Predictive Modelling for Medical Morbidity Risk Related to Insurance

ALEX JOSE

Submitted for SA0 of
The Institute and Faculty of Actuaries

Prepared at

Department of Actuarial Mathematics and Statistics,
School of Mathematical and Computer Sciences

Heriot-Watt University

August 2024

Abstract

The thesis focuses on researching the rates of admission to hospitals (or other health facilities) due to specific illnesses such as cancer and respiratory diseases among a United States working-insured population and their dependence on several demographic and health insurance-related factors. The primary objective is the predictive modelling of these admission rates by employing both traditional methodologies as well as deep learning techniques. As part of this, we develop advanced neural network-based models using a large and complex dataset of admission counts, and compare their predictive performance to that of conventional models.

The admissions dataset is obtained from the Commercial Claims and Encounters Database of Merative MarketScan Research Databases, provided by Merative (US). The dataset contains individual-level information regarding admissions to hospitals and other health facilities, linked with patient information (such as enrollment details) and service provider details. Along with socio-demographic details, the data also consists of geographical information detailing the area of residence.

Among the conventional approaches, classical and Bayesian count regression models with underlying Poisson or negative binomial distributional assumptions are considered for modelling the admission counts data. Neural network embeddings of these regression models are additionally being considered by employing generic Feed Forward Neural Network and specifically designed Combined Actuarial Neural Network approaches. The predictive performance of the network-based models is further enhanced by adopting several model improvement approaches, such as nagging predictors and bias regularization techniques. Furthermore, a k-fold validation process is used to compare the predictive performance of the aforementioned models. The results showcase that the network-based models offer improved predictive performance over traditional regression models in the context of a real-life complex health dataset. The enhanced predictive capability of network-based models, which could be attributed to their capacity to capture potential non-linear interactions within the dataset, often comes at the cost of ease of implementation when compared to conventional models. Additionally, the absence of a statistically robust approach for model selection is also a potential drawback of network-based models.

The neural network modelling approach is further extended to develop models that accommodate the excess zero nature of admissions count data. Towards this, zero-inflated neural network models and zero-inflated combined actuarial neural network models are developed and demonstrate improved predictive performance over the earlier-mentioned models. Moreover, integrating additional GLM-like structure into the neural network models (based on the LocalGLMnet method and its extensions) facilitates the interpretation of outcomes, in a similar way as that obtained from a regression model.

The real-life utility of the work in this thesis lies in the fact that, in addition to facilitating accurate rate setting in the insurance sector, the suite of models and approaches discussed in the thesis can provide precise predictions that have the potential to aid in developing personalised and policy-level healthcare interventions.

Acknowledgements

Firstly, I would like to express my gratitude to my supervisors, Prof. George Streftaris, Dr. George Tzougas, and Prof. Angus S. Macdonald, for their immense support and guidance throughout the years of my research. I am grateful for all their help and motivation, which enabled me to persevere even during the challenging times of the pandemic. Additionally, I appreciate their immense consideration in accommodating my personal circumstances.

I would also like to thank Prof. Andrew J.G. Cairns, as well as other members of the staff in the department, for their assistance, support, and guidance. Furthermore, my gratitude also extends to Prof. Ian Duncan for his substantial support throughout the entire research period, which has been pivotal to the success of this research.

Lastly, but most importantly, I want to express my deepest gratitude to all the individuals who believed in me, supported me, and stood by me, particularly my beloved wife and my family. Her constant encouragement and support have been instrumental throughout the research period, along with the love and support of my family and extended family members.

Data acknowledgement: Certain data used in this study were supplied by Merative as part of one or more Merative MarketScan Research Databases. Any analysis, interpretation, or conclusion based on these data is solely that of the author and not Merative.

Contents

List of Tables

List of Figures

Acronyms

CANN combined actuarial neural network

CCAE Commercial Claims and Encounters

CI critical illness

COBRA Comprehensive Omnibus Budget

CRDs chronic respiratory diseases

EHR Electronic Health Records

FFNN feed-forward neural network

GLMs Generalized Linear Models

ICD-10-CM International Statistical Classification of Diseases and Related Health Problems 10th Revision—Clinical Modification

INLA Integrated Nested Laplace Approximation

LTC long term care

MCMC Markov Chain Monte Carlo

NB negative binomial

PDX Principle Diagnosis Code

ZIP zero-inflated Poisson regression

Chapter 1

Introduction

The term 'morbidity' broadly refers to the state of being ill or the prevalence of a particular illness in a population, and morbidity risk could be interpreted as the risk of falling sick or moving to any other health state from normal well-being. According to the definition of The National Cancer Institute (NCI), morbidity "refers to having a disease or a symptom of disease, or to the amount of disease within a population" (NCI (2021)). The way morbidity risk is perceived varies depending upon the concerned party evaluating the risk.

An actuary would be interested in the incidence/prevalence of a particular disease in a population and its financial implications for the insurance industry, contrary to a clinician or any other health personnel who would be more interested in the deterioration of health. Proper assessment and evaluation of morbidity trends is crucial for insurance companies as this would heavily influence the rates of health insurance and other products such as critical illness (CI) and long term care (LTC). If not appropriately addressed, this could translate to pricing and underwriting risk. Policymakers would also be equally interested in understanding the prevalence and the financial implications since they need this for developing policies and intervention programs. Although the reasons vary, it is clear that quantification of morbidity risk is crucial, which presents a challenge as the morbidity risk is rapidly changing due to various factors such as medical advancements, changes in lifestyle and health

intervention programs.

In this research, our focus is on analysing morbidity by considering the risk associated with admission to a hospital or health facility due to specific illnesses, quantified using rates of admission. In addition to obtaining a deeper understanding of the prevalence and severity of the health condition within the population and identifying the associated key risk factors, admission rates provide insights regarding societal/public health and insurance considerations. Specifically, admission rates offer valuable insights into the overall burden of specific diseases on healthcare systems, allowing for the devising of effective preventive measures as well as more informed resource allocation. Moreover, from an insurance perspective, a nuanced analysis of admission risk/rates contributes significantly to forecasting the financial implications of morbidity trends, thus aiding in insurance product development.

1.1 Motivation and Aims

Noncommunicable diseases (NCDs) or chronic diseases remain one of the leading causes of morbidity and mortality worldwide. The World Health Organisation (WHO) reports that globally, around 41 million people die annually due to chronic diseases, accounting for around 74% of all deaths. Among these, cardiovascular diseases are the leading cause of death (17.9 million), subsequent to cancers (9.3 million) and chronic respiratory diseases (4.1 million) (WHO (2023b)).

In addition to the loss of lives, millions also suffer from different NCDs worldwide, out of which some could be preventable. For example, hundreds of millions of people worldwide suffer from preventable chronic respiratory diseases (CRDs), such as asthma, chronic obstructive pulmonary disease (COPD) and occupational lung disease (Bousquet et al. (2007)). Around 262 million people have asthma globally, and over 3 million people die each year from COPD, which accounts for nearly 6% deaths around the world (WHO (2022b)). The recent COVID-19 pandemic has once again put CRDs in the limelight due to the notion that people with pre-existing

respiratory conditions are at high risk for COVID-related health complications and death. Although some studies, such as Aveyard et al. (2021), support this, owing to the lack of data, the extent of higher risk from COVID for individuals with CRDs is still unclear (WHO (2022a)).

In the case of cancers, which accounted for nearly one-sixth of the global deaths in the year 2020, lung cancer was the leading cause of death, followed by colon and rectum cancer. Nonetheless, in terms of incidence, breast cancer surpassed both the above-mentioned types of cancers (WHO (2023a)). The financial costs incurred due to chronic diseases are also very significant. CDC reports chronic diseases as the leading driver of the Unites state's $4.1 trillion health care expenditure and that cancer care alone is anticipated to cost around $240 billion by the year 2030 (CDC (2023)).

The prevalence of NCDs is attributed to the interaction of various physiological, genetic, behavioural, socio-economic, and demographic risk factors. Even though the incidence of chronic disease might occur later in life, exposure to the risk factors in the early stages of life could add to the risk of contracting the disease later on. Most CRDs and cancers are preventable through early detection and intervention, which require precise identification and understanding of the key risk factors and their association with the incidence and prevalence of these diseases. Past data and studies do provide indications regarding some of the key risk factors leading up to these diseases. For instance, WHO links nearly 77% of the NCD-related death to low- and middle-income countries (WHO (2023b)), highlighting the impact of socio-economic conditions on NCDs. In the case of CRDs, previous studies such as Doney et al. (2014) and CDC (2012) have explored the prevalence of COPD among the US population, whereas Blanc et al. (2019) focused on the relation between workplace exposure and respiratory diseases.

More precise and targeted intervention programs can be devised if a more granular quantification of risk factors is available. Furthermore, this would also enable the insurance companies to accurately price various health insurance products such as

long-term care, disability and critical illness insurance. Thus the granular quantification of the risk of morbidity due to chronic diseases such as cancer and respiratory diseases serves as the primary motivation for this work. We mainly look at this risk from an insurance point of view while also aiding the potential development of intervention programs.

One of the major reasons for the comparatively underdeveloped nature of the field of morbidity risk prediction as compared to mortality risk is attributed to the dynamically changing nature of morbidity trends. The interaction of numerous complex factors, such as changes in life expectancy, improvements in health and care and developments in medical science, contribute towards the rapidly changing morbidity risks and present the timely and pressing need to develop robust predictive models for capturing the relevant impact on medical insurance.

Throughout the thesis, the main focus is on insurance-linked morbidity, evaluated through relevant admission rates to a health facility, and hence we specifically consider an insured population whose morbidity trends might slightly vary from the underlying general population. The principle aim of our research is to develop, evaluate and assess predictive models for morbidity risk and related insurance rates associated with various health insurance products. The core objectives of the thesis are detailed in what follows.

- The development of a suite of statistical predictive models for medical morbidity risks that would enable the accurate prediction of rates of admission to hospitals and other healthcare facilities due to some particular illnesses. Consequently, the models would thus allow for the properly informed and precise pricing of various health insurance products. As part of the model development, various statistical methodologies, both conventional and advanced, were employed. In addition to regression models and their Bayesian adaptions, deep learning methodologies were also utilised for the modelling process. The existing approaches were expanded upon to formulate novel models with more appealing characteristics.

- To construct a suitable and proficient model by utilising the various techniques for feature selection, such as likelihood-based measures, Bayesian variable selection and in-sample model fit.

- To assess and compare the robustness of different models by considering various factors such as predictive performance, complexity, interpretability, ease of implementation and other relevant aspects. In addition to identifying a strong predictive model, the goal is also the comparison of the various underlying modelling methodologies in terms of their advantages and shortcomings. Cross-validation and out-of-sample measures were utilised to evaluate the predictive capability and performance of the different models.

- The quantification of the variability in model results due to the statistical randomness arising from data sampling, parameter estimation and modelling.

- To investigate and understand on a more granular level the impact of various factors such as the nature of employment, geographical and other demographic aspects on the prevalence/incidence of critical illnesses such as cancer and chronic respiratory diseases among the insured, working population in the United States.

1.2 Outline of the Thesis

The rest of this thesis is organised as follows: Chapter 2 provides background and overview of the different modelling methodologies that have been considered. In Chapter 3, we present a detailed description of the data used in this work, and we provide details of the exploratory analysis that was carried out to summarise the main data characteristics. Several data considerations carried out prior to undertaking any modelling approaches are also discussed. The different models developed under classical and Bayesian frameworks for predicting admission rates due to respiratory diseases and cancer are detailed in Chapter 4. In Chapter 5, a detailed description of the different network-based models that were used for modelling the

admission rates is provided. Along with the model fitting process, various model improvement approaches, including bias regularisation and the development of a nagging predictor that was adopted for the network-based modelling, are also presented in the chapter. Furthermore, a comparison between the Bayesian and network-based models is also undertaken in this chapter. In Chapter 6, the neural network methodology is extended to develop interpretable neural network models that also take into account the zero-inflated nature of the underlying admission data set. Finally, conclusions and remarks are provided in Chapter 7.

Chapter 2

Background and Methodologies

2.1 Statistical Learning Methods for Actuarial Problems

Over the last few decades, actuaries have been addressing a multitude of problems in the insurance sector with the help of statistical modelling. Mortality and morbidity rate estimation, pricing, reserving, and risk assessment are some of the more obvious instances from the extensive list of day-to-day tasks addressed by actuaries using statistical approaches. In particular, numerous modelling approaches have been developed for solving various business problems in the life and non-life sectors, driven by the nature of the tasks at hand.

At this point, it is worth noting that for the aforementioned tasks, there has been an increase in complexity due to various reasons, including big data. In addition to the increase in complexity of already existing tasks, recent years have seen actuaries being called upon to address issues that were unattended to previously. For instance, the recent COVID-19 pandemic highlighted the need for addressing the profound impact of the pandemic on mortality and morbidity considerations. Moreover, the ever-evolving business environment, regulatory changes, product innovation and technological advancements have all contributed towards the variations in

the nature of the problems.

Furthermore, newfound emerging risks, such as climate change and cyber risk, are dissimilar to conventional risks and present the need to develop advanced statistical methods. While technological advancement has paved the way for the development of advanced modelling techniques such as deep learning, it has also widened the scope of the problems. Additionally, regarding big data obtained from various sources, technology has altered the gist of the data in terms of source, nature, granularity and, primarily, the amount of data. For example, in the non-life sector, the advent of telematics in motor insurance, and in the life sector, various climate-related hazards impact mortality. In both areas, access to increased computational power and vast amounts of data has allowed for the development and popularity of advanced statistical methods.

