown levels.

The bottom panels will also display output to that shows the estimated number of deaths, broken down by age in an error bar plot, and total age groups in an info box. Deaths are estimated by applying the average death rates in Ireland to the Symptomatic Infected compartment. The estimated total costs of the specified lockdown measures over the 8 week period are also displayed in an info box. These are shown in figure 5.11. The age group selection in the display option also applies to this output.

6 Conclusion

6.1 General Remarks

The focus of most of this thesis has been establishing a regression method for analysing imbalanced binary datasets and quantifying the probability that given covariates are predictive of its value. We introduced the concept of distributed lag models, and more specifically the DLM/MIDAS approaches in chapter 2, along with other concepts required for our modelling purposes, such as binary response imbalance and variable selection.

We broadened the frequentist approach of inferring distributed lag models in chapter 3 so that generalised response data could be appropriately modeled. We further explained some useful tools not strictly specific to frequentist inference, such as introducing notation (like the weight matrix $\mathbf{W}$) allowing us to succinctly extend the model for multiple covariates, and derivation of the gradient for logistic MIDAS regression using the two-degree Nealmon as DLF. We briefly discussed some challenges of practical implementation. Finally, we performed a simulation study which appeared to show promising results, except when the time window was set too small.

We transitioned to a Bayesian approach in chapter 4. On top of the regularisation afforded by the prior, the Bayesian approach bestows a truly elegant implementation of quantile regression and variable selection routines, and this is what we consider the most important novel contribution of this thesis. Simulation studies showed that the resulting MCMC algorithm was very effective when the correlation between the covariates was low. Results were more mixed in the presence of high collinearity but overall it still performed relatively well. We applied this method to investigate the

correlation between pollution and weather exposure of ANCA vasculitis patients and the propensity of their flare events, the motivating question behind our research into distributed lag models. We ultimately found that none of the air quality measurements were predictive of flare events in our dataset.

Despite our focus on the topic of vasculitis, we believe the methods outlined throughout this manuscript can find use in broader applications of distributed lag problems, especially when dealing with large covariate sets and response imbalance.

In chapter 5 we pivoted towards epidemiology and modelling the effect of lockdowns during the COVID–19 pandemic in Dublin taking into account age-specific contact rates that can be deployed to an international context with the appropriate data. Where feasible, the parameters of our model governing disease spread have been estimated from publicly available Irish epidemiological data with a bootstrapping approach used to determine parameter uncertainties. Our fitted model captures much of the structure observed in daily case numbers. Our approach allows for local adaptions and calibrations of models in any region or location where such data is available. We also present a prototype for an app that can create incidence projections under a number of hypothetical government intervention strategies in conjunction with approximate economic costings. This framework is easy to interpret and suitable for describing counterfactual scenarios, which could assist policy makers with regard to minimising morbidity balanced with the costs of prospective suppression strategies.

6.2 Suggestions for Future Research

We feel it is worth revisiting the question of how environmental exposure affects flare event rates in the future when more accurate data is available. As we said in section 4.6, the environmental spatio-temporal snapshots linked to each patient were mostly based on their home address, but for patients who spend a large amount of time away from home this may not be truly representative of their exposure. Originally, we had intended to include more mobile health (or 'mHealth') data via smartphone telemetry but there was not enough engagement for that to be feasible. Cajita et al. (2018) discuss potential reasons for low mHealth uptake among older adults in the context of heart disease. With more accurate location data and mHealth telemetry, we may

yet discover a strong predictor in the environment that can be leveraged to save lives. Another point of consideration about the data are the environmental values; currently they are taken from satellite reanalysis (Inness et al., 2019; Hersbach et al., 2020). More direct land measurements may be more accurate, but as of now Ireland does not have sufficient spatial coverage of land-based measurement stations to be useful for analysis.

There is a lot of scope for future research for the Bayesian Quantile-MIDAS model itself. As previously mentioned in chapter 4, it is worth investigating other methods of updating the DLF parameters $\boldsymbol{\theta}$, for example the Metropolis-Adjusted Langevin Algorithm (MALA) (Roberts and Tweedie, 1996). If we label the parameter of interest as $\mathbf{x}$, a vector of length d, MALA uses the following proposal distribution:

$$g(\mathbf{x}^\star|\mathbf{x}) = \mathrm{N}\left(\mathbf{x} + \frac{h}{2}\nabla \log \pi(\mathbf{x}), h\mathbf{I}_d\right) \tag{6.1}$$

where $\mathbf{I}_d$ is the $d \times d$ identity matrix and $h > 0$ is a tuning parameter set by the user. The proposal is accepted with the Metropolis Hastings acceptance probability,

$$\alpha = \min\left(1, \frac{\pi(\mathbf{x}^\star)g(\mathbf{x}|\mathbf{x}^\star)}{\pi(\mathbf{x})g(\mathbf{x}^\star|\mathbf{x})}\right).$$

MALA utilises the gradient of the log posterior. We have already derived (a multiple of) the gradient of the log-likelihood for the distributed lag parameters when using the Nealmon DLF in logistic regression in section 3.2.5, so MALA may be applied here if the derivative of the log prior can be calculated.

We anticipate that smartphone data (putting issues of engagement aside for now) will become a vital tool for uncovering the relationship between the environment and human health, though data from smartphones, especially those requiring active engagement, will likely be sampled irregularly. As of now our model cannot truly model this for reasons outlined in section 3.2.5, but in the future we aim to remove this limitation. An approach we have in mind is to characterise the weight function using b-splines (Di-Matteo et al., 2001). B-splines are piecewise k-degree polynomial functions connected over a grid of indices $t_0, \ldots, t_B$ called knots. The polynomials are connected in such a

way that all up to (and including) the $n-1$ degree derivatives are continuous at the points of connection. Letting the spline between knots t_i and t_{i+1} be denoted as $\phi_i(t)$, we also impose the constraint that

$$\sum_{i=0}^{B} \phi_i(t) = 1$$

for all t between the endpoints t_0 and t_B. We can apply this to distributed lag models by modelling the lag weighting function $w(t)$ as a linear combination of b-splines,

$$w(t) = \sum_{i=0}^{B} \theta_i \phi_i(t)$$

The endpoints t_0 and t_B are the edges of the time window, with the knots placed within. The more knots we have the more flexible the fit at the cost of heavier parameterisation. The parameters $\boldsymbol{\theta}$ are linear and so straightforward to fit, but there is the added drawback that this non-parametric approach might assign a negative weight at some lags, which we suspect for most applications is not realistic or desired. We can implicitly enforce positive values using transformations but this comes at the cost of spoiling the linearity of $\boldsymbol{\theta}$. Regardless this is an idea we intend to pursue. Wilson et al. (2017a) applied a similar idea in the context of distributed lag functions utilising principle components.

A useful feature to implement into the model going forward would be a random effect to account for group heterogeneity, in our case the patients. For example, in the case of continuous response, the MIDAS model can be re-written as

$$y_{jt} = \beta_0 + \sum_{i=1}^{P} \beta_i \sum_{s \in \mathcal{S}_t^i} w(\Delta t_s; \boldsymbol{\theta}_i) x_{js}^{(i)} + \nu_{jt} + \epsilon_t \tag{6.2}$$

where the subscript j denotes a patient index and ν_{jt} is the patient specific variation from the population at time t which can be modelled by a Gaussian process (see for example Bishop and Nasrabadi (2006)). This would require some tuning, primarily choosing an appropriate covariance function. We could also use random effects to account for spatial heterogeneity, as Warren et al. (2020a) done.

Another project we want to work on in the future is the refinement of the user interface and optimisation of the code described in appendix C. We intend to upload the finished product to the Comprehensive R Archive Network as a freely available package. This will include translating back-end calculations into `C++` to improve efficiency.

The SEIR model of chapter 5 might also benefit from some refinement. We currently assume that the disease spread parameters are uniform across all age groups. This is a simplifying assumption but a necessary one due to the unavailability of such information for the Irish population and the limited literature on such parameters elsewhere. We have also assumed that the effect of the non-pharmaceutical interventions is uniform across age-groups. This could be relaxed by allowing age-dependent scaling parameters for all interventions as follows

$$\begin{pmatrix} \theta_{k1} & 0 & \dots & 0 \\ 0 & \theta_{k2} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & \theta_{kA} \end{pmatrix} \begin{pmatrix} c_{1\,1} & \dots & c_{1\,A} \\ c_{2\,1} & \dots & c_{2\,A} \\ \vdots & \dots & \vdots \\ c_{A\,1} & \dots & c_{A\,A} \end{pmatrix} = \begin{pmatrix} \theta_{k1}c_{1\,1} & \dots & \theta_{k1}c_{1\,A} \\ \theta_{k2}c_{1\,1} & \dots & \theta_{k2}c_{2\,A} \\ \vdots & \dots & \vdots \\ \theta_{kA}c_{A\,1} & \dots & \theta_{kA}c_{A\,A} \end{pmatrix}, \qquad k = 1, \dots, N.$$

Here, the non-zero elements of the leading diagonal matrix represent the effect of lockdown on each age group, and these are free parameters which would need to be inferred. Thus, with $A = 16$ (age groups) and $N = 12$ (lockdown periods), this extension yields almost 200 parameters to be estimated. We can simplify slightly by adding constraints on some diagonal elements to be equivalent (for example, group them into young/middle/old) or perhaps use a regularisation approach to reduce the effective number of estimated parameters, but even this may be over-reaching for shorter lockdown periods where there is very little data. Such approaches might be more feasible if the available daily case data were broken down by each age, as it could essentially be viewed as 16 separate optimisation problems. However, notwithstanding the fact that such data are not publicly available in Ireland, at such a fine scale we would expect to have sparse case count data for some age groups wherein estimation of the intervention effects would be challenging.

Our model built upon the social interaction matrices provided by Prem et al. (2017)

which are confined to four social settings, where we model changes due to public mobility restrictions through a rescaling approach. However, with contact tracing for confirmed cases being used as a control strategy in a number of countries, access to such data would provide an avenue to substantially improve social mixing models, perhaps in conjunction with carefully constructed large scale public mobility surveys. For example, we could expand the number of social mixing venues to more than four with a better understanding on disease transmission settings. This would allow for incorporation of specific venues such as bars or restaurants. Alternatively, we could model the sociability parameters using covariates to describe the specific lockdown (e.g., schools open/closed, pubs open/closed, public events, restrictions in households etc.) rather than fixing these to have a constant value in a given lockdown level. This would allow us to make comparisons between Level 5 and Level 5+ for example – a holiday period effect could be included as another covariate. It would also allow us to construct new hypothetical lockdown regimes.