Additionally, we emphasise that, due to the aforementioned reasons, the adoption of advanced statistical modelling techniques is evident in actuarial applications as well as elsewhere in other fields of studies and applications. Several such instances exist within the actuarial context where a direct comparison of conventional and latest machine learning methodologies is possible. For example, logistic regression and decision trees for classification problems (Kurt et al. (2008)), Auto-regressive Integrated Moving Average (ARIMA) (Box et al. (2015)) models and Long Short-Term Memory models (LSTM) (Hochreiter and Schmidhuber (1997)) for time series forecasting. For more details regarding the advancement of statistical modelling approaches and the transformation of the underlying modelling routines within the insurance sector, interested readers can refer to Wüthrich and Merz (2023) and Frees et al. (2014).

In the context of health insurance as well, actuaries leverage the aforementioned advanced modelling techniques to support the multifaceted aspects of health insurance. Pitacco (2014) provides an overview of the different health insurance products and the fundamental principles of the actuarial approach to managing health risks while also introducing basic actuarial models. A vast body of literature exists con-

cerning the development of approaches to address highly specific issues within the health insurance context, for which we provide details of some such instances. For example, Christiansen et al. (2018) discusses projection models for health expenses, while Vercruysse et al. (2012) examines the indexing of premiums in lifelong health insurance. In Dodd et al. (2015), the authors investigate the impact of model uncertainty on critical illness insurance pricing, and more recently, Arik et al. (2023) has employed multiple state models for insurance pricing related to breast cancer. For more comprehensive insights into the practical application of advanced modelling techniques in the health insurance context, we refer to the textbook of Duncan (2011).

In particular, regarding the main focus of this thesis, we go through the progressive change in the statistical modelling framework by considering alternative modelling approaches for estimating morbidity rates. As far as this area of research is concerned, historically, models dealing with morbidity rates could be divided into two categories: models focusing on disease dynamics and population models. From an actuarial point of view, we are interested in population models that focus on modelling the prevalence/incidence rate of illness within a specific population and on analyzing how various risk factors affect the rate. Moreover, the increased use and accessibility of Electronic Health Records (EHR) in recent years have facilitated the capture of vital medical information, thereby fostering a significant increase in morbidity risk prediction studies using such records. The availability of EHR has enabled the collection of more detailed and substantial amounts of health information, providing the option to carry out a wide range of morbidity studies even using the same dataset. More specifically, different researchers who are interested in various health-related outcomes can utilise the same data for their respective research purposes. For instance, in our context, we use EHR-based data to examine the frequency of hospital admissions, while someone else might utilise the associated drug-related information for a different research topic. Additionally, EHR offers considerable advantages over conventional cohort-based risk prediction studies, which

span prospectively over a long period and involve significant costs (Goldstein et al. (2017)).

A systemic review of the various such studies was conducted by Goldstein et al. (2017), wherein they categorised the studies based on their characteristics as well as the underlying modelling methodology. They identified GLMs as the most widely used method, followed by Bayesian and network-based approaches. In this work, we specifically employ all three aforementioned statistical modelling frameworks to comprehensively examine the rates of admission to hospitals and other healthcare facilities due to critical illnesses such as cancer and chronic respiratory diseases (CRDs).

From an event rate modelling or survival analysis perspective, actuaries have developed numerous approaches for modelling mortality or morbidity rates, where the event of interest is typically an individual falling sick or experiencing death. Our work centres on regression-based approaches for event rate modelling since analysing the impact of various risk factors on specific morbidities is one of the primary objectives. Additionally, it is worth noting that as one of the first to consider this line of approach, the Cox model is one of the most celebrated models. For more information on the Cox model or the so-called Cox proportional hazard model and survival analysis within the actuarial framework, please refer to Cox (1972), as well as the textbooks by Collett (2023), Andersen et al. (2012) and Macdonald et al. (2018).

Furthermore, in contrast to conventional morbidity studies, where the focus is on the incidence or prevalence of a disease, in our case, the event of interest is an individual's admission to hospitals due to a specific illness. Therefore, we employ a data-driven approach tailored to the admission counts data at hand for developing the different models.

Lastly, we narrow down to the methods within the statistical learning framework that are considered in this thesis, namely the classic regression, Bayesian models

and deep learning models, as well as the enhancement of regression models based on neural networks. The aim is to improve the predictive performance of traditional regression techniques. The comparison of models from distinct methodologies revealed both the capabilities and the limitations of each approach. A comprehensive description of these methodologies is provided in the following sections.

2.2 Regression Analysis of Count Data

Regression analysis is one of the most popular and widely accepted statistical techniques with the fundamental aim of modelling the potential relationship between variables. The general logic is to model a response variable in terms of one or more explanatory variables. Numerous regression techniques exist to accommodate different types of data, purposes and the context of the application (Seber and Lee (2003), Draper and Smith (1998)). Among this, we focus on regression analysis of the count/event data, which in itself is a wide-ranging topic and has a great many applications in different research areas. The usage of discrete parametric distributions for the examination of count data goes a long way back, and Cameron and Trivedi (2013) mention instances as early as the 18th century where deaths in the Prussian army due to mule kicks were analysed by von Bortkiewicz (1898). Since then, the domain has come a long way and a description of this gradual evolution is detailed in Cameron and Trivedi (2013).

One significant development in this area was the introduction of Generalized Linear Models (GLMs) by Nelder and Wedderburn (1972). Thereafter, GLMs have been at the forefront of predictive modelling in all areas of research. In addition to the ease of implementation and explainability, one of the main reasons for the wide acceptance of GLM was the flexibility offered by the link function. The GLMs accommodate different types of response variables via the application of the link function, which vastly extends the scope for utilising GLMs in various situations (Agresti (2015)). Details regarding numerous applications of GLMs for tackling

insurance-related actuarial problems in both life and non-life sectors are described in Haberman and Renshaw (1996), De Jong, Heller, et al. (2008), Frees (2009), and Ohlsson and Johansson (2010).

In the context of count data, a regression model with an underlying Poisson distribution assumption is considered as a standard model or a benchmark for the parametric class of models. Although the characterisations of the Poisson distribution make it an appealing choice for handling count data, some of the properties, such as equidispersion (mean same as the variance), fall short of accurately representing the real-life count data (Cameron and Trivedi (2013)). In reality, the rare event dataset is overdispersed in nature. Also, it has a large number of zeros/zero-inflation, which has motivated the development of negative binomial and zero-inflated regression models. The advancement in computational capabilities and the development of software, such as Bayesian inference using Gibbs sampling (BUGS), have further facilitated the Bayesian implementation of regression models. For instance, in an insurance-related context of morbidity studies, Ozkok et al. (2014a) adapted Bayesian methodologies for modelling claim diagnosis rates of critical illness; more recently, Arık et al. (2021) conducted a population study for assessing the socioeconomic disparities in cancer incidence and mortality in England.

In this work, an ensemble of statistical predictive models is developed that can be used to predict admission rates to hospitals (or other health facilities) related to critical illnesses such as respiratory diseases and cancer in a US population. The main modelling methodologies utilised as part of this work are outlined in what follows.

2.2.1 Poisson GLM model

We first consider a Poisson GLM model for modelling count data in the current context of the admission data set (e.g., Frees (2009); Hardin and Hilbe (2007)). For the Poisson GLM, we assume that for $i = 1, \ldots, n$, n being the number of records in the data set $\mathcal{D}$, admission numbers, Y_i, follow a Poisson distribution with

$$Y_i \sim Poisson(\lambda_i e_i) \tag{2.1}$$

where the mean ($\mu_i^{Pois} = \lambda_i e_i$) depends on the policyholder's characteristics $\boldsymbol{x_i}$ through $\lambda_i = \exp(\boldsymbol{\beta}^\top \boldsymbol{x_i})$, and the exposure e_i which represent the period at risk or the time under observation. By choosing the logarithmic link function, which is in fact the canonical link function for the Poisson GLM, we have a predictor of the form

$$\mu_i^{Pois} : \mathcal{X} \mapsto \mathbb{R}_+, \qquad (o_i, \boldsymbol{x_i}) \mapsto \log(\mu_i^{Pois}) = o_i + \boldsymbol{\beta}^\top \boldsymbol{x_i} = o_i + \langle \boldsymbol{\beta}, \boldsymbol{x_i} \rangle, \tag{2.2}$$

where $\mathcal{X} \subset \mathbb{R}^q$ is the feature space with $\mathbf{x}_i = (\mathbf{x}_{i,1}, \ldots, \mathbf{x}_{i,q})^\top$ giving the feature information. $o_i = \log(e_i)$ is the offset term, and $\boldsymbol{\beta}^\top = (\beta_1, .., \beta_q)$ is the unknown vector of coefficients to be estimated using the iterated weighted least squares (IWLS) technique (Nelder and Wedderburn (1972), McCullagh and Nelder (1989)). The $\langle \boldsymbol{\beta}, \boldsymbol{x_i} \rangle$ represent the inner products of vectors $\boldsymbol{\beta}$ and $\boldsymbol{x}_i$ and are equivalent to $\boldsymbol{\beta}^\top \boldsymbol{x_i}$; both notations are used interchangeably in this thesis. For the models under the Poisson distribution assumption, with y_i being the observation and μ_i the mean, the log-likelihood is of the form

$$l(\boldsymbol{\mu}; \boldsymbol{y}) = \sum_{i=1}^{n} \{y_i \ln(\mu_i) - \mu_i - \ln \Gamma(y_i + 1)\}. \tag{2.3}$$

2.2.2 Negative binomial regression model

In the case of the negative binomial (NB) regression model, admission numbers, Y_i, $i = 1, \ldots, n$, are assumed to follow a NB distribution with a dispersion parameter $\phi > 0$:

$$Y_i \sim NB(\mu_i^{NB}, \phi) \tag{2.4}$$

which has probability mass function

$$f(y|\mu,\phi) = \frac{\Gamma\Big(y+1/\phi\Big)}{\Gamma\Big(y+1\Big)\Gamma\Big(1/\phi\Big)}\left(\frac{1}{1+\phi\mu}\right)^{1/\phi}\left(1-\frac{1}{1+\phi\mu}\right)^{y}, \quad y=0,1,2,... \tag{2.5}$$

were mean and variance is given by

$$\mathbb{E}[Y_i] = \mu_i^{NB} \qquad \text{and} \qquad \mathbb{V}(Y_i) = \mu_i^{NB} + \phi(\mu_i^{NB})^2. \tag{2.6}$$

For a logarithmic link function, the predictor μ_i^{NB} has the form

$$\mu_i^{NB} : \mathcal{X} \mapsto \mathbb{R}_+ \qquad \log(\mu_i^{NB}) = o_i + \langle \boldsymbol{\beta}, \boldsymbol{x_i} \rangle. \tag{2.7}$$

For the NB models, under the `NB2` parameterisation (Hardin and Hilbe (2007)), the log-likelihood is given by

$$l(\boldsymbol{\mu};\boldsymbol{y},\phi) = \sum_{i=1}^{n}\Big\{y_i \ln\Big(\frac{\phi\mu_i}{1+\phi\mu_i}\Big) - \frac{1}{\phi}\ln\left(1+\phi\mu_i\right) + \ln\Gamma\Big(y_i+\frac{1}{\phi}\Big) - \ln\Gamma\Big(y_i+1\Big) - \ln\Gamma\Big(\frac{1}{\phi}\Big)\Big\}. \tag{2.8}$$

Additionally, it is worth noting that the NB distribution reduces to Poisson if $\phi = 0$ (page 85 Cameron and Trivedi (2013)). In other words, Poisson is a limiting or special case of NB distribution. Although regression models with an underlying NB distributional assumption could address the issue of over-dispersion, they fall short of effectively addressing the issue of excess zeros (Gurmu and Trivedi (1996)).

2.2.3 Zero-inflated regression model

The problem of excess zeros magnifies once we start looking at cause-specific events. For instance, in our context, we are interested in admissions to hospitals or health-care facilities, due to specific critical illnesses. Even though many individuals could suffer from specific diseases, admission to hospital implies a severe condition and could be less frequent. This highlights the difficulty and necessity for accurately

modelling admission rates and analysing the impact of different risk characteristics on the medical condition. In addition to the above-mentioned insurance context, numerous instances from other fields of research also deal with zero-inflated data. Irrespective of the area of research, for any studies based on rare events data, examples of which could be found in Ridout et al. (1998), the issue of excess zeros presents a challenge.

Different methodologies have been proposed to handle rare event data, out of which the zero-inflated Poisson regression (ZIP) model, proposed by Lambert (1992), and the hurdle model by Mullahy (1986), are two of the most popular approaches. The underlying principle behind both approaches is similar. In essence, both approaches set forth a mixture distribution with two components; the zero component and the count component. The zero component is used to model the zeros (in case of hurdle models) or the excess zeros (in case of zero-inflated models), and the count component is for the frequency data of the event of interest.