Another extension to our approach would be to incorporate uncertainty estimates around the mechanistic parameters of the SEIR model. To obtain reasonable uncertainty intervals on these parameters we might try to use a central composite design scheme used in response surface construction Box and Draper (1987), based on some transformation of these uncertainty intervals. However, such an approach would introduce a steep computational overhead and would require sufficient coding and hardware solutions to enable a time-feasible implementation. If attention is focused on a small subset of mechanistic parameters then the problem is less demanding and this can be handled instead by straightforward bootstrapping. For example Prem et al. (2020) fits a model and quantifies uncertainty via bootstrapping when only allowing the reproduction rate R to vary. Another option is to include these parameters in the optimisation alongside the contact matrix scales, while imposing heavy box constraints on the range of candidate values for these parameters within the optimisation, as done so by Náraigh and Byrne (2020).

Another potential complication with modelling infectious diseases is their propensity to mutate over time. More infectious strains will manifest in the data as rising case numbers that we can feed into the model but if it evolves rather drastically, as seems to be the case with some coronavirus variants (Li et al., 2021a), then it may be necessary

to re-evaluate the parameters.

A Bayesian MIDAS MCMC Algorithm

Bayes Quantile MIDAS MCMC, Using Two Degree Nealmon DLF

Input: Covariate data $\mathbf{X}$, response data $\mathbf{y}$, the quantile q, starting values $\boldsymbol{\beta}$, $\boldsymbol{\theta}$, $\boldsymbol{\gamma}$. Prior mean and variance for $\boldsymbol{\beta}$, $\mathbf{b}$ and $\mathbf{v}$ respectively. Also need to give dimension jumping proposal distributions for variable selection, $g_{\theta^+}(\cdot)$. Hyperparameters a and b for the $\boldsymbol{\gamma}$ Beta hyperprior.

Output: Samples from the posterior distributions $\boldsymbol{\beta}$, $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$.

procedure MCMC

`% % Setup: define immutable parameters`

$\psi = \frac{1-2q}{q(1-q)}, \quad \omega = \frac{2}{q(1-q)}, \quad \delta = 2 + \frac{\psi^2}{\omega}.$

$s_2 = 2.38^2/2, \quad N =$ size of response data $\mathbf{y}$.

`% % Initialise latent variables`

$\nu_j = 1, j = 1, \ldots, N$

$z_j = 0, j = 1, \ldots, N$

$\Omega = \text{diag}(\omega^2, \boldsymbol{\nu})$

`% % Main loop`

for $i = 2$ to number of MCMC iterations **do**

$\mathbf{V}_{\boldsymbol{\gamma}} = (\mathbf{v}_{\boldsymbol{\gamma}}^{-1} + \mathbf{X}_{\boldsymbol{\gamma}}^{\top}\boldsymbol{\Omega}^{-1}\mathbf{X}_{\boldsymbol{\gamma}})^{-1}$

$\mathbf{B}_{\boldsymbol{\gamma}} = \mathbf{V}_{\boldsymbol{\gamma}}(\mathbf{v}_{\boldsymbol{\gamma}}^{-1}\mathbf{b}_{\boldsymbol{\gamma}} + \mathbf{X}_{\boldsymbol{\gamma}}^{\mathrm{T}}\boldsymbol{\Omega}^{-1}(\mathbf{z} - \psi\boldsymbol{\nu}))$

14: % % Update $\boldsymbol{\gamma}$

15: Randomly select component of $\boldsymbol{\gamma}$ to propose a change, say $\gamma_k^\star$, i.e., if $\gamma_k = 1$ then $\gamma_k^\star = 0$ and vice-versa. If this move results in the birth of a new variable, sample corresponding $\{\theta_1, \theta_2\}$ from proposal $g_{\theta^+}(\boldsymbol{\theta})$.

16: Let $\zeta^\star = \log(-\theta_2^\star)$.

17: Compute first factor of the acceptance ratio, call it r;

$$r = \frac{|\mathbf{v}_{\boldsymbol{\gamma}^\star}|^{-1/2}|\mathbf{V}_{\boldsymbol{\gamma}^\star}|^{1/2}\exp\left\{\frac{1}{2}\left(\mathbf{B}_{\boldsymbol{\gamma}^\star}^\top\mathbf{V}_{\boldsymbol{\gamma}^\star}^{-1}\mathbf{B}_{\boldsymbol{\gamma}^\star} - \mathbf{b}_{\boldsymbol{\gamma}^\star}^\top\mathbf{v}_{\boldsymbol{\gamma}^\star}^{-1}\mathbf{b}_{\boldsymbol{\gamma}^\star}\right)\right\}\pi(\boldsymbol{\gamma}^\star)}{|\mathbf{v}_{\boldsymbol{\gamma}}|^{-1/2}|\mathbf{V}_{\boldsymbol{\gamma}}|^{1/2}\exp\left\{\frac{1}{2}\left(\mathbf{B}_{\boldsymbol{\gamma}}^\top\mathbf{V}_{\boldsymbol{\gamma}}^{-1}\mathbf{B}_{\boldsymbol{\gamma}} - \mathbf{b}_{\boldsymbol{\gamma}}^\top\mathbf{v}_{\boldsymbol{\gamma}}^{-1}\mathbf{b}_{\boldsymbol{\gamma}}\right)\right\}\pi(\boldsymbol{\gamma})}.$$

18: Note: $\mathbf{B}_{\boldsymbol{\gamma}^\star}$ and $\mathbf{V}_{\boldsymbol{\gamma}^\star}$ are computed the same way as $\mathbf{B}_{\boldsymbol{\gamma}}$ and $\mathbf{V}_{\boldsymbol{\gamma}}$ above, using only the columns of $\mathbf{X}$ supported by $\boldsymbol{\gamma}^\star$.

19: Compute the second factor of the acceptance probability α:

20: **if** γ_k originally equalled 0 **then**

21:
$$\alpha = \min\left(1, r\frac{\pi(\theta_1^\star)\pi(\theta_2^\star)}{g_{\theta^+}(\theta_1^\star)g_{\theta^+}(\zeta^\star)}\exp(\zeta^\star)\right)$$

22: **else**

23:
$$\alpha = \min\left(1, r\frac{g_{\theta^+}(\theta_1)g_{\theta^+}(\zeta)}{\pi(\theta_1)\pi(\theta_2)}\frac{1}{\exp(\zeta)}\right)$$

24: **end if**

25: Accept $\boldsymbol{\gamma}^\star$ with probability α. $\mathbf{B}_{\boldsymbol{\gamma}}$ and $\mathbf{V}_{\boldsymbol{\gamma}}$ are re-computed if so. Otherwise $\boldsymbol{\gamma}$ remains the same as the previous iteration.

26:

27: % % Update $\boldsymbol{\gamma}$ prior hyperparameter p

28:
$$p|\boldsymbol{\gamma} \sim \text{Beta}\left(a + \sum_i \gamma_i, b + \sum_i (1-\gamma_i)\right)$$

29:

30: % % Update $\boldsymbol{\beta}$

31:
$$\boldsymbol{\beta}|\mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\nu} \sim \text{N}(\mathbf{B}_{\boldsymbol{\gamma}}, \mathbf{V}_{\boldsymbol{\gamma}})$$

% % Update latent variable $\mathbf{z}$

$$z_i|y_i, \boldsymbol{\beta}, \boldsymbol{\theta} \sim \text{TruncALD}(\mathbf{x}_i^\top \boldsymbol{\beta}, \sigma = 1, q), i = 1, \ldots, N$$

Only sample from the positive axis if $y_i = 1$, otherwise only sample from the negative axis.

% % Update latent variable $\boldsymbol{\nu}$

$$\chi_i^2 = \frac{(z_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2}{\omega^2}$$

$$\nu_i|z_i, \boldsymbol{\beta}, \boldsymbol{\theta} \sim \text{GIG}(1/2, \chi_i^2, \delta_i^2)$$

% % Update $\boldsymbol{\theta}$

for $j = 1$ to number of covariates supported by $\boldsymbol{\gamma}$ **do**

Update the adaptive proposal $\boldsymbol{\Sigma}_j^i$:

$$\boldsymbol{\Sigma}_j^i = s_2 \text{cov}(\boldsymbol{\theta}_j^1, \cdots, \boldsymbol{\theta}_j^{i-1}) + s_2 \epsilon \mathbf{I}_2$$

Propose new θ_1 and ζ parameters for covariate j, from random walk proposal distribution:

$$g_\theta(\theta_{j1}^\star, \zeta^\star | \theta_{j1}, \zeta) = \text{N}\left((\theta_{j1}, \zeta)^\top, \boldsymbol{\Sigma}_j^i\right)$$

Compute acceptance probability,

$$\alpha_\theta = \min\left(1, \frac{\pi(\boldsymbol{\theta}_j^\star | \boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\nu}, ...)}{\pi(\boldsymbol{\theta}_j | \boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\nu}, ...)} \frac{\exp(\zeta^\star)}{\exp(\zeta)}\right)$$

The current iteration of $\boldsymbol{\theta}_j$ is set to $\boldsymbol{\theta}_j^\star$ with probability α_θ. Otherwise it retains its value from the previous iteration.