In this thesis, we adopt zero-inflated models over hurdle models, given their enhanced capability to better represent zeros from multiple sources, which is more representative of the admissions data we consider (Hardin and Hilbe (2007)). More specifically, the reasons or circumstances leading to an individual having zero admissions might vary. For example, an individual could also have zero admissions due to factors unrelated to their health status, such as financial constraints, personal preferences, or even a lack of awareness. With the zero component specifically addressing the excess zeros, the zero-inflated model is more indicative of situations where zero admissions originate from various sources. In practice, a count distribution such as the Poisson or NB distribution is used to represent the count component and a Bernoulli distribution is assumed for the zero component. Hence, a zero-inflated Poisson regression model is of the form

$$Y_i \sim \begin{cases} 0 & \text{with probability } \pi_i \\ Poisson(\lambda_i e_i) & \text{with probability } (1-\pi_i) \qquad \text{... for } \; i = 1, ..., n \end{cases} \tag{2.9}$$

with $\mu_i = \lambda_i e_i$ being the mean of the Poisson part of the ith record with exposure e_i and rate parameter λ_i, while π_i represents the probability of zero component which represents the excess zeros. Zeros arise from both the zero component and the count component, with the two components having probability π_i and $(1-\pi_i)$ respectively. Hence, the probability mass function (PMF) of the ZIP mixture distribution is

$$Pr(Y_i = y_i) = \begin{cases} \pi_i + (1-\pi_i)e^{-\mu_i}, & y_i = 0 \\ \\ (1-\pi_i)\frac{e^{-\mu_i}\mu_i^{y_i}}{y_i!} & y_i > 0 \end{cases} \tag{2.10}$$

and the corresponding mean and variance are given by

$$E(Y_i) = (1-\pi_i)\mu_i, \qquad V(Y_i) = (1-\pi_i)\mu_i(1+\pi_i\mu_i). \tag{2.11}$$

The ZIP model allows both μ_i and π_i to be modelled using a set of covariates. This would be of the form :

$$\log(\mu_i) = o_i + \beta_0 + \boldsymbol{\beta}_{reg}^\top \boldsymbol{x_i} = o_i + \beta_0 + \langle \boldsymbol{\beta}_{reg}, \boldsymbol{x_i} \rangle \tag{2.12}$$

and

$$logit(\pi_i) = \gamma_0 + \boldsymbol{\gamma}^\top \boldsymbol{w_i} \tag{2.13}$$

where $logit(\pi_i) = \log(\pi_i/(1-\pi_i))$, $o_i = \log(e_i)$ is the offset term, β_0, γ_0 the intercept terms and $\{\boldsymbol{\beta}_{reg}^\top, \boldsymbol{\gamma}^\top\} = (\beta_1, .., \beta_q, \gamma_1, .., \gamma_s)$ being the unknown vector of coefficients to be estimated corresponding to the sets of covariates $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,q})^\top$ and $\boldsymbol{w}_i = (\boldsymbol{w}_{i,1}, \ldots, \boldsymbol{w}_{i,s})^\top$ considered in the two regression functions with dimensions $q \times 1$ and $s \times 1$ respectively. Although the exposure could be factored in as part of both the predictor functions (Feng (2022)), we employ the general practice of treating it as an offset term in the regression function for μ as discussed in Lee et al. (2001).

Several variants and extensions have been developed ever since, and the application of the same under different contexts has also been considered. Lambert (1992) considered the application to defects in manufacturing, whereas Gurmu (1997) used a semi-parametric version of the hurdle model for Medicaid utilisation. Famoye and Singh (2006) proposed a generalised variant of the zero-inflated model in the context of domestic violence data. In an actuarial setting, Yip and Yau (2005) used zero-inflated models for modelling insurance claim frequency data. Ridout et al. (1998) provide details regarding other similar instances. Additional information regarding the zero-adjusted models for handling data sets with excess zeros,could be found in the textbook of Cameron and Trivedi (2013).

2.3 Bayesian Models

The fundamental divergence of the Bayesian paradigm from the classical/frequentist approach is that it enables parameters to be defined as random variables with an underlying probability distribution. Thus instead of looking at the point estimate and the confidence interval as in a frequentist approach, the Bayesian framework considers the posterior distribution of a parameter. Bayesian inference takes into account not just the data but also the understanding and assumptions about the parameters before observing the data in the form of the prior distribution. Using Bayes theorem, given observed data $\boldsymbol{x}$, and θ being the parameter of the interest, then the posterior distribution of θ is given by:

$$\pi(\theta|\boldsymbol{x}) = \frac{f(\boldsymbol{x}|\theta)p(\theta)}{f(\boldsymbol{x})} \tag{2.14}$$

where, $\pi(\theta|\boldsymbol{x})$ is the posterior distribution of θ, $f(\boldsymbol{x}|\theta)$ is the likelihood, $p(\theta)$ is the prior distribution of θ and $f(\boldsymbol{x})$ is the normalising constant. The normalising constant is given by $f(\boldsymbol{x}) = \int_\theta f(\boldsymbol{x}|\theta)p(\theta)d\theta$, and in most cases normalising constant

is difficult to calculate. Hence in general, the Bayes theorem is normally quoted as:

$$\pi(\boldsymbol{\theta}|\boldsymbol{x}) \propto f(\boldsymbol{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{2.15}$$

omitting the normalising constant and is considered to be central to Bayesian inference (e.g., Gelman et al. (1995), Box and Tiao (1992), Carlin and Louis (2008)). The logic is that prior to data being observed, some preconceptions exist regarding the parameters. These beliefs or understanding about the parameters are updated by combining the information attained from the data (using likelihood) and the prior knowledge and give the updated understanding or posterior distribution of parameters. This can be extended to complex settings of multilevel/hierarchical models and also with potential missing data (Carlin and Louis (2008)).

Although the Bayesian framework has been around for a very long time, the increase in popularity in the past couple of decades is attributed to computational advancements that facilitated the adaptation of iterative sampling-based approaches, as detailed in Gelfand and Smith (1990). As in other research fields, Bayesian techniques have been extensively adapted to tackle various insurance-related actuarial problems. For instance, in Ntzoufras and Dellaportas (2002) and Stephens et al. (2004), Bayesian techniques were used for modelling outstanding claims in the context of motor insurance. Moreover, instances of the adaptation of Bayesian methodologies for morbidity modelling can be found in Streftaris and Gibson (2002), Ozkok et al. (2014b), Arık et al. (2020), and Arık et al. (2021). The Markov Chain Monte Carlo (MCMC) method enables Bayesian inference even for complex situations where it is difficult to analytically calculate the posterior distributions (Gamerman and Lopes (2006)). An overview of the different MCMC techniques, such as Gibbs sampler or Metropolis-Hastings algorithm and its history is detailed in Brooks et al. (2011).

In MCMC, a Markov chain is constructed, with the stationary distribution being equal to the posterior distribution under consideration. The general rationale is that as time t-which represents the discrete time (step) for the discrete time Markov

chain increases, the state of the chain θ^t is updated in such a way that the distribution of the Markov chain converges to the posterior distribution. Once it has converged, the realisations of the chain can be considered as samples from the posterior distribution, which then can be used to determine the empirical estimates of the relevant posterior summaries such as posterior mean and credible intervals (e.g., Brooks (1998), Gamerman and Lopes (2006)). Another approach would be Integrated Nested Laplace Approximation (INLA) (Rue et al. (2009)). The main difference between MCMC and INLA approaches is that MCMCs are based on sampling using simulations, whereas the INLA method uses numerical approximations.

In our work, we are specifically interested in the Bayesian regression models for count data. In addition to the Bayesian Poisson regression model, we also employ the hierarchical Poisson-Gamma model and Poisson Log-normal model. The motivation for considering hierarchical models is that it would address the heterogeneity and the associated over-dispersion in a count data set (Congdon (2019)).

2.3.1 Bayesian Poisson model

In a Bayesian Poisson model, we assign vague normal priors to the model parameters. I.e the model structure is of the form $Y_i|\lambda_i \sim Poi(\lambda_i e_i)$, for $i = 1, .., n$, where $\lambda_i = \exp(\boldsymbol{\beta}^\top \boldsymbol{x})$ with β coefficients having independent vague normal priors ($\beta_k \sim N(0, 10^4)$, for $k = 1, .., q$). Contrary to this, in the case of the Poisson-Gamma and Poisson Log-normal model, a prior assumption is also applied to the rate (λ) parameter, specifics of which is included here.

2.3.2 Bayesian Poisson-Gamma model

The model structure is as follows:

$$Y_i|\lambda_i \sim Poi(\lambda_i e_i), \qquad \text{for } i = 1, .., n \tag{2.16}$$

where

$$\lambda_i|\phi, \xi_i \sim Gamma(\phi, \frac{\phi}{\xi_i}). \tag{2.17}$$

$\xi_i = \exp(\boldsymbol{\beta}^\top \boldsymbol{x})$ with precision parameter ϕ having a vague $Gamma(1, 0.001)$ prior and $\beta_k \sim N(0, 10^4)$ for $k = 1, .., q$. This is equivalent to Y_i having a NB distributional assumption of the form $Y_i \sim NB(\xi_i, \phi)$ (Congdon (2019)).

2.3.3 Bayesian Poisson Log-normal model

A Bayesian Poisson Log-normal model is given below.

$$Y_i|\lambda_i \sim Poi(\lambda_i e_i), \qquad \text{for } i = 1, .., n \tag{2.18}$$

where

$$\log(\lambda_i)|\mu_i, \sigma^2 \sim N(\mu_i, \sigma^2) \tag{2.19}$$

with $\mu_i = \boldsymbol{\beta}^\top \boldsymbol{x}$ and $\beta_k \sim N(0, 10^4)$ for $k = 1, .., q$. A vague prior is also defined for the precision parameter $\tau = (1/\sigma^2)$ as $\tau \sim Gamma(1, 0.001)$ (Congdon (2019), Ntzoufras (2011)). By defining a vague prior, the model relies more on the likelihood and thus the data for estimating the posterior (Lunn et al. (2012)).

In addition to the Bayesian and non-Bayesian models mentioned in the previous sections, in this thesis, we have also employed machine learning approaches for modelling the admission counts. An overview of the adapted methodologies is provided in the next section.

2.4 Machine Learning

The advancing field of deep learning, a subset of artificial intelligence, embodies a modern approach to designing and training neural networks (NNs)-computerised systems inspired by the human brain that learn from complex data to make predictions or decisions. This methodology has garnered remarkable successes in fields like computer vision, natural language processing, and speech recognition. Improve-

ments in the ease of access to fast computing over the last few decades have paved the way for the extensive adaptation of deep learning approaches in many research and practice areas. Furthermore, it is attracting growing attention within the actuarial community in recent years, as evidenced by works both in the academic research field (Wüthrich and Merz (2019), Denuit et al. (2019), Wüthrich (2020), Denuit et al. (2021) and Wüthrich and Merz (2023)) and in the insurance business sector. There is an increasing trend in applying deep learning approaches for addressing various insurance-specific problems. Blier-Wong et al. (2020) provided a detailed description of recent adaptions of different machine learning approaches in the actuarial field, particularly for pricing and reserving. More specifically, Ferrario et al. (2020) detailed the adaptation of neural network regression models for modelling claim rates using the French motor third-party liability (MTPL) insurance data set. The approach facilitates the embedding of traditional regression models for count data into a neural network framework using a class of feed-forward neural network (FFNN) models. For the rest of the thesis, we refer to the FFNN and neural network (NN) indifferently. The approach was further extended by Schelldorfer and Wuthrich (2019) to develop the combined actuarial neural network (CANN) approach (see Section 5.2.2 for details), which showed superior performance in comparison to the previous NN model. Tzougas and Li (2021) added to the approach by developing both NN and CANN models under a NB distributional assumption.

An ensemble of network-based models has been developed as part of the work carried out as part of the thesis. The fundamental building configuration of all these models is a feed-forward neural network structure. Generally, a FFNN comprises an input layer, one or more hidden layers, and an output layer. The feature space $\mathcal{X}$ makes up the input layer, and each of the hidden layers comprises a set number of neurons. The output from a given hidden layer acts as the input for the next layer. The output from a neuron depends on the linear combination of the output from the previous layer and the choice of activation function assigned to the layer that it is part of (for more details on activation function, see Section 5.3). The number of

hidden layers $d \in \mathbb{N}$ is treated as a hyper-parameter and is also referred to as the depth of the network. The last layer of the architecture, which is connected to the last hidden layer, is the output layer. In a neural network architecture, each layer is a function of the previous layer (see LeCun et al. (2015) and Ferrario et al. (2020) for more details). The m^{th} hidden layer $\mathbf{z}^{(m)}, 1 \leq m \leq d$ with dimension $q_m \in \mathbb{N}$ can be defined as

$$\mathbf{z}^{(m)} : \mathbb{R}^{q_{m-1}} \to \mathbb{R}^{q_m}, \quad \mathbf{z} \mapsto \mathbf{z}^{(m)}(\mathbf{z}) = (1, z_1^{(m)}(\mathbf{z}), \ldots, z_{q_m}^{(m)}(\mathbf{z}))^\top, \tag{2.20}$$

inclusive of the intercept component, where the neurons $z_j^{(m)}, 1 \leq j \leq q_m$, are given by

$$z_j^{(m)}(\mathbf{z}) = \psi\left(\langle \boldsymbol{\beta}_j^{(m)}, \mathbf{z} \rangle\right), \tag{2.21}$$

with $\psi : \mathbb{R} \to \mathbb{R}$, being the activation function and $\boldsymbol{\beta}_j^{(m)} \in \mathbb{R}^{q_{m-1}+1}$ the network parameters. The hidden layer, $\mathbf{z}^{(m)}$ depends on the network parameters $\left(\boldsymbol{\beta}_1^{(m)}, \ldots, \boldsymbol{\beta}_{q_m}^{(m)}\right) \in \mathbb{R}^{q_m}$. For $q_0 = q$, with q being the dimension of the feature space $\mathcal{X}$, the network parameter $\boldsymbol{\beta} = \left(\boldsymbol{\beta}_1^{(1)}, \ldots, \boldsymbol{\beta}_{q_d}^{(d)}, \boldsymbol{\beta}^{(d+1)}\right) \in \mathbb{R}^r$ will have dimension r where

$$r = \sum_{m=1}^{d} q_m \left(q_{m-1} + 1\right) + (q_d + 1). \tag{2.22}$$

A diagrammatic representation of a feed-forward neural network with three hidden layers with 20,15,10 neurons in each layer is shown in Figure 2.1. The risk features are the covariates in the context of a GLM model, and the model outcome or response (Y) is obtained by adding the output from the last hidden layer ($Y0$) and the offset term o_i, which is $\log(e_i)$.

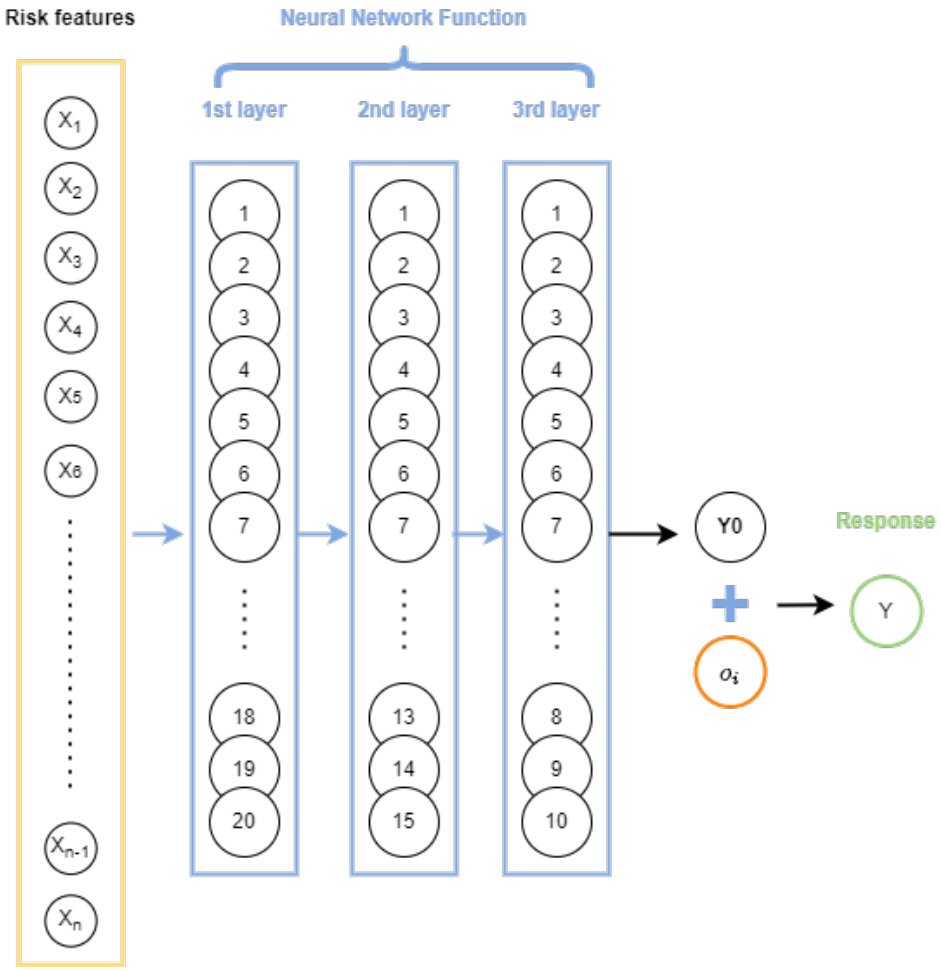


Figure 2.1: A sample representation of a feed-forward neural network with three hidden layers with 20,15,10 neurons in each layer.

Under a neural network regression model, the predictors μ^{Pois} and μ^{NB} of the traditional Poisson and NB regression models are replaced by the neural network predictors μ^{PoisNN} and μ^{NBNN}. To illustrate the structure of the network-based model, we will refer here to the models under the Poisson distributional assumption. For an FFNN model with depth d under the Poisson assumption, the predictor is of the form

$$(o_i, \mathbf{x}_i) \mapsto \log(\mu_i^{PoisNN}) = o_i + \langle \boldsymbol{\beta}^{(d+1)}, (\mathbf{z}^{(d)} \circ \cdots \circ \mathbf{z}^{(1)})(\mathbf{x}_i) \rangle, \tag{2.23}$$

for $i = 1, \ldots, n$, where $\boldsymbol{\beta}^{(d+1)} \in \mathbb{R}^{q_d+1}$ are the weights that map the neurons of the last hidden layer $\mathbf{z}^d$ to the output layer $\mathbb{R}_+$ and $o_i = \log(e_i)$ is the offset term.

The approaches discussed in this chapter lay the foundation for the collection of models developed as part of this work. In addition to the development of various

predictive models, one of the primary objectives, as outlined in Chapter 1, is the comparison of the models by considering numerous aspects of the model. The predictive performance of a given model needs to be evaluated in conjunction with the other attributes, such as complexity, ease of implementation etc and not as a separate silo. The fact that the various models are developed within distinct modelling frameworks makes the comparing process tedious. The approaches used for the comparison of different models, such as K-fold validation, are detailed later on.

The effectiveness of any undertaken modelling methodology heavily depends on the data at hand in terms of its quality and quantity. The data that we consider is the MarketScan health data for the US population. A detailed description of this data set, as well as the data pre-processing and the exploratory analysis that was undertaken, is detailed in the next chapter.

Chapter 3

Data

The extent of high-quality morbidity studies has always been limited due to the lack of flawless health data and access restrictions arising from the sensitive nature of health data. In general, the primary source of health data for actuarial studies is either claims data or medical records from hospitals and clinics. The depth and validity will vary for different data sets and are also influenced by various data practices. This chapter discusses the IBM data set, which is used as the primary source of data for the work done as part of this thesis.

3.1 MarketScan Data

The Merative (formerly IBM Watson Health) MarketScan Research Data Bases is a group of databases containing individual-level claims data obtained from various health plans, employers and organisations. It contains individual-level paid claims and encounter data over time linked with patient information and service provider details. The encounter data differs from claims data as the financial information is not accurate since it comes from health plans that work on a capitation basis and not on a reimbursement basis. The raw data thus obtained from various sources are reviewed by Merative to ensure quality and completeness. After conducting a data quality review, the data provider standardises and cumulates the data from

multiple sources. The data are then made available through the database tables constructed as shown in Figure 3.1. Merative MarketScan Research Data Bases comprise multiple databases, out of which we will focus on the Commercial Claims and Encounters (CCAE) database.

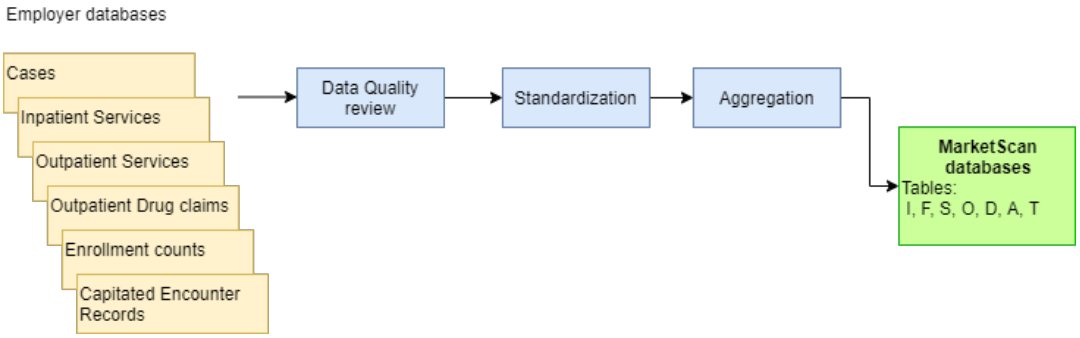


Figure 3.1: Description of the procedure behind the construction of MarketScan databases (MarketScan (2017)).

3.1.1 Commercial Claims and Encounters Database

The CCAE database comprises different tables with data from multiple employers and health insurance plans in the US. It contains medical and drug data of active employees, their dependents, early retirees, and Comprehensive Omnibus Budget (COBRA) continuees within the age range 0 to 65. Along with the coverage information, it also contains details (medical and financial) regarding the services used by these individuals under various plans that are either fee-for-service or fully or partially capitated plans. In the case of a fully or partially capitated plan, an encounter record is created instead of a claim record, and the financial information for these encounter records is limited. The database contains details about the claims, encounters and the services associated with it and has the structure as detailed in Figure 3.2. A detailed description of the different tables within the CCAE database is given in the next section.

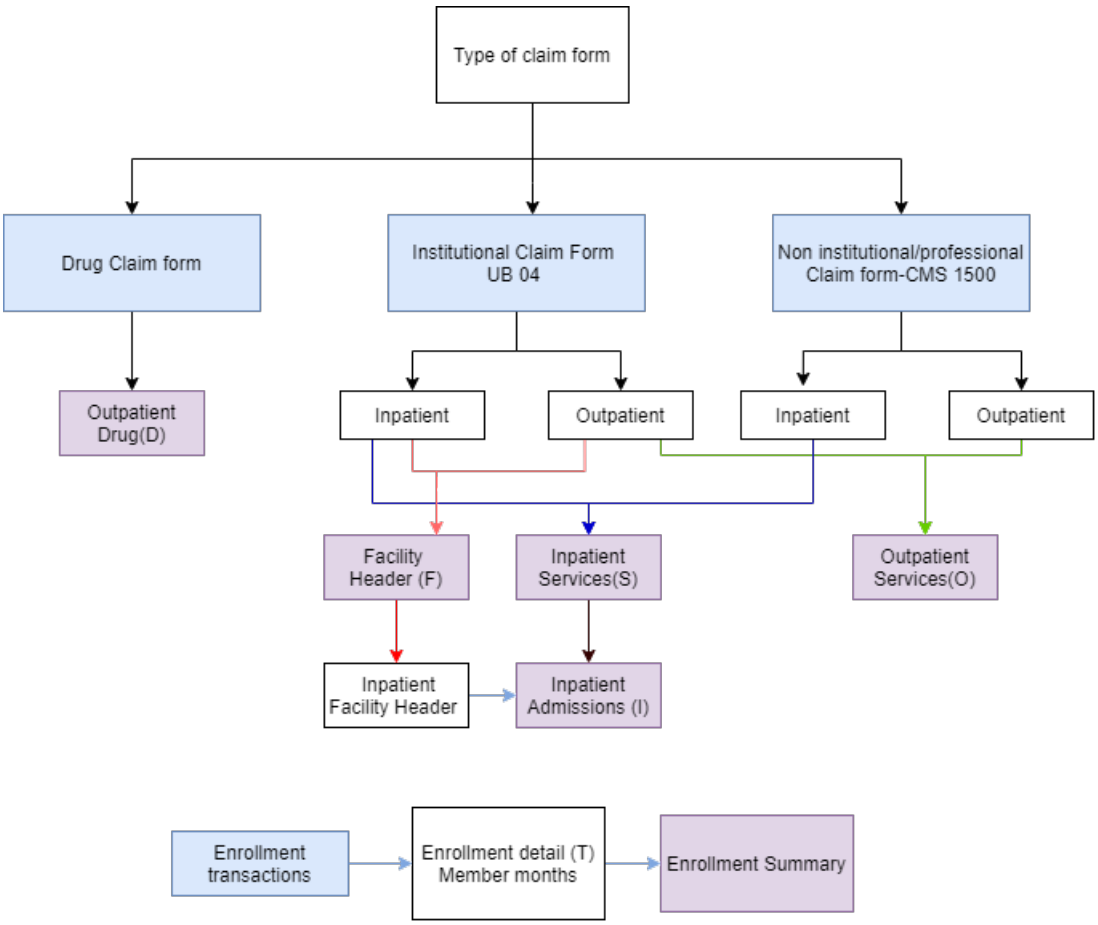


Figure 3.2: A schematic representation of the data flow between the the different tables within the commercial claims and encounters database (MarketScan (2017)).

3.2 Data Overview

As mentioned in the previous section, the CCAE database is made up of different interconnected tables, and for each of these tables, separate yearly data files exist. Table 3.1 shows the number of individuals whose information is available for the calendar years 2016 through 2019.

Year	# of individuals
2016	3,411,832
2017	3,106,063
2018	4,464,494
2019	3,841,727

Table 3.1: Number of individuals within the CCAE database in each year.

- **Inpatient Admissions Table (I):** contains the summarised information about admissions to hospitals and other healthcare facilities and is useful if the interest is in the number of admissions. Each admission record contains summarised details regarding all the claims and services associated with a single admission. A grouping of services is considered an admission only if it meets certain criteria, such as the presence of a room and board claim. If not, it is added to the outpatient service table.

  Apart from financial variables, the table also contains admission-related details such as Major Diagnostic Category (MDC), diagnosis-related group (DRG), date of admission, length of stay etc... It also contains other enrollment and individual-related details such as age, sex, year-of-birth (YOB), family identification code and plan type. Table A.1 gives the complete list of variables and their descriptions.