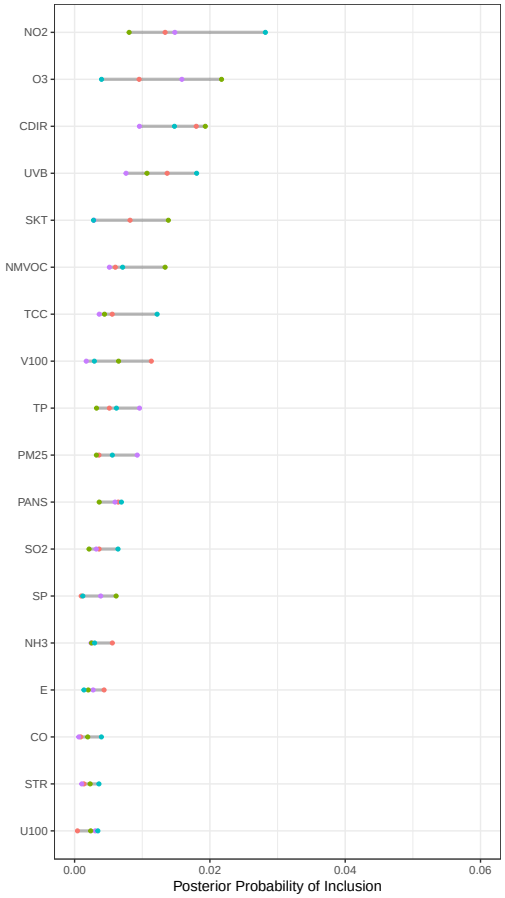
end for

end for

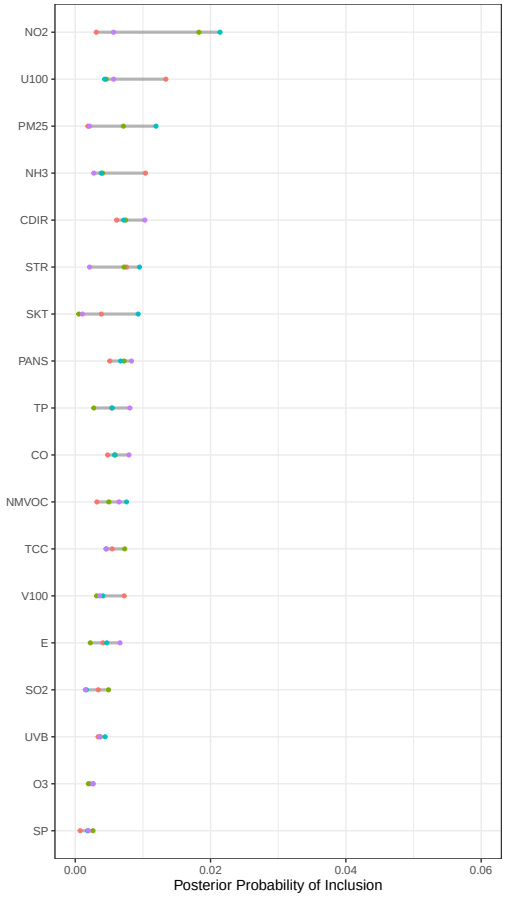
Return posterior sample.

end procedure

B Flare Data Analysis Sensitivity to Assumed Onset Date



(a) Using clinic visit as flare onset date.



(b) Clinic visit minus 20 days as onset date.

The results from re-running the flare data analysis from chapter 4 but using a different flare onset date (recall that we used the clinic visit minus 10 days for the main analysis). The x-axis is scaled to match figure 4.9. These figures show an even lower acceptance rate than what was found in section 4.6.

C Software Implementation of Methodology

Here we describe our code for implementing the models discussed in chapters 3 and 4. Our code is written entirely through the statistical computing software R (R Core Team, 2022) and depends heavily on third-party packages downloaded from the Comprehensive R Archive Network (link: https://cran.r-project.org/) that we will cite as they come up. All of the code is available in Github through the following link: https://github.com/DanDempsey/DD_Thesis_Files/tree/master/BayesQMIDAS

C.1 MIDAS Regimes

Distributed lag / MIDAS models offer a lot of flexibility that can be challenging to translate into an efficient user interface. We built our software to use a two stage approach: the first stage is to define the MIDAS characteristics of the covariates and response; i.e., their values and time indices, their time window, and the distributed lag functions they should use. In the code, we call this a *regime object*, and they can be created by feeding in your covariates into the `midas_regimes` function, whose definition is

```
midas_regimes <- function(value, time, group, time_window = 30,
                          DLF = "irts_nealmon", DLF_parameters,
                          DLF_gradient = "nealmon_gradient_fun")
```

`value` are the covariate's observations supplied as a matrix or dataframe and `time` are the time indices of the observations supplied as a vector. `group` is to assign a hierarchical ordering to the rows, for example patient labels. If there is no grouping in

the data then this can be left blank. `time_window` supplies the time window length, by default equal to 30. The `DLF` argument can be a function or a string referring to the name of function in the global environment. The given function must have two named arguments: `delta_t`, the vector of time lags, and `theta`, the vector of DLF parameters. The DLF parameters must be supplied as they will be used as starting values for numerical optimisation and MCMC algorithms. It is also a way for the algorithm to know how many components should be in $\boldsymbol{\theta}$. The default DLF is the two degree Nealmon:

```
irts_nealmon <- function(delta_t, theta) {

  lw <- matrix(rep(delta_t, each = 2) ^ (1:2),
               ncol = 2, byrow = TRUE) %*% theta
  exp(lw - matrixStats::logSumExp(lw))

}
```

Notice the numerator and denominator are computed on the log scale, partially using the `logSumExp` function from the `matrixStats` package (Bengtsson, 2022), to avoid potential numerical problems as suggested in section 3.2.2.

Optionally, you can also supply the gradient for the $\boldsymbol{\theta}$ parameters. We by default supply the Nealmon gradient (3.15) from section 3.2.5, which goes by the name `nealmon_gradient_fun`:

```
nealmon_gradient_fun <- function( delta_t, theta, M ) {

  dat <- matrix( rep( delta_t, each = 2 ) ^ ( 1:2 ),
                 ncol = 2, byrow = TRUE )
  w1 <- exp( dat %*% theta )

  dat_trim <- dat[, M ]
  sw <- sum( w1 )
  s1 <- sw * dat_trim
  s2 <- c( crossprod( w1, dat_trim ) )

  w2 <- s1 - s2
```

```
  w3 <- sw^2

  ( w2 * w1 ) / w3

}
```

Be aware: the function supplied here should only be the derivative of the DLF with respect to its parameters, *not* the entire log-likelihood gradient from section 3.2.5. The full gradient is handled by a larger function that we will talk about further below, which incorporates the function given to the `DLF_gradient` argument.

There is also a `response_regime` object for the response data to align with the regime object created for the covariates,

```
response_regime <- function(value, time, group)
```

This is more simplified since there are not as many options available to affect the response.

The point of this framework is that it allows the user to assemble the data and its distributed lag characteristics in a convenient and flexible way. The inferential code is specially built to handle these objects so that they know how to unpack and utilise them. An example of a usage case is as follows:

```
# X is a dataframe whose first p columns are covariates.
# It also contains one time index column labelled INDEX,
# and one column for the response series labelled y

mrtest <- midas_regimes(value = X[, 1:p], time = X$INDEX,
                        DLF_parameters = list(c(0, -1)),
                        time_window = 60)

rtest <- reponse_regime(value = X$y, time = X$INDEX)

formula <- rtest ~ mrtest
```

Here we have created a regime object named `mrtest` using the two-degree Nealmon as the the DLF (the default) with DLF parameters 0 and -1. The time window is set to 60 units of time. A response regime is also created.

In the event that the response variable and covariates are not observed at the same time, the matrix `X` should be constructed so that all the time indices are included in the `INDEX` column, and `NA` should be placed in the cells where the variable is not observed at the corresponding index. `midas_regimes` and `response_regime` will filter out the unobserved cells accordingly.

In the last line of the above snippet, we created a `formula` object using the response and covariate regimes. This will be used as input into the model fitting algorithms, similarly to how `glm` works.

C.2 Weight Matrix

Before we discuss the inferential algorithms, we want to draw attention to the weight matrix construction. Bear in mind the user does not need to actually use the following functions themselves over the course of standard application since they are handled automatically by the optimisation / MCMC code, but we wanted to draw attention to our implementation since it is a vital component of the distributed lag paradigm.

As discussed in section 3.2.2, the weight matrix is a substantial bottleneck when fitting the models. It's important to isolate the computations of $\Delta t_s = t - s$ since these will not change with $\boldsymbol{\theta}$ and so only need to be computed once. We compute this using the following set of functions:

```
time_delta_list <- function(regime_object) {

  regime_object$time_delta <-
    Map( time_delta_matrix,
         series = regime_object$series,
         response_index = regime_object$response_index,
         MoreArgs = list(time_window = regime_object$time_window) )
  regime_object
```

```
}

time_delta_matrix <- function(series, response_index, time_window) {

  # Calculate differences between the response indices
  # and the covariate indices
  xt <- series$time
  rownum <- length(response_index)
  colnum <- length(xt)
  yt <- matrix( rep(response_index, each = colnum), nrow = rownum,
                byrow = TRUE )
  diff_mat <- sweep(yt, 2, xt)

  # Determine which values are inside the time window,
  # store results inside as a list
  within <- ( 0 <= diff_mat ) & ( diff_mat < time_window )
  inds <- which(within, arr.ind = TRUE)
  list(nrow = rownum, ncol = colnum, i = inds[, 1], j = inds[, 2],
       x = split(diff_mat[within], inds[, 1]))

}
```

The first function, `time_delta_list`, takes a regime object as input which is unpacked and iteratively fed into the `time_delta_matrix` function, which in turn creates a matrix that subtracts each response index from all covariate indices. It then identifies which of these differences fall inside the desired time window. The output contains all the information necessary to construct a matrix in triplet format. Once the time lags are computed for every covariate, the result is included in the regime object.

The weight matrix is then constructed by quite simply applying the chosen DLF weighting scheme across the time lags using the `Map` function,

```
weight_matrix <- function(DLF, DLF_parameters, time_window_list) {

  DLF_weights <- Map( DLF, delta_t = time_window_list$x,
```

```
                        MoreArgs = list(theta = DLF_parameters) )
  Matrix::spMatrix( nrow = time_window_list$nrow,
                    ncol = time_window_list$ncol,
                    i = time_window_list$i, j = time_window_list$j,
                    unsplit(DLF_weights, time_window_list$i) )

}
```

Here is where we construct the aforementioned triplet represented matrix using the `spMatrix` function from the `Matrix` package (Bates et al., 2022), taking advantage of the sparseness of the weight matrices.

C.3 IRTS-MIDAS

The point estimate inference for the IRTS-MIDAS makes great use of the `family` functions for general linear regression, and the `optim` function. The main function the user runs is

```
irts_midas <- function(formula, data, group = NULL, start = NULL,
                       family = "binomial", n_cores = 1L,
                       gr = midas_gradient, Ofunction = "optim", ...)
```

The formula that should be supplied here is one constructed from the regime objects, for example `response_regime_object ~ covariate_regime_object`. Notice that it doesn't take time windows, DLFs, etc. as inputs as they are implicitly passed through the `formula` object. The gradient is supplied using the `gr` argument as is standard for optimisation functions. If you do not wish to use a gradient, you can pass `NULL` to `gr` instead. We will discuss this gradient function more further below.

How this function works draws heavily from the `midas_r` function from the `midasr` package (Ghysels et al., 2016). The user supplies their choice of optimisation function and the supported choices are `optim` (the default), `spg` from the `BB` package (Varadhan and Gilbert, 2009) and `optimx` from the package of the same name (Nash and Varadhan, 2011; Nash, 2014b). `"dry_run"` is also a valid option, but this performs no actual inference and is only for debugging purposes. Any additional arguments for

the chosen optimisation function can be supplied in the function call, granting the user a great deal of control over the fitting procedure. After sense-checking the inputs, the function unpacks the given `formula` and converts it into a more conventional format that `R` can conveniently parse through the `model.matrix` function.