- **Facility Header Table (F):** This table contains the header information from each facility claim form UB-04. In other words, it only contains information regarding institutional claims and is helpful if the focus is only on institutional claims. The claims from facilities such as hospitals, nursing facilities and other institutional providers are called institutional claims, whereas claims from physicians, nurses, suppliers and other healthcare professionals are considered professional/non-institutional claims. Different claim forms are used in each case, and the detailed information associated with each claim will be there in Inpatient Services (S) or Outpatient Claims (O) tables.

- **Inpatient Services Table (S):** It contains all the details about all claims, encounters (facility/professional) and services associated with an admission record in admissions table(I).

- **Outpatient Services Table (O):** The details of claims and encounters in any outpatient facility, emergency department or doctor's office are included in this table. Some of the records in this table are from inpatient services.

These are the records that did not meet the criteria to be considered as an admission.

- **Outpatient Pharmaceutical Claims Table (D):** This table contains the pharmaceutical claims data associated with the individuals in the medical/-surgical (I, F, S, O) tables. Drug data are unavailable for all the people in the enrollment file, and the RX variable indicates whether the drug data are available for a particular individual.

- **Annual Enrollment Summary Table (A) & Enrollment Detail Table (T):** These tables contain the enrollment details of the individuals. Along with the personal details, it also gives coverage type and duration. The annual Enrollment Summary Table contains summarised annual enrollment details, whereas the Enrollment Detail Table has multiple entries for an individual (on a monthly basis). The Enrollment Detail Table was considered instead of the Annual enrollment table to take into account the potential change in an individual's risk characteristics during any given year. The individual-level socio-demographic information includes age, gender, employment-related information, and geographic details regarding the area of residence. The complete list of variables, along with a detailed description, is given in Table A.2.

Among these different tables, the focus is on the inpatient admissions table and the enrollment detail table. The admission counts data set used in this thesis was constructed by combining the admission information and the enrollment details for individual years from 2016 through 2019.

3.3 Admission Counts Data

The data preparation and the additional data considerations undertaken as part of the data pre-processing for creating the admission counts data are detailed in this section. The aim is to create a refined admission counts data set by combining relevant details from various tables within the CCAE database, which will be used

for modelling purposes.

3.3.1 Data preparation

The initial focus was on data cleaning and preparation, which was carried out as a multi-step process. Firstly, we calculated the exposure from the enrollment table. Secondly, we calculated the admission counts, and finally, the admission details were combined with the information from the enrollment table.

- Calculating the exposure from the enrollment table: as mentioned earlier, the circumstances of some individuals change over the year, and hence, their risk characteristics change as well. Hence, it is crucial to calculate the period for which the individual remained under a particular risk profile. Multiple records with different risk characteristics exist for those individuals for the corresponding period. This results in having more records than the number of unique individuals. Although multiple records might exist for some individuals, the sum of the exposure over different records for any given individual is less than or equal to one year. The duration or number of days for which the individual's risk characteristics remained unaltered is given by the MEMDAYS variable and is used to calculate the yearly exposure. This implies that the exposure is calculated by dividing the MEMDAYS by the number of days in the year.

- Calculating the admission counts: number of admissions is calculated by identifying the records with unique ENROLID and CASEID. Each unique CASEID represents an admission, and the primary cause of each is attained from the Principle Diagnosis Code (PDX) variable. An alternative option was the Major Diagnostic Category (MDC) variable which contains disease or body system-related grouping assigned by Watson Health to clinical conditions using the diagnosis codes. The issue is that the MDC variable does not provide detailed information regarding the cause of diagnosis. PDX uses the International Statistical Classification of Diseases and Related Health Problems 10th

Revision—Clinical Modification (ICD-10-CM) codes, and this is used for categorising the admissions with respect to the primary cause of admission. The ICD-10-CM codes are categorised into Chapters as given in Table 3.2, and the first three characters of a diagnosis code give the chapter details. The principal diagnosis code of an admission is mapped with this chapter information to identify the primary cause, and thus the number of admissions each individual had under each chapter is estimated.

Chapter	Codes	Title
1	A00–B99	Certain infectious and parasitic diseases
2	C00–D49	Neoplasms
3	D50–D89	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
4	E00–E89	Endocrine, nutritional and metabolic diseases
5	F01–F99	Mental and behavioral disorders
6	G00–G99	Diseases of the nervous system
7	H00–H59	Diseases of the eye and adnexa
8	H60–H95	Diseases of the ear and mastoid process
9	I00–I99	Diseases of the circulatory system
10	J00–J99	Diseases of the respiratory system
11	K00–K95	Diseases of the digestive system
12	L00–L99	Diseases of the skin and subcutaneous tissue
13	M00–M99	Diseases of the musculoskeletal system and connective tissue
14	N00–N99	Diseases of the genitourinary system
15	O00–O9A	Pregnancy, childbirth and the puerperium
16	P00–P96	Certain conditions originating in the perinatal period
17	Q00–Q99	Congenital malformations, deformations and chromosomal abnormalities
18	R00–R99	Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
19	S00–T88	Injury, poisoning and certain other consequences of external causes
20	V00–Y99	External causes of morbidity and mortality
21	Z00–Z99	Factors influencing health status and contact with health services

Table 3.2: Chapter wise grouping of diagnosis codes for ICD-10 (CDC (2016)).

- Combining the enrollment and admission details: prior to collating the information from the two tables, the relevant fields from both tables are identified. Some of the variables are excluded due to a high level of granularity. For instance, the Metropolitan Statistical Area (MSA) variable is not included,

but an alternative Urbal-Rural (UR) indicator variable is derived using the information from the MSA variable. The logic is that if the individual resides in a non-MSA region, then the corresponding area of residence is taken as rural (UR=1) else as urban (UR=2). Also, for variables that are common to both tables, the information from the enrollment table is used. Furthermore, some financial and medical variables are also excluded as the focus is on the number of admissions. A detailed description of included and excluded variables from both enrollment and admission tables is given in Tables A.1 and A.2. Additionally, the admissions with unclear diagnosis codes and without the enrollment data are excluded. Each admission record is linked with the individual's enrollment details using the ENROLID (unique individual level identifier) variable and is ensured that the admissions' date of admission (ADMDATE) falls between the exposure start date and end date of the individual. The number of admissions excluded from each year as part of the data preparation process is shown in Table 3.3.

Year	**# of admissions**	**w/o enrollment data**	**w/ missing or unclear codes**	**# of admissions included**
2016	165,781	10,106	50	155,625
2017	149,091	6,991	22	142,078
2018	203,364	5,837	20	197,507
2019	173,148	6,072	34	167,042

Table 3.3: Number of admissions in each year after the data preparation process.

The combined data set thus constructed contains individual-level socio-demographic data and the number of admissions the individual had under each chapter in the ICD-10-CM. The list of variables within the combined data set and their characteristics is given in Table 3.4.

Variable	Description	Comment	Categories
ENROLID	Unique id for individual	ID variable	-
AGE	Age last birthday of the individual	$\in [30, 65]$	-
SEX	Gender of the individual	Factor w/2 categories	1: Male, 2: Female
EMPREL	Relation to the primary beneficiary	Factor w/3 categories	1: Employee, 2: Spouse, 3: Child/Other
PLANTYP	Type of health plan individual is part of	Factor w/9 categories	1. Basic/Major Medical Plan, 2: Comprehensive Plan, 3: Exclusive Provider Organization Plan, 4: Health Maintenance Organization Plan, 5: Non-Capitated (Non-Cap) Point-of-Service, 6: Preferred Provider Organization Plan, 7: Capitated (Cap) or Partially Capitated (Part-Cap) Point-of-Service Plan, 8: Consumer-Driven Health Plan, 9: High-Deductible Health Plan
REGION	Geographical region of residence	Factor w/5 categories	1: Northeast, 2: North Central, 3: South, 4: West, 5: Unknown
EGEOLOC	Geographic location based on postal code of individual's residence	Factor w/53 categories	See Table A.3
UR	Urban/ rural indicator based on individual's residence	Factor w/2 categories	1:Rural, 2:Urban
EECLASS	Employee classification	Factor w/9 categories	1: Salary Non-union, 2: Salary Union, 3: Salary Other, 4: Hourly Non-union, 5: Hourly Union, 6: Hourly Other, 7: Non-union, 8: Union, 9: Unknown
EESTATU	Status of employment	Factor w/9 categories	1: Active Full Time, 2: Active Part Time or Seasonal, 3: Early Retiree, 4: Medicare Eligible Retiree, 5: Retiree (status unknown), 6: COBRA Continuee, 7: Long Term Disability, 8: Surviving Spouse/Depend, 9: Unknown
INDSTRY	Industry were the primary beneficiary is employed in	Factor w/10 categories	1: Oil & Gas Extraction, Mining, 2: Manufacturing, Durable Goods, 3: Manufacturing, Nondurable Goods, 4: Transportation, Communications, Utilities, 5: Retail Trade, 6: Finance, Insurance, Real Estate, 7: Services, A: Agriculture, Forestry, Fishing, C: Construction, W: Wholesale

Table 3.4: Description of variables in the admission counts data set.

Variable	Description	Comment	Categories
HLTHPLAN	Whether the data are provided by employer or a health plan	Factor w/2 categories	0: Employer, 1: Health plan
DATATYP	Whether the plan is on reimbursement or capitation basis	Factor w/2 categories	1: Fee for service, 2: Encounter
EXPOSURE	Period of enrollment- yearly exposure	$\in (0, 1]$	-
YEAR	Calendar year	Factor w/4 categories	$\in [2016, 2019]$
CH01-CH021	Number of admission for the individual under each ICD chapter	$\in \mathbb{R}^+$	-

Table 3.4: *Cont.* Description of variables in the admission counts data set.

3.3.2 Data description

The ENROLID is a unique identifier assigned to each individual. The EMPREL variable specifies the individual's relationship to the primary beneficiary/employee. The different plan types defined by the PLANTYP variable vary in terms of their characteristics, such as incentives for using specific service providers, deductibles, copay, etc.... For more details regarding the different plan types, see Table A.4. The geographical variable EGEOLOC gives more granular information regarding the primary beneficiary's residence location than the REGION variable. The UR variable was created using the Metropolitan Statistical Area (MSA) variable by assigning value 1 (rural) if the primary beneficiary resides in a non-MSA or rural area and value 2 (urban) if the primary beneficiary resides in MSAs. The EECLASS and EESTATU variables give information regarding the employment of the individual. The DATATYP indicates whether the individual's health plan operates as a fee-for-service plan or a capitation plan. The difference is that the fee-for-service works on a reimbursement basis, while the encounter record arises from fully or partially capitated managed care plans. The prepaid capitation amount paid by the employer or the health plan to the service provider could be on an individual or a bulk basis. The INDSTRY variable contains information regarding the employer's industry classification but is not an indicator of the individual's work conditions. For in-

stance, an individual could have a desk job at a company in the agriculture sector. The CH01-CH21 fields give the number of admissions for an individual under each chapter during that corresponding enrollment period. The details of the primary beneficiary are assigned to the spouse and other dependents as well. Furthermore, several data considerations and feature pre-processing were undertaken to construct the final data set, details of which are described in the following subsection.

3.3.3 Additional data considerations

Preliminary exploratory analysis revealed some issues within the data set that need to be addressed. For instance, missing data, the association between data variables and the unusually high number of admissions could arise due to readmission or the common healthcare data practices.

3.3.3.1 Readmission

It is reasonable to expect that an individual with chronic conditions may experience multiple admissions over the course of time, and the time span between admissions would vary from days to months. The frequency of admissions per individual (see Table 3.5) showed instances of individuals having more than 20 admissions in a year. Admissions of this nature could be treated as from a single episode of illness if the focus is on episodes of illness.

# of admissions	# of individuals in each year			
	2016	**2017**	**2018**	**2019**
0	3,286,147	2,991,521	4,306,103	3,708,223
1	107,199	97,748	134,359	112,985
2	12,755	11,464	16,484	14,153
3	3,206	2,975	4,257	3,497
4	1,216	1,117	1,604	1,385
5	575	538	747	650
6	313	304	368	338
7	175	149	223	178
8	92	89	114	107
9	51	51	79	65
10	30	37	62	52
11	32	25	32	26
12	9	14	19	26
13	6	9	12	14
14	6	7	10	7
15	5	4	3	3
16	4	3	5	9
17	6	3	3	1
18	2	2	5	3
19	1	2	-	4
20	1	1	3	-
21	-	-	-	1
22	-	-	2	-
23	1	-	-	-
Total	**155,625**	**142,078**	**197,507**	**167,042**

Table 3.5: Frequency of admissions per individual in each year.

Furthermore, the admission records in the Merative data set are influenced by the administrative coding/filing practices of the data providers and the hospitals in-

volved. For instance, an individual has separate admission records within a short period. In some cases, individuals have the date of admission being the same as the discharge date of their previous admission record. This implies that the person got discharged and readmitted on the same day, which is rare to occur in real life. These sorts of instances could lead to inflated admission counts. This demanded developing criteria for defining an admission. In order to address this, the following approach was adopted:

- For any given individual, if a new admission record is within a specified period from an earlier admission and under the same ICD chapter, it is treated as part of the same admission. In other words, if the difference between the discharge date on the previous admission record and the admission date on the new record under the same ICD chapter is less than a specified period, then it is treated as a single admission.

As the choice of period defining an episode/admission would impact the number of admissions, different choices were considered. Table 3.6 shows the variation in the number of admissions in 2016 depending on the period used for defining an admission. Based on expert opinion, the choice of this period used to group the records under the same admission is taken as two days. Hence the 155,625 records of admissions are aggregated into 152,610 admissions. The same criteria were applied to the data from other years as well.

period defining an admission	same day	1 day	2 days	3 days	4 days	5 days	6 days	7 days	14 days	30 days	90 days
# of records under a single admission	frequency										
1	152,039	150,846	149,830	149,009	148,349	147,767	147,175	146,589	143894	140,063	134,786
2	1,681	2,181	2,598	2,948	3,217	3,437	3,680	3,908	4830	5,945	7,324
3	61	109	152	178	202	234	256	281	462	741	1,161
4	9	17	22	27	33	39	43	54	105	216	348
5	1	2	3	3	7	10	14	15	24	56	114
6		2	2	3	4	4	4	5	9	24	47
7			2	3	3	4	2	2	3	9	33
8											6
9							2	2	4	1	4
10				1	1	1	1	1	2	3	6
11										3	3
12										1	1
13											
14			1	1	1	1	1	1	1	1	1
15											2
16											
17											
18											
19											
20											
# of admissions	153,791	153,157	152,610	152,173	151,817	151,497	151,178	150,858	149,334	147,063	143,836

Table 3.6: Number of admissions in the year 2016 depending on the period used for defining an admission

3.3.3.2 Missing data

Another issue was the missing values and the 'Unknown' category for some variables. Although, in effect, in both cases, the information is not available, the underlying reason causing this might be different. Hence the 'Unknown' values were distinguished from the missing values and were treated as an additional level in the categorical variables. Regarding missing data, three variables, PLANTYP, UR and INDSTRY, had the information missing in all four years. Table 3.7 shows the number and proportion of missing values for each of these variables in all the years.

Year	# of records	# of records w/ missing data			% of records with missing data		
		PLANTYP	UR	INDSTRY	PLANTYP	UR	INDSTRY
2016	4,249,517	72,415	16,047	1,562,173	1.7%	0.4%	36.8%
2017	3,904,094	70,420	424,069	1,218,259	1.8%	10.9%	31.2%
2018	5,611,541	152,533	536,255	1,902,729	2.7%	9.6%	33.9%
2019	4,773,929	80,189	494,753	1,689,882	1.7%	10.4%	35.4%
Total	18,539,081	375,557	1,471,124	6,373,043	2.0%	7.9%	34.4%

Table 3.7: Year-wise proportion of missing values for the PLANTYP, UR and INDSTRY variables.

Among these three variables, the INDSTRY variable was excluded as the proportion of missing data was considerably high. For the PLANTYP variable, the proportion of missingness was 2%, and for the UR variable, it was 7.9%. For the UR variable, the proportion of missing data in 2016 was considerably low compared to other years. As mentioned previously, the MarketScan data are influenced by the data practices of more than 300 data providers, so the nature of the missing data is not directly evident, making it difficult to adopt particular data imputation-related approaches to addressing it. In general, it is quite plausible that missing data could lead to potential bias if the data missingness is not random (Little and Rubin (2019)). However, in our context, we found no evidence pointing towards data missingness being not random, so we proceed with a complete case analysis approach, which is commonly adopted in similar circumstances. It is also worth noting that, where data are missing at random and the proportion of missing data is low, complete case analyses can be unbiased (Ibrahim et al. (2005)). Therefore, we proceed by excluding the records with missing values for PLANTYP and UR variables.

3.3.3.3 Association between variables

The HLTHPLAN and DATATYP variables were excluded due to the high level of relationship with other variables. The HLTHPLAN variable had a high level of association with the EECLASS and EESTATU variables. The employment-related information, particularly the employment classification, was only available if the data came from the employer (HLTHPLAN = 0). Hence, for all the individuals

whose data came from health plans, the EECLASS was unknown (category 9). A high level of association was observed between PLANTYP and DATATYP variables as well. The association between PLANTYP and DATATYP arises since the value of the DATATYP variable is 2 (Encounter) only for individuals under the Health Maintenance Organization plan (PLANTYP = 4) and Capitated (Cap) or Partially Capitated (PartCap) Point-of-Service plan (PLANTYP = 7). Although geographical variables REGION and EGEOLOC showed a strong association, both variables were retained and will be selected based on the required level of granularity.

Additionally, for the PLANTYP variable, a minor variation was observed regarding its composition between the data from different years. Records with the plan type being 'Basic/major medical' (PLANTYP=1) were only present in 2019 in a nominal proportion. Due to the similarity between the characteristics of 'Basic/major medical' and 'Comprehensive Plan' (PLANTYP=2) (see Table A.4), those records were combined with the Comprehensive plan by replacing the value of the PLANTYP variable.

3.4 Descriptive Statistics and Exploratory Analysis

The refined data set constructed by means of the multi-step data pre-processing process comprises 16,710,601 records with 6,530,341 unique individuals. A detailed exploratory analysis was conducted to better understand the data and its inherent characteristics. The age range selection of 30 to 65 was used as the focus is on the working population, and 30–65 seemed to be a reasonable choice for working age. Table 3.8 shows the overall summary of the data in terms of the number of records, individuals, exposure and admissions for each year.

Year	2016	2017	2018	2019	Total
Age range 0-65					
# of records	4,162,532	3,416,383	4,928,629	4,203,057	16,710,601
# of individuals	3,345,254	2,715,979	3,913,725	3,382,111	6,530,341
# of admissions	149,179	120,086	167,770	142,176	579,211
Exposure	2,878,337	2,336,630	3,432,949	2,959,904	11,607,820
Age range 30-65					
# of records	2,050,100	1,680,641	2,468,037	2,126,883	8,325,661
# of individuals	1,902,840	1,540,786	2,261,357	1,973,316	3,729,868
# of admissions	100,212	79,315	115,727	98,957	394,211
Exposure	1,667,545	1,350,233	2,008,552	1,749,273	6,775,603

Table 3.8: Number of records, individuals, exposure and the admissions in each year for age range 0-65 and 30-65.

For some individuals, the data are available for multiple years; hence, the total number of individuals is not the aggregate of individuals over different years. The data set for the age range 30-65 contains 8,325,661 records from 3,729,868 unique individuals with 394,211 admissions under different ICD chapters. The distribution of admissions over the different ICD chapters is shown in Figure 3.3.

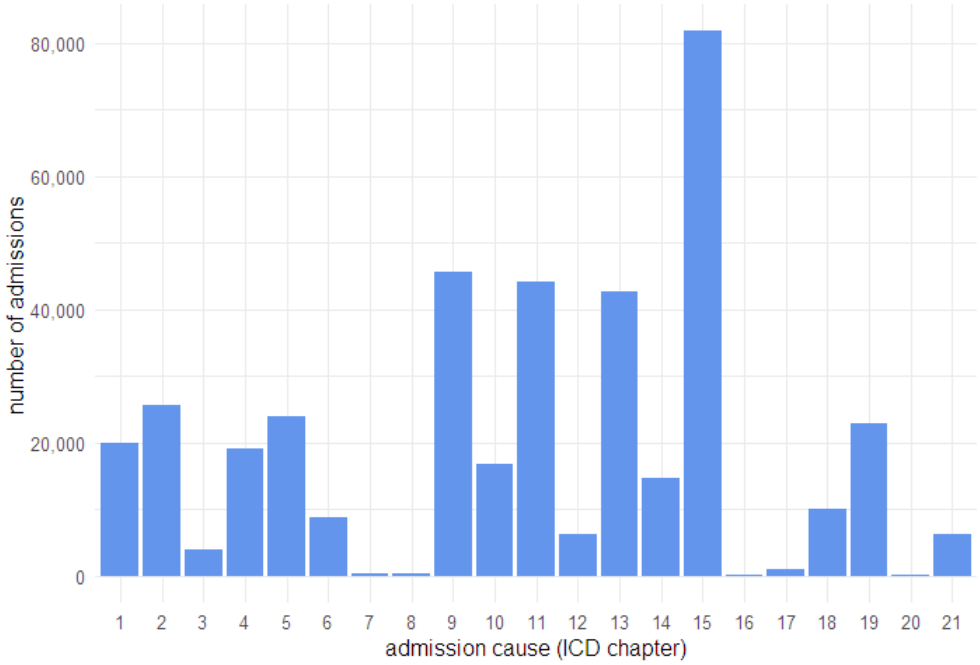


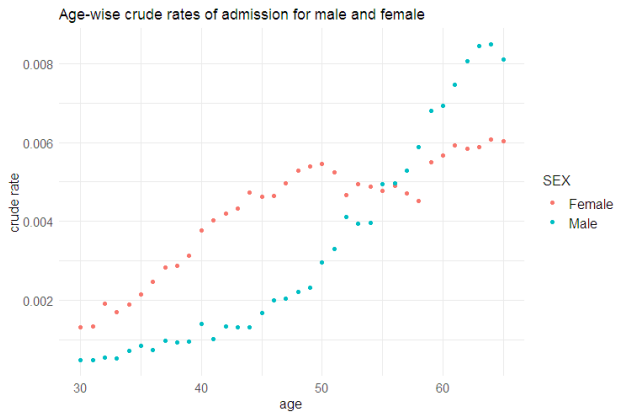
Figure 3.3: Distribution of admissions over different ICD chapters (details about chapters in Table 3.2).

Among these 394,211 admissions, 16,840 were related to respiratory diseases (ICD

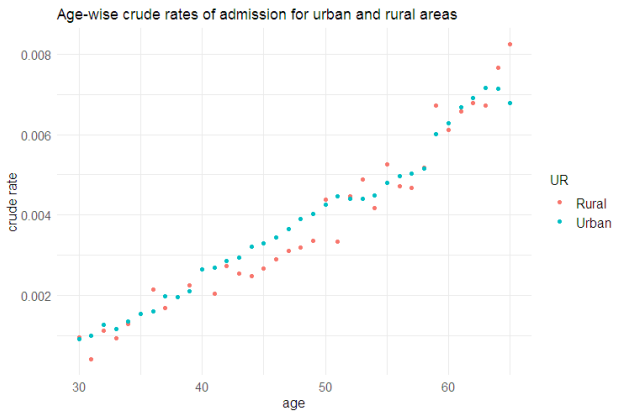
chapter 10), and some individuals had more than one admission. The most prominent cause of admission was pregnancy and childbirth (ICD chapter 15), followed by diseases of the circulatory system (ICD chapter 9), diseases of the digestive system (ICD chapter 11) and diseases related to the musculoskeletal system and connective tissue (ICD chapter 13). At the modelling stage, to mitigate the impact of unusually high admission numbers, a threshold of five admissions per Chapter in a year was set for an individual, taking into account the number of admissions under all the different Chapters. The age-wise crude rates of admission related to all diseases for the male and female population are shown in Figure 3.4.

Figure 3.4: Age-wise crude rates of admission due to all diseases for males and females.

The crude rates of admission for males is showing an increasing trend with age. In contrast, the females' rate is swayed by the admissions related to pregnancy and childbirth up to age 50, making it challenging to ascertain further patterns by looking at all the admissions jointly. As the primary focus of this work is on the admissions rates due to critical illnesses such as cancer and respiratory diseases (Chapters 02 & 10), disease-specific crude rates of admission is assessed for both chapter groupings. Admission frequencies due to both these critical illnesses are given in Table 3.9, and the age-wise crude rates of admission for different categories of the variables such as SEX, UR, EMPREL and YEAR are shown in Figures 3.5 and 3.6.

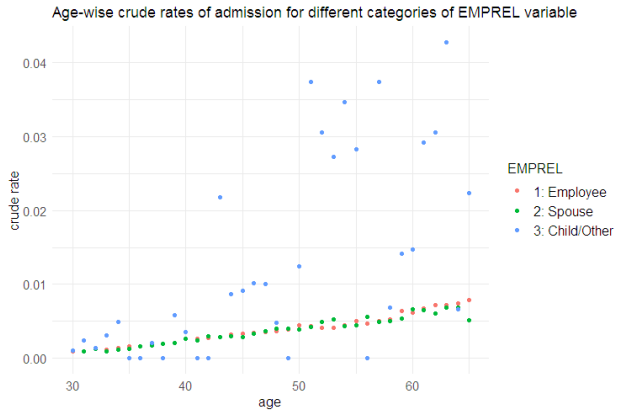


(a)

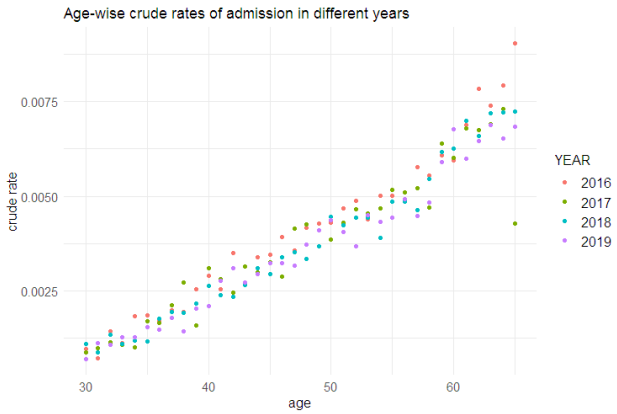


(b)

Figure 3.5: Age-wise crude rates of admission due to neoplasms (ICD-chapter 02): **(a)** male, female; **(b)** urban and rural areas



(c)



(d)

Figure 3.5: *Cont.* Age-wise crude rates of admission due to neoplasms (ICD-chapter 02): **(c)** categories of EMPREL; **(d)** different years.