The objective function is

```
midas_LOSS <- function(pars, pinds, formula, family) {

  ### Extract formula environment
  Zenv <- environment(formula)

  ### Set the DLF parameters
  par_list <- split(pars, pinds)
  Zenv$regime_object <- Map("[[<-", x = Zenv$regime_object,
                            i = "DLF_parameters",
                            value = par_list[-1])

  ### Compute the linear predictor
  WX <- model.matrix( formula, Zenv )
  WXb <- WX %*% par_list[[1]]
  eta <- family$linkinv(WXb)

  ### Return loss function
  sum( family$dev.resids(Zenv$yv, eta, 1L) )

}
```

The input `pars` is the vector of parameters as required by the `optim` function. The `pinds` argument is simply an indexing vector used to identify the different parameters and split them accordingly in the `parlist` object for easy reference. The first element of `parlist` contains the $\boldsymbol{\beta}$ vectors and the subsequent elements contain the $\boldsymbol{\theta}$ vectors for each covariate. `dev.resids`, the objective function, is minus twice times the log-likelihood of the model.

A lot of the work here is being done by the `family` suite of functions from base `R`,

which allows us to compute link functions and deviance residuals in a generic way. We also see here the `model.matrix` function acting on the unpacked formula object to conveniently compute the MIDAS design matrix as specified in section 3.2. The optimisation proceeds as the user specified through the inputs of the `irts_midas` function. The output of the function will be the output of the optimisation function used, as well as `glm` output for the $\boldsymbol{\beta}$ parameters.

Returning to the gradient, the default function we pass is `midas_gradient`,

```
midas_gradient <- function( pars, pinds, formula, family ) {

  ### Extract formula environment
  Zenv <- environment(formula)

  ### Set the parameters
  par_list <- split(pars, pinds)

  ### Compute necessary parameters
  WX <- model.matrix( formula, Zenv )
  eta <- WX %*% par_list[[1]]
  p <- family$linkinv( eta )
  yv <- Zenv$response_vector

  ### Compute common components
  dldp <- ( yv / p ) - ( ( 1 - yv ) / ( 1 - p ))
  dpde <- ( ( 1 + exp( -eta ) )^(-2) ) * exp( -eta )
  dlde <- dldp * dpde

  ### Beta components
  dldb <- crossprod( dlde, WX )

  ### Theta components
  WX_grad <- Map( midas_design_matrices, M = 1:2,
                  MoreArgs = list(regime_object = Zenv$regime_object,
                                  gr = TRUE) )
```

```
  dlde_WX_grad <- do.call( 'rbind', lapply( WX_grad, crossprod,
                                            x = dlde ) )
  dldt_mat <- sweep( dlde_WX_grad, 2, par_list[[ 1 ]][-1], '*' )
  dldt_list <- split( dldt_mat, rep( 1:ncol( dldt_mat ), each = 2 ) )

  ### Return result
  # Minus two since objective is -2 * log likelihood
  -2 * c( dldb, unlist( dldt_list ) )

}
```

This gradient is only suitable for binary logistic regression; we have not implemented a version for other families of response distributions. This function computes the gradient for the $\boldsymbol{\beta}$ components and then uses the `DLF_gradient` function to compute the $\boldsymbol{\theta}$ components. We again make use of the `model.matrix` function, taking advantage of the fact that (3.14) is itself essentially a MIDAS design matrix, but using the derivative as a DLF.

C.4 IRTS-MIDAS Code Example

Let us generate a dataset named `midas_dat` from the in-built simulation function,

```
binom_innov <- function(n, x) {
  rbinom(n, prob = binomial()$linkinv(x), size = 1)
}

set.seed( 777 )
midas_dat <- irts_midas_sim( n_y = 1000, n_vars = 1,
                             seasonal_adjust = TRUE, pars = 0.01,
                             periodicity = 30, beta = c(0, 1.5),
                             innov_fun = binom_innov,
                             time_window = 15,
                             response_rate = 1/10 )$Data
```

The first 20 rows of the dataset looks like this:

```
midas_dat[1:20, ]

            num_1    INDEX  y
1     -0.20567717  1.00000 NA
2     -0.31427573  2.00000 NA
3     -0.63408463  3.00000 NA
4     -0.01976994  4.00000 NA
5     -1.23813629  5.00000 NA
6      0.02838942  6.00000 NA
7     -1.08507937  7.00000 NA
8     -0.03283709  8.00000 NA
9     -0.82776131  9.00000 NA
10    -1.97482939 10.00000 NA
11     0.32185153 11.00000 NA
12    -1.38239980 12.00000 NA
13    -0.65619993 13.00000 NA
14     0.12869793 14.00000 NA
15     0.09598911 15.00000 NA
16     0.35275984 16.00000 NA
17     0.66722227 17.00000 NA
18     0.32187474 18.00000 NA
10059          NA 18.75715  0
19     0.56524079 19.00000 NA
```

it shows we have a covariate column, followed by a time index column, followed by the response column. Wherever an index is not available for one of the values, an NA is put in its place.

We then create the regime object so that the function can unpack and analyse the data as shown in section C.1:

```
covar_ro <- midas_regimes( value = midas_dat$num_1,
                           time = midas_dat$INDEX,
                           DLF = 'irts_nealmon',
                           DLF_parameters = c(0, -1),
                           time_window = 30 )
```

We must do the same for the response variable and then create the formula object that will be used as the input for the main inference function:

```
response_ro <- response_regime( value = midas_dat$y,
                                time = midas_dat$INDEX )
form <- response_ro ~ covar_ro
```

Finally, we can fit the model. Let us use the `L-BFGS-B` method from the `optim` function, specifying an upper bound of zero for the second theta parameter. We must also be sure to remember to specify that this is a logistic regression with the `family` argument:

```
fit <- irts_midas( form, data = midas_dat, family = "binomial",
                   Ofunction = 'optim', method = 'L-BFGS-B',
                   upper = c(rep(Inf, 3), 0) )
```

the `upper` argument specifies the upper bound. The first two elements of the vector correspond to β_0 and β_1, followed by the next two that correspond to θ_1 and θ_2. The fit took 2.8 seconds on a Dell Latitude 5400 laptop.

The object `fit` is a list containing the results of the IRTS-MIDAS model fit and much more. The most important element is the `opt` element, that contains the raw output of the `optim` function. From there we can access the parameter estimates:

```
round( fit$opt$par, 2 )

(Intercept)   midas_covariate_1_beta
0.04          1.52

midas_covariate_1_theta1   midas_covariate_1_theta2
9.41                       -0.67
```

C.5 Bayes Quantile MIDAS MCMC Implementation

The MCMC method described in chapter 4 is a lot more involved. It still uses most of the same functionality discussed above, such as the regime objects, the same weight matrix construction code, and the concise formula unpacking mechanism that makes it

easy to implement through base R functionality. However, this MCMC algorithm is not quite as flexible as the frequentist inference we discussed above; it will only work for ALD likelihoods (quantile regression) and the two-degree Nealmon DLF. To implements other choices would require a complete restructuring of the MCMC algorithm.

The MCMC method is called with the following function:

```
IRTS_MIDAS_AuxVar <- function(formula, data, quantile = 0.5, prior,
                              beta_start, varsel = FALSE,
                              MCMC_length = 10000)
```

The `formula` and `data` arguments work the same way as they did above. The `quantile` arguement is the desired quantile for the regression model, as discussed in chapter 4. `beta_start` is the vector of starting values for the slope coefficients (the DLF parameters are initialised through the regime object, as before). `varsel` is an indicator expressing whether or not the user wishes to perform variable selection via RJ-MCMC. `MCMC_length` is the number of iterations the user wishes to run the Markov chain.

There is also an argument to set a `prior`; this should be a list with three named elements: `beta`, `DLF_pars`, and `vars`. `beta` should itself be a list with elements `beta0`, a vector of prior means, and `V0`, the covariance matrix of the prior. `DLF_pars` is another named list, with elements `DLF1`, a vector containing the mean and standard deviation of the θ_1 prior, and `DLF2`, a vector containing the shape and rate of the gamma distribution for θ_2 (note that for all analyses, we set the shape $= 1$, which collapses to the exponential distribution). Finally, `vars` should contain a vector that corresponds to the a and b beta distribution hyperparameters from section 4.2.1.

The function that performs the MCMC begins with initialisation of the parameters, as well as some pre-processing to help make the rest of the iterations more efficient. The first parameter we update inside the loop is the variable indicator,

```
### Update covariate indicator
V0i <- V0i_full[varsel_int, varsel_int]
V0ib0 <- V0ib0_full[varsel_int]
XtNi <- t( X_mat * Ni )
```

```
V_posti <- V0i + XtNi%*%X_mat
V_post <- chol2inv( chol(V_posti) )
B_post <- V_post%*%( V0ib0 + XtNi%*%z_pn )
```

Matrix inversions of covariance matrices are performed using Cholesky Decomposition, reducing the number of redundant calculations for positive definite matrix inversions.

```
### Propose dimension change
change_ind <- sample( var_inds, 1 )
varsel_star <- varsel
varsel_star[change_ind] <- !varsel_star[change_ind]
varsel_star_int <- c( TRUE, varsel_star )
change_name <- names(regime_object)[change_ind]

if( varsel[change_ind] ) {

  #  Propose a move to a lower dimension
  Theta_drop <- regime_object[[change_name]]$DLF_parameters
  lDLFprior <- all_DLF_prior( Theta_drop, prior )
  Theta_drop[2] <- ljacob <- log( -Theta_drop[2] )
  lproposal <- dmvnorm( t(Theta_drop), DLF_prior_mean,
                        jump_proposal_DLF_sigma,
                        log = TRUE, checkSymmetry = FALSE )
  DLF_component <- sum( lproposal, -lDLFprior, -ljacob )
  death_oppurtunity[change_name] <-
    death_oppurtunity[change_name] + 1

} else {

  # Propose a move to a higher dimension
  Theta_star <- t( rmvnorm( 1, DLF_prior_mean,
                   jump_proposal_DLF_sigma,
                   checkSymmetry = FALSE ) )
  lproposal <- dmvnorm( t(Theta_star), DLF_prior_mean,
```

```
                        jump_proposal_DLF_sigma,
                        log = TRUE, checkSymmetry = FALSE )
  ljacob <- Theta_star[2]
  Theta_star[2] <- -exp( Theta_star[2] )
  regime_object[[change_name]]$DLF_parameters <- Theta_star
  lDLFprior <- all_DLF_prior( Theta_star, prior )
  DLF_component <- sum( -lproposal, lDLFprior, ljacob )
  birth_oppurtunity[change_name] <-
    birth_oppurtunity[change_name] + 1
  regime_object[[change_name]]$WX <-
    midas_design_matrix( regime_object[[change_name]] )

}
```