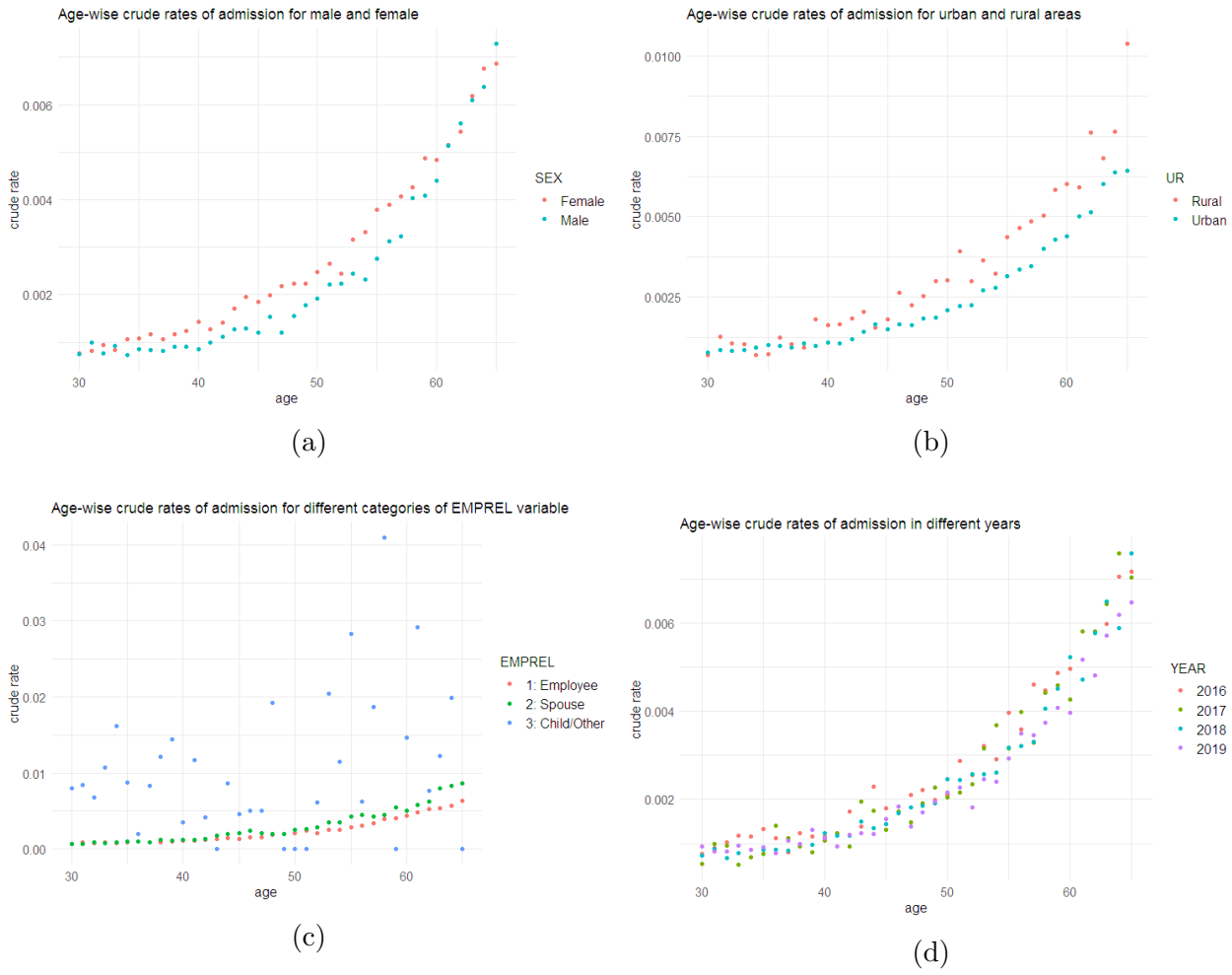


Figure 3.6: Age-wise crude rates of admission due to respiratory diseases (ICD-chapter 10): **(a)** male, female; **(b)** urban and rural areas; **(c)** categories of EMPREL; **(d)** different years.

Age-wise crude rates of admission due to neoplasm for males show an increasing trend with age. Females have a higher rate compared to males in the age range 30 to 55, followed by a lower rate in the older ages. The higher rates for females in the initial period could be attributed to the prevalence of breast cancer among females. Taking admission rates as an indicator of prevalence, the rates also suggest that the prevalence of cancer is slightly higher for the Urban population for the majority of the ages. The cruder rates among the different categories of the EMPREL variable show that the rate of admission at all ages of the 'Child/ Spouse' is mainly different to that of the 'Employee' or 'Spouse' categories.

This inconsistency between the categories of EMPREL variable is also observed for

the rates of admission due to respiratory diseases. In contrast to neoplasm, the prevalence of respiratory diseases is high in rural areas compared to urban areas. The crude rates also suggest that admissions due to respiratory disease are steadily increasing with age, with females having higher rates than males.

No of admissions	Frequency	
	Neoplasms	Respiratory diseases
0	8,302,420	8,310,944
1	21,295	13,201
2	1,639	1,140
3	247	247
4	42	68
5	12	36
6	4	18
7	2	3
8		2
9		-
10		1
11		1
Total	**25,580**	**16,840**

Table 3.9: Frequency of number of admissions related to neoplasms and respiratory diseases for individuals in the age range 30–65 in the years 2016 to 2019.

Finally, in order to determine whether to use EXPOSURE as an offset variable, an exploratory analysis similar to the one mentioned in Ferrario et al. (2020) was carried out using the data on respiratory diseases from the year 2016. The records were categorised using an additional 'Exposure group' variable defined by $E_k = \left(\frac{k-1}{10}, \frac{k}{10}\right]$ with $k = 1, ...10$, which indicated that a significant proportion of the 2,050,100 records belonged to group E_{10} with exposure $e_i \in (0.9, 1]$. A group-wise empirical frequency was also calculated by $\bar{f}_k = \frac{\sum_{i=1}^{n} Y_i \mathbb{1}_{\{e_i \in E_k\}}}{\sum_{i=1}^{n} e_i \mathbb{1}_{\{e_i \in E_k\}}}$, where Y_i represent the number of admissions for i^{th} record (see Table 3.10).

Exposure group E_k	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
relative no. of records	5.07%	3.39%	3.24%	2.87%	5.03%	4.40%	2.36%	2.58%	2.40%	68.66%
empirical frequency $\bar{f}_k$	0.85%	0.71%	0.60%	0.48%	0.53%	0.35%	0.42%	0.36%	0.39%	0.24%

Table 3.10: Relative number of records and empirical frequency for each each exposure group

Even though the empirical frequency of exposure groups gives no evidence to suspect non-linearity of exposure, to identify any potential relationship between the exposure and other features, the distribution of exposure over other variables was also investigated (see Figure 3.7).

Although the analysis indicates slight variation in the exposure composition over different levels of some variables, the evidence is not strong enough to consider alternative treatments of the exposure variable. For instance, for the EESTATU variable, unlike the other categories, for category six (Comprehensive Omnibus Budget Reconciliation Act (COBRA) continuees), the exposure seems to be evenly distributed. COBRA allows employees and their dependents to continue with a health plan provided by an employer for a particular period even after the cessation of their employment with that employer. Individuals with EESTATU six (COBRA) may reach the end of that specified period sometime during the year of study 2016. Although this could be a case of right censoring, we lack sufficient information to confirm this. Hence, it was decided to proceed with treating exposure as an offset variable

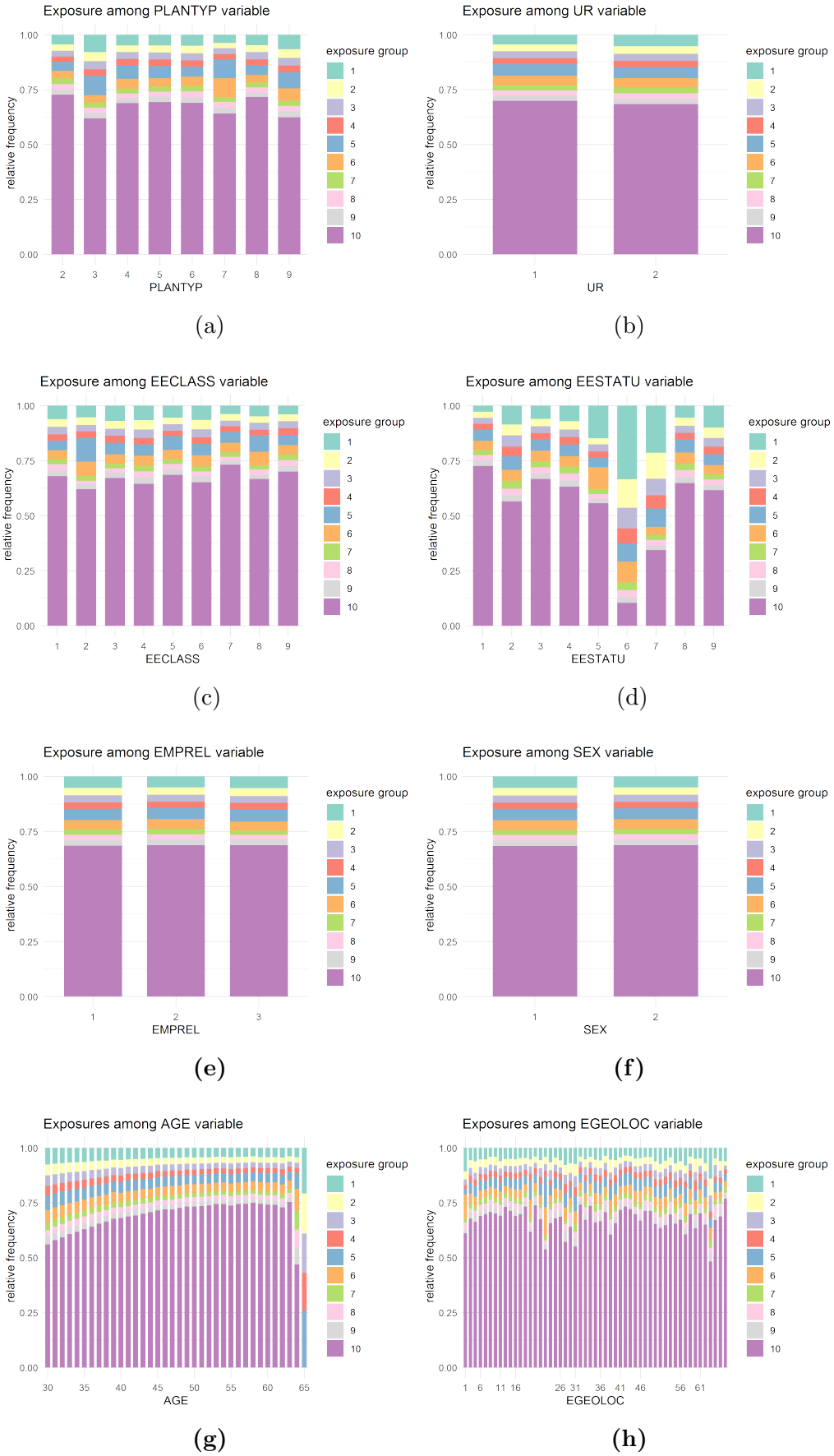


Figure 3.7: Distribution of exposure over different variables: **(a)** PLANTYP; **(b)** UR ; **(c)** EECLASS ; **(d)** EESTATU ; **(e)** EMPREL ; **(f)** SEX ; **(g)** AGE ; **(h)** EGEOLOC.

3.5 Chapter Summary

In conclusion, the admission counts data set fashioned from the MarketScan database is used for the modelling work discussed in the following chapters. For the different segments of work, the size and granularity of the data are decided based on the task at hand. Owing to the particularly large size of the data set, the initial development and discussion of different modelling methodologies were mainly carried out using the data from a single year, 2016, which was then extended for the combined data from all years. The impact of the variation in the size and granularity of the data on the modelling methodologies has also been investigated.

Chapter 4

Estimation and Prediction using Statistical Methods

The development and evaluation of classical and Bayesian regression models for rates of admission related to critical illnesses are detailed in this chapter. The admissions due to respiratory diseases and cancer are considered, and distinct cause-specific models are developed for each of these illnesses.

The different data considerations and exploratory analysis undertaken prior to the modelling stage were described in the previous chapter. Along with the above-mentioned preparatory phase, the development of different models was carried out primarily using the programming language `R` using `RStudio IDE` (R Core Team (2021) and RStudio Team (2021)). The Poisson and negative binomial regression models were fitted using the `glm()` function in the `stats` package and the `gamlss()` function in the `gamlss` package. The `glm()` uses the IWLS method, whereas the `gamlss()` function uses the Rigby and Stasinopoulos (RS) algorithm for estimating the model coefficients (R Core Team (2021) and Rigby and Stasinopoulos (2005)). In Addition to `R & RStudio IDE`, the `WinBUGS` software was also utilised for the development of Bayesian models Lunn et al. (2000) .

4.1 Admissions Related to Respiratory Diseases

The model development is initially carried out using individual year data from 2016 and is then extended for the combined data set from years 2016 to 2019. Considering the size of the data set and the computational challenges presented by the limitations of the WinBUGS software, instead of considering the data on an individual level, the data were consolidated to the risk profile level. In other words, the admission counts and the yearly exposure from different individuals with the same risk characteristics were aggregated, thus reducing the size of the data without altering its attributes. The aggregated data set thus created comprises 425,202 records, with 104,930 in the year 2016.