The above snippet selects an indicator at random, and then computes the relevant $\boldsymbol{\theta}$ component of the acceptance probability.

```
# Holmes and Held (2006) ratio
X_mat_star <- make_design_mat( regime_object, int, varsel_star )
V0i_star <- V0i_full[varsel_star_int, varsel_star_int]
V0ib0_star <- V0ib0_full[varsel_star_int]
XtNi_star <- t( X_mat_star * Ni )

V_posti_star <- V0i_star + XtNi_star%*%X_mat_star
V_post_star <- chol2inv( chol(V_posti_star) )
B_post_star <- V_post_star%*%( V0ib0_star + XtNi_star%*%z_pn )

ldet_V_post <- sum( log(diag(chol(V_post))) )
ldet_V_post_star <- sum( log(diag(chol(V_post_star))) )
ldet_V0i <- sum( log( diag(chol(V0i)) ) )
ldet_V0i_star <- sum( log(diag(chol(V0i_star))) )

varsel_lprior <- dbinom( varsel, 1, model_sel_prob[i-1], log = TRUE )
varsel_lprior_star <- dbinom( varsel_star, 1, model_sel_prob[i-1],
                              log = TRUE )
```

```
lkernel <- crossprod(B_post, V_posti)%*%B_post/2
lkernel_star <- crossprod(B_post_star, V_posti_star)%*%B_post_star/2

lnum <- sum( ldet_V_post_star, ldet_V0i_star, lkernel_star,
             varsel_lprior_star, DLF_component )
ldenom <- sum( ldet_V_post, ldet_V0i, lkernel, varsel_lprior )

# Acceptance probability
if ( (lnum - ldenom) > log(runif(1)) ) {
  if( !varsel[change_ind] ) {
    birth[change_name] <- birth[change_name] + 1
  } else {
    DLFres[[change_name]][i, ] <- DLFres[[change_name]][1, ]
    death[change_name] <- death[change_name] + 1
  }
  vsres[i, ] <- varsel <- varsel_star
  varsel_int <- varsel_star_int
  X_mat <- X_mat_star
  V_post <- V_post_star
  B_post <- B_post_star
} else {
  vsres[i, ] <- vsres[i-1, ]
}
```

We then compute the rest of the acceptance probability as shown above. For numerical stability, we calculate the log of the acceptance probability. Again, wherever possible, we take advantage of the positive-definiteness of covariance matrices whenever computing determinants and inverses. With the covariate indicator updated, sampling the hyperparameter p is easy:

```
### Update the prior success probability p
model_sel_prob[i] <- rbeta( 1, mod_sel_prior[1] + sum(varsel),
                            mod_sel_prior[2] + nvar - sum(varsel) )
```

Updating the the slope coefficient and latent variables is also straightforward,

```
### Update ALD parameters
# Update beta
bet <- betares[i, varsel_int] <- t( rmvnorm( 1, mean = B_post,
                                             sigma = V_post,
                                             checkSymmetry = FALSE ))

# Update z
Xb <- X_mat %*% bet
z <- rTALD( n = n_y, rtrunc = rtrunc, mu = Xb, sigma = 1,
            p = quantile )

# Update nu
chi <- ( z - Xb )^2 / omega
nu <- 1/rinvgauss( n_y, mean = sqrt( delta/chi ), shape = delta )

### Update DLF parameters
z_pn <- z - ( psi * nu )
Ni <- 1 / ( omega * nu )
ll <- ( z_pn * Ni )%*%Xb - crossprod( Xb*Ni, Xb )/2
```

The final step for each iteration is to update the DLF parameters $\boldsymbol{\theta}$. To do this we loop over every variable in the model currently supported by the covariate indicator. We then update the two Nealmon parameters jointly using a Metropolis-Hastings step with an adaptive proposal distribution.

```
# Loop over DLF updates
for (j in which(varsel)) {

  # Acceptance Ratio Denominator
  DLF_last <- DLF_trans_last <- regime_object[[j]]$DLF_parameters
  ldenom <- ll + all_DLF_prior( DLF_last, prior )

  # Proposal
  #n_updates <- sum( vsres[1:(i-1), j] )
  #if ( n_updates >= 100 ) {
```

```
if ( i >= 100 ) {
  new_metrics <- compute_covar( DLF_last, i, DLF_last_mean[[j]],
                                DLF_last_covar[[j]] )
  DLF_last_mean[[j]] <- new_metrics[[1]]
  DLF_last_covar[[j]] <- new_metrics[[2]]
}

DLF_trans_last[2] <- log( -DLF_trans_last[2] )
DLF_star <- t( rmvnorm( 1, mean = DLF_trans_last,
                        sigma = covar_sd *
                          (DLF_last_covar[[j]] + small_pos),
                        checkSymmetry = FALSE ) )
ljacob <- DLF_star[2] - DLF_trans_last[2]
DLF_star[2] <- -exp( DLF_star[2] )

# Acceptance Ratio Numerator
regime_object[[j]]$DLF_parameters <- DLF_star
WX_old <- regime_object[[j]]$WX
regime_object[[j]]$WX <- midas_design_matrix( regime_object[[j]] )
X_mat_star <- make_design_mat( regime_object, int, varsel )
Xb_star <- X_mat_star %*% bet
ll_star <- ( z_pn * Ni )%*%Xb_star -
              crossprod( Xb_star*Ni, Xb_star )/2
lnum <- ll_star + all_DLF_prior( DLF_star, prior ) + ljacob

if ( (lnum - ldenom) > log(runif(1)) ) {
  DLFres[[j]][i, ] <- DLF_star
  accept[j] <- accept[j] + 1
  ll <- ll_star
  X_mat <- X_mat_star
} else {
  DLFres[[j]][i, ] <- regime_object[[j]]$DLF_parameters <- DLF_last
  regime_object[[j]]$WX <- WX_old
}
```

```
}
```

C.6 Bayesian Quantile MIDAS Code Example

Let us use the same data `midas_sim` and formula regime object `form` that we created in section C.4. With these same objects, running the Bayesian version of the model fit is done as so:

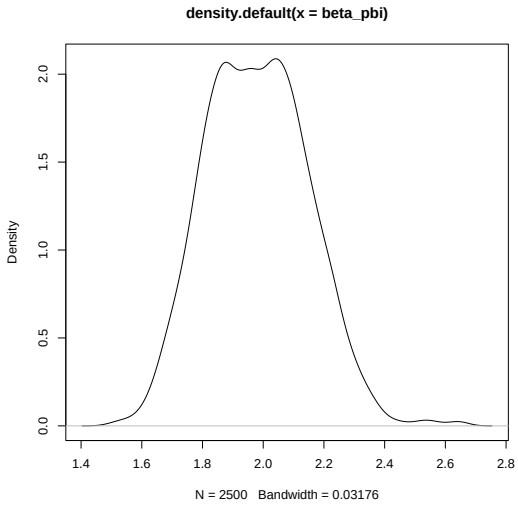
```
post_sample <- IRTS_MIDAS_AuxVar( formula = form, data = midas_dat,
                                  quantile = 0.5, varsel = TRUE,
                                  response_dist = "ald",
                                  MCMC_length = 5000 )
```

We request 5,000 iterations of RJ-MCMC. This is a somewhat small number of iterations (remember that it includes burn-in) but this is simply for the purpose of illustration. This code takes roughly 70 seconds to run on a Dell Latitude 5400.

The result is stored in `post_sample`, which contains the posterior draws of every MCMC iteration as well as a few other metrics for diagnostic purposes (such as the number of proposed births/deaths in the chain, and so on). We can examine the output as follows (discarding the first 2,500 iterations as burn-in):

```
beta_pbi <- post_sample$betares[2501:5000, 2]
gamma_pbi <- post_sample$vsres[2501:5000]

sum( gamma_pbi ) / length( gamma_pbi ) # Result is 100% acceptance
plot( density(beta_pbi) )
```


density.default(x = beta_pbi)
Density
0.0
0.5
1.0
1.5
2.0
1.4
1.6
1.8
2.0
2.2
2.4
2.6
2.8
N = 2500 Bandwidth = 0.03176

Bibliography

Albert, J. H. and Chib, S. (1993). Bayesian Analysis of Binary and Polychotomous Response Data. *Journal of the American Statistical Association*, 88(422):669–679.

Aldrich, J. H. and Nelson, F. D. (1984). *Linear probability, logit, and probit models*. Number 45. Sage.

Almon, S. (1965). The distributed lag between capital appropriations and expenditures. *Econometrica: Journal of the Econometric Society*, pages 178–196.

Andreou, E., Ghysels, E., and Kourtellos, A. (2010). Regression models with mixed sampling frequencies. *Journal of Econometrics*, 158(2):246–261.

Andrews, D. F. and Mallows, C. L. (1974). Scale mixtures of normal distributions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(1):99–102.

Antonelli, J., Wilson, A., and Coull, B. (2021). Multiple exposure distributed lag models with variable selection. *arXiv preprint arXiv:2107.14567*.

Bates, D., Maechler, M., and Jagan, M. (2022). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.4-1.

Bengtsson, H. (2022). *matrixStats: Functions that Apply to Rows and Columns of Matrices (and to Vectors)*. R package version 0.62.0.

Benoit, D. F. and den Poel, D. V. (2012). Binary quantile regression: A Bayesian approach based on the asymmetric Laplace distribution. *Journal of Applied Econometrics*, 27(7):1174–1188.

Benoit, D. F. and den Poel, D. V. (2017). bayesQR: A Bayesian Approach to Quantile Regression. *Journal of Statistical Software*, 76(1):1–32.

Beukenhorst, A., Schultz, D., McBeth, J., Lakshminarayana, R., Sergeant, J., and Dixon, W. (2017). Using smartphones for research outside clinical settings: How operating systems, app developers, and users determine geolocation data quality in mhealth studies. *Studies in health technology and informatics*, 245:10–14.

Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*, volume 4. Springer.

Box, G. E. P. and Draper, N. R. (1987). *Empirical model-building and response surfaces*. John Wiley and Sons.

Brauner, J. M., Mindermann, S., Sharma, M., Johnston, D., Salvatier, J., Gavenčiak, T., Stephenson, A. B., Leech, G., Altman, G., Mikulik, V., et al. (2021). Inferring the effectiveness of government interventions against covid-19. *Science*, 371(6531).

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics*, 37(4):373–384.

Breiman, L. (1996). Heuristics of instability and stabilization in model selection. *The annals of statistics*, 24(6):2350–2383.

Broyden, C. G. (1970). The convergence of a class of double-rank minimization algorithms. *IMA Journal of Applied Mathematics*, 6(3):222–231.

Buitrago-Garcia, D., Egli-Gany, D., Counotte, M. J., Hossmann, S., Imeri, H., Ipekci, A. M., Salanti, G., and Low, N. (2020). Occurrence and Transmission Potential of Asymptomatic and Presymptomatic SARS-CoV-2 Infections: A Living Systematic Review and Meta-Analysis. *PLOS Medicine*, 17(9):e1003346.

Buluç, A., Fineman, J. T., Frigo, M., Gilbert, J. R., and Leiserson, C. E. (2009). Parallel sparse matrix-vector and matrix-transpose-vector multiplication using compressed sparse blocks. In *Proceedings of the twenty-first annual symposium on Parallelism in algorithms and architectures*, pages 233–244.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208.

Byrne, A. W., McEvoy, D., Collins, A. B., Hunt, K., Casey, M., Barber, A., Butler, F., Griffin, J., Lane, E. A., McAloon, C., O'Brien, K., Wall, P., Walsh, K. A., and More, S. J. (2020). Inferred Duration of Infectious Period of SARS-CoV-2: Rapid Scoping Review and Analysis of Available Evidence for Asymptomatic and Symptomatic COVID-19 Cases. *BMJ Open*, 10(8):e039856.

Cajita, M. I., Hodgson, N. A., Lam, K. W., Yoo, S., and Han, H.-R. (2018). Facilitators of and barriers to mhealth adoption in older adults with heart failure. *Computers, informatics, nursing: CIN*, 36(8):376.

Canabarro, A., Tenório, E., Martins, R., Martins, L., Brito, S., and Chaves, R. (2020). Data-driven study of the covid-19 pandemic via age-structured modelling and prediction of the health system failure in brazil amid diverse intervention strategies. *Plos one*, 15(7):e0236310.

Caron, R., Sinha, D., Dey, D. K., and Polpo, A. (2018). Categorical data analysis using a skewed weibull regression model. *Entropy*, 20(3):176.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2010). The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480.

Castillo, I., Schmidt-Hieber, J., and van der Vaart, A. (2015). Bayesian linear regression with sparse priors. *The Annals of Statistics*, 43(5):1986 – 2018.

Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A., and Borges, B. (2022). *shiny: Web Application Framework for R*. R package version 1.7.2.

Chang, W., Ribeiro, B. B., RStudio, theme for Bootstrap), A. S. A., and font), A. S. I. S. S. P. (2018). Shinydashboard: Create Dashboards with 'Shiny'.

Chatterjee, A. and Lahiri, S. (2010). Asymptotic properties of the residual bootstrap for lasso estimators. *Proceedings of the American Mathematical Society*, 138(12):4497–4509.

Chatterjee, A. and Lahiri, S. N. (2011). Bootstrapping lasso estimators. *Journal of the American Statistical Association*, 106(494):608–625.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Chen, D., Courtney, R., and Schmitz, A. (1972). A polynomial lag formulation of milk production response. *American Journal of Agricultural Economics*, 54(1):77–83.

Chen, D. P., McInnis, E. A., Wu, E. Y., Stember, K. G., Hogan, S. L., Hu, Y., Henderson, C. D., Blazek, L. N., Mallal, S., Karosiene, E., et al. (2022). Immunological interaction of hla-dpb1 and proteinase 3 in anca vasculitis is associated with clinical disease activity. *Journal of the American Society of Nephrology*, 33(8):1517–1527.

Chen, M.-H., Dey, D. K., and Shao, Q.-M. (1999). A new skewed link model for dichotomous quantal response data. *Journal of the American Statistical Association*, 94(448):1172–1186.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266 – 298.

Cho, L. K., Carette, S., and Pagnoux, C. (2021). Anca status and renal parameters at month 12 post-diagnosis can help predict subsequent relapses in patients with granulomatosis with polyangiitis. *Seminars in Arthritis and Rheumatism*, 51(5):1011–1015.

Chow, G. C. and Lin, A.-l. (1971). Best linear unbiased interpolation, distribution, and extrapolation of time series by related series. *The review of Economics and Statistics*, pages 372–375.

Chow, G. C. and Lin, A.-L. (1976). Best linear unbiased estimation of missing observations in an economic time series. *Journal of the American Statistical Association*, 71(355):719–721.

Chowell, G. (2017). Fitting Dynamic Models to Epidemic Outbreaks with Quantified Uncertainty: A Primer for Parameter Uncertainty, Identifiability, and Forecasts. *Infectious Disease Modelling*, 2(3):379–398.

Clements, M. P. and Galvão, A. B. (2008). Macroeconomic forecasting with mixed-frequency data. *Journal of Business & Economic Statistics*, 26(4):546–554.

Corporation, M. and Weston, S. (2022). *doParallel: Foreach Parallel Adaptor for the 'parallel' Package*. R package version 1.0.17.

Cuevas-Maraver, J., Kevrekidis, P. G., Chen, Q. Y., Kevrekidis, G. A., Villalobos-Daniel, V., Rapti, Z., and Drossinos, Y. (2021). Lockdown Measures and Their Impact on Single- and Two-Age-Structured Epidemic Model for the COVID-19 Outbreak in Mexico. *Mathematical Biosciences*, page 108590.

Czado, C. (1994). Parametric link modification of both tails in binary regression. *Statistical Papers*, 35:189–201.

Czado, C. and Santner, T. J. (1992). The effect of link misspecification on binary regression inference. *Journal of statistical planning and inference*, 33(2):213–231.

Dal Pozzolo, A., Caelen, O., and Bontempi, G. (2015). When is undersampling effective in unbalanced classification tasks? In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part I 15*, pages 200–215. Springer.

Darmody, M., Smith, E., and Russell, H. (2020). Implications of the COVID-19 pandemic for policy in relation to children and young people: A research review. *ESRI Survey and Statistical Report Series*.

Dashtbali, M. and Mirzaie, M. (2021). A compartmental model that predicts the effect of social distancing and vaccination on controlling covid-19. *Scientific Reports*, 11(1):8191.

Devroye, L. (1986). *Non-Uniform Random Variate Generation*. Springer-Verlag.

Dhrymes, P. J. (1981). *Distributed lags; problems of estimation and formulation*, volume 2. Holden-Day.

Diekmann, O., Heesterbeek, J. A. P., and Metz, J. A. J. (1990). On the Definition and the Computation of the Basic Reproduction Ratio R0 in Models for Infectious Diseases in Heterogeneous Populations. *Journal of Mathematical Biology*, 28(4):365–382.

DiMatteo, I., Genovese, C. R., and Kass, R. E. (2001). Bayesian curve-fitting with free-knot splines. *Biometrika*, 88(4):1055–1071.

Dixon, W. G., Beukenhorst, A. L., Yimer, B. B., Cook, L., Gasparrini, A., El-Hay, T., Hellman, B., James, B., Vicedo-Cabrera, A. M., Maclure, M., et al. (2019). How the weather affects the pain of citizen scientists using a smartphone app. *NPJ digital medicine*, 2(1):1–9.

Draibe, J., Rodo, X., Fulladosa, X., Martínez-Valenzuela, L., Diaz-Encarnación, M., Santos, L., Marco, H., Quintana, L., Rodriguez, E., Barros, X., et al. (2018). Seasonal variations in the onset of positive and negative renal anca-associated vasculitis in spain. *Clinical kidney journal*, 11(4):468–473.

Efron, B. (1979). Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, 7(1):1 – 26.

Epskamp, S., Deserno, M. K., and Bringmann, L. F. (2021). *mlVAR: Multi-Level Vector Autoregression*. R package version 0.5.

Estabrooks, A., Jo, T., and Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets. *Computational intelligence*, 20(1):18–36.

Evoy, D. M., McAloon, C. G., Collins, A. B., Hunt, K., Butler, F., Byrne, A. W., Casey, M., Barber, A., Griffin, J. M., Lane, E. A., Wall, P., and More, S. J. (2020). The Relative Infectiousness of Asymptomatic SARS-CoV-2 Infected Persons Compared with Symptomatic Individuals: A Rapid Scoping Review. *medRxiv*, page 2020.07.30.20165084.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360.

Fernández, A., Garcia, S., Herrera, F., and Chawla, N. V. (2018). Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, 61:863–905.

Flaxman, S., Mishra, S., Gandy, A., Unwin, H. J. T., Mellan, T. A., Coupland, H., Whittaker, C., Zhu, H., Berah, T., Eaton, J. W., Monod, M., Ghani, A. C., Donnelly, C. A., Riley, S., Vollmer, M. A. C., Ferguson, N. M., Okell, L. C., and Bhatt, S. (2020). Estimating the Effects of Non-Pharmaceutical Interventions on COVID-19 in Europe. *Nature*, 584(7820):257–261.

Flom, P. L. and Cassell, D. L. (2007). Stopping stepwise: Why stepwise and similar selection methods are bad, and what you should use. In *NorthEast SAS Users Group Inc 20th Annual Conference*, volume 11.

Foroni, C., Marcellino, M., and Schumacher, C. (2015). Unrestricted mixed data sampling (midas): Midas regressions with unrestricted lag polynomials. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 178(1):57–82.

Foroni, C. and Marcellino, M. G. (2013). A survey of econometric methods for mixed-frequency data. *SSRN Electronic Journal*.

Frühwirth-Schnatter, S. and Frühwirth, R. (2010). Data augmentation and mcmc for binary and multinomial logit models. *Statistical modelling and regression structures: Festschrift in honour of Ludwig Fahrmeir*, pages 111–132.

Fumanelli, L., Ajelli, M., Manfredi, P., Vespignani, A., and Merler, S. (2012). Inferring the structure of social contacts from demographic data in the analysis of infectious diseases spread. *PLoS Comput Biol*, 8(9):e1002673.

Gasparrini, A. (2011). Distributed lag linear and non-linear models in R: the package dlnm. *Journal of Statistical Software*, 43(8):1–20.

Gasparrini, A., Armstrong, B., and Kenward, M. G. (2010). Distributed lag non-linear models. *Statistics in medicine*, 29(21):2224–2234.

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

Gelman, A., Roberts, G. O., Gilks, W. R., et al. (1996). Efficient metropolis jumping rules. *Bayesian statistics*, 5(599-608):42.

Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis*. Chapman and Hall/CRC.

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, 6(6):721–741.

Ghysels, E., Kvedaras, V., and Zemlys, V. (2016). Mixed frequency data sampling regression models: TheRPackagemidasr. *Journal of Statistical Software*, 72(4).

Ghysels, E., Santa-Clara, P., and Valkanov, R. (2002). The midas touch: Mixed data sampling regression models. Working paper, UNC and UNCLA.

Ghysels, E., Sinko, A., and Valkanov, R. I. (2006). MIDAS regressions: Further results and new directions. *SSRN Electronic Journal*.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. (1995). *Markov chain Monte Carlo in practice*. CRC press.

Gleeson, J. P., Brendan Murphy, T., O'Brien, J. D., Friel, N., Bargary, N., and O'Sullivan, D. J. (2022). Calibrating covid-19 susceptible-exposed-infected-removed models with time-varying effective contact rates. *Philosophical Transactions of the Royal Society A*, 380(2214):20210120.

Green, P. J. (1984). Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *Journal of the Royal Statistical Society: Series B (Methodological)*, 46(2):149–170.

Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732.

Griffin, J., Casey, M., Collins, Á., Hunt, K., McEvoy, D., Byrne, A., McAloon, C., Barber, A., Lane, E. A., and More, S. (2020). Rapid review of available evidence on the serial interval and generation time of COVID-19. *BMJ open*, 10(11):e040263.

Griffiths, W. E., Hill, R. C., and Pope, P. J. (1987). Small sample properties of probit model estimators. *Journal of the American Statistical Association*, 82(399):929–937.

Grimm, V., Mengel, F., and Schmidt, M. (2021). Extensions of the SEIR Model for the Analysis of Tailored Social Distancing and Tracing Approaches to Cope with COVID-19. *Scientific Reports*, 11(1):4214.

Haario, H., Saksman, E., and Tamminen, J. (2001). An adaptive metropolis algorithm. *Bernoulli*, pages 223–242.

Hallinan Jr, A. J. (1993). A review of the weibull distribution. *Journal of Quality Technology*, 25(2):85–93.

Hannan, E. J. (1965). The estimation of relationships involving distributed lags. *Econometrica: Journal of the Econometric Society*, pages 206–224.

Harrell, F. E. et al. (2001). *Regression modeling strategies: with applications to linear models, logistic regression, and survival analysis*, volume 608. Springer.

Harvey, A. C. and Pierse, R. G. (1984). Estimating missing observations in economic time series. *Journal of the American statistical Association*, 79(385):125–131.

Hastie, D. I. and Green, P. J. (2012). Model choice using reversible jump markov chain monte carlo. *Statistica Neerlandica*, 66(3):309–338.

Hastie, T. and Tibshirani, R. (1987). Generalized additive models: some applications. *Journal of the American Statistical Association*, 82(398):371–386.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. CRC press.

Hastings, W. K. (1970). Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika*, 57(1):97–109.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284.

He, S., Peng, Y., and Sun, K. (2020). Seir modeling of the covid-19 and its dynamics. *Nonlinear dynamics*, 101:1667–1680.

Heffernan, J., Smith, R., and Wahl, L. (2005). Perspectives on the Basic Reproductive Ratio. *Journal of the Royal Society Interface*, 2(4):281–293.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al. (2020). The era5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730):1999–2049.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Holmes, C. C. and Held, L. (2006). Bayesian auxiliary variable models for binary and multinomial regression. *Bayesian Analysis*, 1(1):145–168.

Horowitz, J. L., Bolduc, D., Divakar, S., Geweke, J., Gönül, F., Hajivassiliou, V., Koppelman, F. S., Keane, M., Matzkin, R., Rossi, P., et al. (1994). Advances in random utility models report of the workshop on advances in random utility models duke invitational symposium on choice modeling behavior. *Marketing Letters*, 5:311–322.

IEMAG (2020). A Population-Level SEIR Model for COVID-19 Scenarios. Technical report, Irish Department of Health.

Inness, A., Ades, M., Agustí-Panareda, A., Barré, J., Benedictow, A., Blechschmidt, A.-M., Dominguez, J. J., Engelen, R., Eskes, H., Flemming, J., et al. (2019). The cams reanalysis of atmospheric composition. *Atmospheric Chemistry and Physics*, 19(6):3515–3556.

Jaouimaa, F.-Z., Dempsey, D., Van Osch, S., Kinsella, S., Burke, K., Wyse, J., and Sweeney, J. (2021). An age-structured seir model for covid-19 incidence in dublin, ireland with framework for evaluating health intervention cost. *Plos one*, 16(12):e0260632.

Karangizi, A. H. and Harper, L. (2018). Small vessel vasculitides. *Medicine*, 46(2):98–106.

Kimathi, M., Mwalili, S., Ojiambo, V., and Gathungu, D. K. (2021). Age-Structured Model for COVID-19: Effectiveness of Social Distancing and Contact Reduction in Kenya. *Infectious Disease Modelling*, 6:15–23.

Kitching, A. R., Anders, H.-J., Basu, N., Brouwer, E., Gordon, J., Jayne, D. R., Kullman, J., Lyons, P. A., Merkel, P. A., Savage, C. O., et al. (2020). Anca-associated vasculitis. *Nature reviews Disease primers*, 6(1):1–27.

Knuiman, M. and Speed, T. (1988). Incorporating prior information into the analysis of contingency tables. *Biometrics*, pages 1061–1071.

Kordas, G. (2006). Smoothed binary regression quantiles. *Journal of Applied Econometrics*, 21(3):387–407.

Kozumi, H. and Kobayashi, G. (2011). Gibbs sampling methods for Bayesian quantile regression. *Journal of Statistical Computation and Simulation*, 81(11):1565–1578.

Kumar, I. E., Venkatasubramanian, S., Scheidegger, C., and Friedler, S. (2020). Problems with shapley-value-based explanations as feature importance measures. In *International Conference on Machine Learning*, pages 5491–5500. PMLR.

Kvedaras, V. and Zemlys, V. (2012). Testing the functional constraints on parameters in regressions with variables of different frequency. *Economics Letters*, 116(2):250–254.

Kwong, A. S. F., Pearson, R. M., Adams, M. J., Northstone, K., Tilling, K., Smith, D., Fawns-Ritchie, C., Bould, H., Warne, N., Zammit, S., and et al. (2020). Mental health before and during the COVID-19 pandemic in two longitudinal UK population cohorts. *The British Journal of Psychiatry*, page 1–10.

Lee, D. (2018). A locally adaptive process-convolution model for estimating the health impact of air pollution. *The Annals of Applied Statistics*, 12(4):2540–2558.

Lee, J. D., Sun, D. L., Sun, Y., and Taylor, J. E. (2016). Exact post-selection inference, with application to the lasso. *The Annals of Statistics*, 44(3):907–927.

Lee, T., Kwon, H.-D., and Lee, J. (2021). The Effect of Control Measures on COVID-19 Transmission in South Korea. *PLOS ONE*, 16(3):e0249262.

Leeb, H., Pötscher, B. M., and Ewald, K. (2015). On Various Confidence Intervals Post-Model-Selection. *Statistical Science*, 30(2):216 – 227.

Lemaître, G., Nogueira, F., and Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *The Journal of Machine Learning Research*, 18(1):559–563.

Lemos-Paiao, A. P., Silva, C. J., and Torres, D. F. (2020). A new compartmental epidemiological model for covid-19 with a case study of portugal. *Ecological Complexity*, 44:100885.

Leontitsis, A., Senok, A., Alsheikh-Ali, A., Al Nasser, Y., Loney, T., and Alshamsi, A. (2021). Seahir: A specialized compartmental model for covid-19. *International journal of environmental research and public health*, 18(5):2667.

Li, B., Deng, A., Li, K., Hu, Y., Li, Z., Xiong, Q., Liu, Z., Guo, Q., Zou, L., Zhang, H., et al. (2021a). Viral infection and transmission in a large well-traced outbreak caused by the delta sars-cov-2 variant. *medRxiv*.

Li, X., Yu, H., Xie, Y., and Li, J. (2021b). Attention-based novel neural network for mixed frequency data. *CAAI Transactions on Intelligence Technology*, 6(3):301–311.

Liu, J. S. and Wu, Y. N. (1999). Parameter expansion for data augmentation. *Journal of the American Statistical Association*, 94(448):1264–1274.

Liu, X.-Y., Wu, J., and Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Mani, I. and Zhang, I. (2003). knn approach to unbalanced data distributions: a case study involving information extraction. In *Proceedings of workshop on learning from imbalanced datasets*, volume 126, pages 1–7. ICML.

McAloon, C., Collins, Á., Hunt, K., Barber, A., Byrne, A. W., Butler, F., Casey, M., Griffin, J., Lane, E., McEvoy, D., Wall, P., Green, M., O'Grady, L., and More,

S. J. (2020). Incubation Period of COVID-19: A Rapid Systematic Review and Meta-Analysis of Observational Research. *BMJ Open*, 10(8):e039652.

McCullagh, P. (1983). *Generalized linear models*. Routledge.

McKinney, E. F., Lee, J. C., Jayne, D. R., Lyons, P. A., and Smith, K. G. (2015). T-cell exhaustion, co-stimulation and clinical outcome in autoimmunity and infection. *Nature*, 523(7562):612–616.

Mehta, P., Balakrishnan, A., Phatak, S., Pathak, M., and Ahmed, S. (2022). Diagnostic accuracy of antineutrophil cytoplasmic antibodies (anca) in predicting relapses of anca-associated vasculitis: systematic review and meta-analysis. *Rheumatology International*, pages 1–12.

Meier, L., Van De Geer, S., and Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71.

Miller, G. W. (2013). *The Exposome: A Primer*. ACADEMIC PR INC.

Mitchell, T. J. and Beauchamp, J. J. (1988). Bayesian variable selection in linear regression. *Journal of the american statistical association*, 83(404):1023–1032.

Mogliani, M. and Simoni, A. (2020). Bayesian MIDAS Penalized Regressions: Estimation, Selection, and Prediction. *arXiv:1903.08025 [econ]*.

Moor, C. C., Wapenaar, M., Miedema, J. R., Geelhoed, J. M., Chandoesing, P. P., and Wijsenbeek, M. S. (2018). A home monitoring program including real-time wireless home spirometry in idiopathic pulmonary fibrosis: a pilot study on experiences and barriers. *Respiratory Research*, 19:1–5.

Moore, S., Hill, E. M., Tildesley, M. J., Dyson, L., and Keeling, M. J. (2021). Vaccination and non-pharmaceutical interventions for covid-19: a mathematical modelling study. *The Lancet Infectious Diseases*, 21(6):793–802.

Mork, D., Kioumourtzoglou, M.-A., Weisskopf, M., Coull, B. A., and Wilson, A. (2021). Heterogeneous distributed lag models to estimate personalized effects of maternal exposures to air pollution. *arXiv preprint arXiv:2109.13763*.

Mossong, J., Hens, N., Jit, M., Beutels, P., Auranen, K., Mikolajczyk, R., Massari, M., Salmaso, S., Tomba, G. S., Wallinga, J., Heijne, J., Sadkowska-Todys, M., Rosinska, M., and Edmunds, W. J. (2008). Social Contacts and Mixing Patterns Relevant to the Spread of Infectious Diseases. *PLOS Medicine*, 5(3):e74.

Mudholkar, G. S. and George, E. O. (1978). A remark on the shape of the logistic distribution. *Biometrika*, 65(3):667–668.

Mulhern, B., O'Gorman, H., Rotherham, N., and Brazier, J. (2015). Comparing the measurement equivalence of eq-5d-5l across different modes of administration. *Health and quality of life outcomes*, 13(1):1–9.

Náraigh, L. Ó. and Byrne, Á. (2020). Piecewise-Constant Optimal Control Strategies for Controlling the Outbreak of COVID-19 in the Irish Population. *Mathematical Biosciences*, 330:108496.

Nash, J. C. (2014a). On best practice optimization methods in R. *Journal of Statistical Software*, 60(2):1–14.

Nash, J. C. (2014b). On best practice optimization methods in R. *Journal of Statistical Software*, 60(2):1–14.

Nash, J. C. and Varadhan, R. (2011). Unifying optimization algorithms to aid software system users: optimx for R. *Journal of Statistical Software*, 43(9):1–14.

Nash, J. C., Varadhan, R., et al. (2011). Unifying optimization algorithms to aid software system users: optimx for R. *Journal of Statistical Software*, 43(9):1–14.

Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *The computer journal*, 7(4):308–313.

Nishiura, H. and Chowell, G. (2009). The Effective Reproduction Number as a Prelude to Statistical Estimation of Time-Dependent Epidemic Trends. *Mathematical and Statistical Estimation Approaches in Epidemiology*, pages 103–121.

Park, T. and Casella, G. (2008). The bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686.

Pawitan, Y. (2001). *In all likelihood*. Oxford University Press.

Piironen, J. and Vehtari, A. (2017). Sparsity information and regularization in the horseshoe and other shrinkage priors. *Electronic Journal of Statistics*, 11(2):5018 – 5051.

Polson, N. G., Scott, J. G., and Windle, J. (2013). Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349.

Prem, K., Cook, A. R., and Jit, M. (2017). Projecting Social Contact Matrices in 152 Countries Using Contact Surveys and Demographic Data. *PLOS Computational Biology*, 13(9):e1005697.

Prem, K., Liu, Y., Russell, T. W., Kucharski, A. J., Eggo, R. M., Davies, N., Jit, M., Klepac, P., Flasche, S., Clifford, S., Pearson, C. A. B., Munday, J. D., Abbott, S., Gibbs, H., Rosello, A., Quilty, B. J., Jombart, T., Sun, F., Diamond, C., Gimma, A., van Zandvoort, K., Funk, S., Jarvis, C. I., Edmunds, W. J., Bosse, N. I., and Hellewell, J. (2020). The Effect of Control Strategies to Reduce Social Mixing on Outcomes of the COVID-19 Epidemic in Wuhan, China: A Modelling Study. *The Lancet Public Health*, 5(5):e261–e270.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ramos, P. L., Nascimento, D., and Louzada, F. (2017). The long term fréchet distribution: Estimation, properties and its application. *arXiv preprint arXiv:1709.07593*.

Rehg, J. M., Murphy, S. A., and Kumar, S. (2017). Mobile health. *Cham: Springer International Publishing*.

Roberts, G. O. and Tweedie, R. L. (1996). Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, pages 341–363.

Rushworth, A., Lee, D., and Mitchell, R. (2014). A spatio-temporal model for estimating the long-term effects of air pollution on respiratory hospital admissions in greater london. *Spatial and spatio-temporal epidemiology*, 10:29–38.

Rădulescu, A., Williams, C., and Cavanagh, K. (2020). Management strategies in a seir-type model of covid 19 community spread. *Scientific reports*, 10(1):1–16.

Sama, P. R., Eapen, Z. J., Weinfurt, K. P., Shah, B. R., and Schulman, K. A. (2014). An evaluation of mobile health application tools. *JMIR mHealth and uHealth*, 2(2):e3088.

Schwartz, J. (2000). The Distributed Lag between Air Pollution and Daily Deaths. *Epidemiology*, 11(3):320–326.

Scott, J., Hartnett, J., Mockler, D., and Little, M. A. (2020). Environmental risk factors associated with anca associated vasculitis: a systematic mapping review. *Autoimmunity Reviews*, 19(11):102660.

Sievert, C., Parmer, C., Hocking, T., Chamberlain, S., Ram, K., Corvellec, M., Despouy, P., and Inc, P. T. (2020). Plotly: Create Interactive Web Graphics via 'Plotly.Js'.

Sims, C. A. (1971). Discrete approximations to continuous time distributed lags in econometrics. *Econometrica: Journal of the Econometric Society*, pages 545–563.

Soetaert, K., Petzoldt, T., and Setzer, R. W. (2010). Solving Differential Equations in R: Package deSolve. *Journal of Statistical Software*, 33(9):1–25.

Stojan, G., Kvit, A., Curriero, F. C., and Petri, M. (2019). A spatial-temporal analysis of organ-specific lupus flares in relation to fine particulate matter pollution and temperature. *Available at SSRN 3393710*.

Stukel, T. A. (1988). Generalized logistic models. *Journal of the American Statistical Association*, 83(402):426–431.

Tasci, E., Zhuge, Y., Camphausen, K., and Krauze, A. V. (2022). Bias and class imbalance in oncologic data—towards inclusive and transferrable ai in large scale oncology data sets. *Cancers*, 14(12):2897.

Taylor, L. (2021). Covid-19: Brazil's spiralling crisis is increasingly affecting young people. *BMJ*, 373.

Teimouri, A. (2020). An SEIR Model with Contact Tracing and Age-Structured Social Mixing for COVID-19 Outbreak. *medRxiv*, page 2020.07.05.20146647.

Ter Horst, R., Jaeger, M., Smeekens, S. P., Oosting, M., Swertz, M. A., Li, Y., Kumar, V., Diavatopoulos, D. A., Jansen, A. F., Lemmers, H., et al. (2016). Host and environmental factors influencing individual human cytokine responses. *Cell*, 167(4):1111–1124.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Tibshirani, R. J., Taylor, J., Lockhart, R., and Tibshirani, R. (2016). Exact post-selection inference for sequential regression procedures. *Journal of the American Statistical Association*, 111(514):600–620.

Urban, A., Di Napoli, C., Cloke, H. L., Kyselỳ, J., Pappenberger, F., Sera, F., Schneider, R., Vicedo-Cabrera, A. M., Acquaotta, F., Ragettli, M. S., et al. (2021). Evaluation of the era5 reanalysis-based universal thermal climate index on mortality data in europe. *Environmental research*, 198:111227.

Varadhan, R. and Gilbert, P. (2009). BB: An R package for solving a large system of nonlinear equations and for optimizing a high-dimensional nonlinear objective function. *Journal of Statistical Software*, 32(4):1–26.

Wang, X. and Dey, D. K. (2010). Generalized extreme value regression for binary response data: An application to B2B electronic payments system adoption. *The Annals of Applied Statistics*, 4(4):2000 – 2023.

Warren, J. L., Luben, T. J., and Chang, H. H. (2020a). A spatially varying distributed lag model with application to an air pollution and term low birth weight study. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 69(3):681–696.

Warren, J. L., Luben, T. J., and Chang, H. H. (2020b). A spatially varying distributed lag model with application to an air pollution and term low birth weight study. *Journal of the Royal Statistical Society. Series C, Applied statistics*, 69(3):681.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.

Wilson, A., Chiu, Y.-H. M., Hsu, H.-H. L., Wright, R. O., Wright, R. J., and Coull, B. A. (2017a). Bayesian distributed lag interaction models to identify perinatal windows of vulnerability in children's health. *Biostatistics*, 18(3):537–552.

Wilson, A., Chiu, Y.-H. M., Hsu, H.-H. L., Wright, R. O., Wright, R. J., and Coull, B. A. (2017b). Potential for bias when estimating critical windows for air pollution in children's health. *American journal of epidemiology*, 186(11):1281–1289.

Xu, Q., Zhuo, X., Jiang, C., and Liu, Y. (2019). An artificial neural network for mixed frequency data. *Expert Systems with Applications*, 118:127–139.

Xu, X. and Ghosh, M. (2015). Bayesian variable selection and estimation for group lasso. *Bayesian Analysis*, 10(4):909–936.

Yates, M. and Watts, R. (2017). Anca-associated vasculitis. *Clinical Medicine*, 17(1):60.

Yin, S., Dey, D. K., Valdez, E. A., Gan, G., and Vadiveloo, J. (2020). Skewed link regression models for imbalanced binary response with applications to life insurance. *arXiv preprint arXiv:2007.15172*.

Yu, K. and Zhang, J. (2005). A Three-Parameter Asymmetric Laplace Distribution and Its Extension. *Communications in Statistics - Theory and Methods*, 34(9-10):1867–1879.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

Zanobetti, A., Schwartz, J., Samoli, E., Gryparis, A., Touloumi, G., Atkinson, R., Le Tertre, A., Bobros, J., Celko, M., Goren, A., et al. (2002). The temporal pattern of mortality responses to air pollution: a multicity assessment of mortality displacement. *Epidemiology*, pages 87–93.

Zens, G., Frühwirth-Schnatter, S., and Wagner, H. (2020). Ultimate pólya gamma samplers–efficient mcmc for possibly imbalanced binary and categorical data. *arXiv preprint arXiv:2011.06898*.

Zhao, S., Witten, D., and Shojaie, A. (2021). In defense of the indefensible: A very naive approach to high-dimensional inference. *Statistical Science*, 36(4):562–577.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.