4.1.1 Classical GLM-type models

Initially, a full Poisson model with all the covariates is considered for modelling the number of admissions. The following functional form is employed for the admission counts due to respiratory diseases (A_i) for the i^{th} risk profile:

$$A_i \sim Poi(\lambda_i e_i) \tag{4.1}$$

where λ_i and e_i are the admission rate and exposure respectively. The mean ($\mu_i^{Pois} = \lambda_i e_i$) depends on risk characteristics $\boldsymbol{x_i}$ through $\lambda_i = \exp(\boldsymbol{\beta}^\top \boldsymbol{x_i})$, and the exposure e_i. By choosing the logarithmic link function, which is in fact the canonical link function for the Poisson GLM, we have a predictor of the form

$$\begin{aligned} \log(\mu_i^{Pois}) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,x_{i,4}} \\ + \beta_{5,x_{i,5}} + \beta_{6,x_{i,6}} + \beta_{7,x_{i,7}} + \beta_{8,x_{i,8}} + o_i \end{aligned} \tag{4.2}$$

where, $o_i = \log(e_i)$ is treated as an offset term. $\boldsymbol{\beta}^\top = (\beta_0, .., \beta_{4,1}, ..., \beta_{4,5}, ..., \beta_{8,8})$ is the unknown vector of coefficients to be estimated with $\boldsymbol{x}_i = (1, x_{i,1}, ..., x_{i,8})^\top$ giving the covariate information. The complete set of covariates $\boldsymbol{x}$ is given in Table 4.1 and the corresponding set of coefficients $\boldsymbol{\beta}$ is presented in Table 4.2. The x_9 in

Table 4.1 corresponds to the YEAR covariate and is only included in the predictor while considering data from multiple years. Also, note that for notation purposes, we will be using the name of the variable where appropriate. For example, $x_{i,4}$ will be represented as x_{region_i}.

Covariate		Description	Number of levels	Categories
x_1	AGE	Age last birthday	Numerical	-
x_2	SEX	Gender	2	0=Female1, 1=Male
x_3	UR	Urban/ rural indicator based on individual's residence	2	0=Rural, 1=Urban
x_4	REGION	Geographical region of residence	5	1: Northeast, 2: North Central, 3: South, 4: West, 5: Unknown
x_5	EECLASS	Employee classification	9	1: Salary Non-union, 2: Salary Union, 3: Salary Other, 4: Hourly Non-union, 5: Hourly Union, 6: Hourly Other, 7: Non-union, 8: Union, 9: Unknown
x_6	EESTATU	Status of employment	9	1: Active Full Time, 2: Active Part Time or Seasonal, 3: Early Retiree, 4: Medicare Eligible Retiree, 5: Retiree (status unknown), 6: Comprehensive Omnibus Budget Reconciliation Act (COBRA) Continuee, 7: Long Term Disability, 8: Surviving Spouse/Depend, 9: Unknown
x_7	EMPREL	Relation to the primary beneficiary	3	1: Employee, 2: Spouse, 3: Child/Other
x_8	PLANTYP	Type of health plan individual is part of	8	2: Comprehensive Plan, 3: Exclusive Provider Organization Plan, 4: Health Maintenance Organization Plan, 5: Non-Capitated (Non-Cap) Point-of-Service, 6: Preferred Provider Organization Plan, 7: Capitated (Cap) or Partially Capitated (PartCap) Point-of-Service Plan, 8: Consumer-Driven Health Plan, 9: High-Deductible Health Plan
x_9	YEAR	Year in which the admission occurred	4	2016, 2017, 2018, 2019

Table 4.1: Description of covariates.

Covariate	Coefficient	Description
Intercept	β_0	$\beta_{intercept}$
AGE	β_1	β_{age}
SEX	β_2	β_{sex}
UR	β_3	β_{ur}
REGION	$\{\beta_{4,1}, ..., \beta_{4,5}\}$	β_{region_a} $a = 1, .., 5$
EECLASS	$\{\beta_{5,1}, ..., \beta_{5,9}\}$	$\beta_{eeclass_b}$ $b = 1, ..., 9$
EESTATU	$\{\beta_{6,1}, ..., \beta_{6,9}\}$	$\beta_{eestatu_c}$ $c = 1, ..., 9$
EMPREL	$\{\beta_{7,1}, ..., \beta_{7,3}\}$	β_{emprel_d} $d = 1, ..., 3$
PLANTYP	$\{\beta_{8,1}, ..., \beta_{8,8}\}$	$\beta_{plantype_e}$ $e = 1, ..., 8$
YEAR	$\{\beta_{9,1}, ..., \beta_{9,4}\}$	β_{year_f} $f = 1, ..., 4$

Table 4.2: List of covariates and the corresponding coefficient parameters in the Poisson regression model.

While fitting the model, for categorical variables with more than two levels, a sum-to-zero constraint ($\sum_k(\beta_{j,k}) = 0, \ \forall\ j$) was applied to the corresponding coefficients for ease of interpretation. Additionally, the age variable, which spans from 30 to 65, is standardised using the transformation $(age - mean(age))/SD(age)$. The sum-to-zero constraint and the standardisation of age variable have been consistently applied for all models discussed throughout this chapter. The parameter estimates for the full Poisson regression model (Equation 4.2) is shown in Table 4.3. The coefficient estimates of the different levels of categorical variables (β_{jk}) represent the potential change in the effect of that specific level on the response, compared to the average effect of the different levels (or to the base level if a corner constraint was used instead of a sum-to-zero constraint), and the significance indicators show whether the change is significant. If the estimates of all the levels of a categorical variable appear to be not significant, this implies that there does not exist substantial variation between individuals under those different levels in their admission rates. In other words, this conveys that specific variable is not a key risk driver of admission rates, and that the variable is not a determinant factor within the model.

Coefficient		Estimate	Std. Error	z value	Pr(> |z|)	Signi.
β_0	$\beta_{intercept}$	-5.0525	0.1256	-40.2400	< 2e-16	***
β_1	β_{age}	0.6044	0.0189	31.9240	< 2e-16	***
β_2	$\beta_{sex2-female}$	0.0945	0.0307	3.0830	0.0021	**
β_3	$\beta_{ur2-urban}$	-0.1596	0.0414	-3.8550	0.0001	***
$\beta_{4,region_a}$	$\beta_{region1-northeast}$	-0.1768	0.0882	-2.0040	0.0451	*
	$\beta_{region2-northcentral}$	-0.0508	0.0868	-0.5850	0.5584	
	$\beta_{region3-south}$	-0.0610	0.0849	-0.7180	0.4727	
	$\beta_{region4-west}$	-0.4980	0.0908	-5.4820	0.0000	***
	$\beta_{region5-unknown}$	0.7866	0.3275	2.4020	0.0163	*
$\beta_{5,eeclass_b}$	$\beta_{eeclass1-salarynon-union}$	-0.2820	0.0460	-6.1360	0.0000	***
	$\beta_{eeclass2-salaryunion}$	-0.0647	0.1272	-0.5080	0.6114	
	$\beta_{eeclass3-salaryother}$	-0.2537	0.0799	-3.1740	0.0015	**
	$\beta_{eeclass4-hourlynon-union}$	0.2179	0.0451	4.8350	0.0000	***
	$\beta_{eeclass5-hourlyunion}$	0.3188	0.0478	6.6750	0.0000	***
	$\beta_{eeclass6-hourlyother}$	0.1917	0.0758	2.5300	0.0114	*
	$\beta_{eeclass7-non-union}$	0.1213	0.0453	2.6770	0.0074	**
	$\beta_{eeclass8-union}$	-0.0889	0.0829	-1.0720	0.2835	
	$\beta_{eeclass9-unknown}$	-0.1604	0.0449	-3.5710	0.0004	***
$\beta_{6,eestatu_c}$	$\beta_{eestatu1-activefulltime}$	-0.3855	0.0535	-7.2050	0.0000	***
	$\beta_{eestatu2-activeparttimeorseasonal}$	-1.0486	0.1852	-5.6610	0.0000	***
	$\beta_{eestatu3-earlyretiree}$	-0.2734	0.0637	-4.2910	0.0000	***
	$\beta_{eestatu4-medicareeligibleretiree}$	-0.1978	0.1081	-1.8300	0.0672	.
	$\beta_{eestatu5-retiree(statusunknown)}$	-0.3024	0.2421	-1.2490	0.2116	
	$\beta_{eestatu6-COBRAcontinuee}$	0.3863	0.1281	3.0160	0.0026	**
	$\beta_{eestatu7-longtermdisability}$	1.2988	0.1291	10.0620	< 2e-16	***
	$\beta_{eestatu8-survivingspouse/depend}$	0.3783	0.1542	2.4530	0.0142	*
	$\beta_{eestatu9-unknown}$	0.1442	0.0649	2.2240	0.0262	*
$\beta_{7,emprel_d}$	$\beta_{emprel1-employee}$	-0.5525	0.0694	-7.9550	0.0000	***
	$\beta_{emprel2-spouse}$	-0.2978	0.0697	-4.2730	0.0000	***
	$\beta_{emprel3-child/other}$	0.8503	0.1354	6.2810	0.0000	***
$\beta_{8,plantyp_e}$	$\beta_{plantyp2-ComprehensivePlan}$	0.1213	0.0728	1.6670	0.0956	.
	$\beta_{plantyp3-EPOPlan}$	0.0888	0.1467	0.6050	0.5449	
	$\beta_{plantyp4-HMOPlan}$	0.0474	0.0502	0.9450	0.3448	
	$\beta_{plantyp5-Non-CapPoSPlan}$	-0.1588	0.0607	-2.6160	0.0089	**
	$\beta_{plantyp6-PPOPlan}$	0.0430	0.0355	1.2110	0.2257	
	$\beta_{plantyp7-CaporPartCapPoSPlan}$	0.1878	0.1071	1.7530	0.0795	.
	$\beta_{plantyp8-CDHP}$	-0.0005	0.0537	-0.0080	0.9932	
	$\beta_{plantyp9-HDHP}$	-0.3291	0.0680	-4.8400	0.0000	***

Table 4.3: Coefficient estimates based on the Poisson regression model with the significance codes (‘***’,‘**’,‘*’, ‘.’,‘ ’) indicating the level of significance of the estimates within the intervals (0, 0.001], (0.001, 0.01], (0.01, 0.05], (0.05, 0.1], and (0.1, 1].

As part of the model fitting process, a backward step-wise feature selection process was carried out based on likelihood measures such as Akaike's Information Criterion (AIC) and Bayesian Information Criterion (BIC). The AIC measure is given by $(D(\hat{\theta}_{MLE}) + 2K)$ and BIC by the formula $(D(\hat{\theta}_{MLE}) + K \ln n)$ where $D(.)$ is the deviance function with $\hat{\theta}_{MLE}$ being the maximum likelihood estimate (MLE) of the parameters. K is the number of parameters and n is the sample size (Ward (2008)). Under the backward step-wise variable selection process, we start from a full model with all the covariates and consider removing one variable at a time. The outcome of any particular step is the model that yields the lowest AIC or BIC value, and the whole process is repeated until there is no further reduction in the value of the chosen measure. Contrary to the AIC measure known to favour complex models, the BIC metric penalises complex models and leans towards simpler models (Vrieze (2012)). The Bayesian variant Deviance Information Criterion (DIC) was utilised for the hierarchical Bayesian models presented later in this chapter. As detailed in Ward (2008), each of these measures has its own benefits and shortcomings. Therefore, when relying on a particular measure, feature selection should also consider other aspects related to modelling, such as the complexity of models, ease of implementation and explainability, as well as practical importance in the context of our application. For the Poisson regression model, the backward step-wise variable selection was carried out using both AIC and BIC measures, details of which are given in Table 4.4 and Table 4.5 respectively.

As anticipated, the BIC proposed a smaller model by excluding the plan type and gender variables, whereas the AIC suggested the full model with all the covariates. As the focus is not just on the models' fit but also on the impact of different risk features on the admission rates, it was decided to retain both variables and proceed with the full model.

Model	# of estimated parameters	AIC
Step 1:∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL + PLANTYP		
Full model	33	24551
- SEX	32	24558
- UR	32	24563
- PLANTYP	26	24579
- REGION	29	24630
- EMPREL	31	24636
- EECLASS	25	24691
- EESTATU	25	24783
- AGE	32	25668

Table 4.4: Summary of the variable selection process for Poisson regression model using AIC.

Model	# of estimated parameters	BIC
Step 1:∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL + PLANTYP		
- PLANTYP	26	24828
- SEX	32	24864
Full model	33	24866
- UR	32	24869
- REGION	29	24908
- EECLASS	25	24930
- EMPREL	31	24932
- EESTATU	25	25022
- AGE	32	25974
Step 2:∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL		
- SEX	25	24826
- UR	25	24831
- REGION	22	24874
- EMPREL	24	24894
- EECLASS	18	24906
- EESTATU	18	24998
- AGE	25	25944
Step 3:∼ AGE + REGION + UR + EECLASS + EESTATU + EMPREL		
- UR	24	24828
- REGION	21	24872
- EMPREL	23	24900
- EECLASS	17	24904
- EESTATU	17	25000
- AGE	24	25938

Table 4.5: Summary of the variable selection process for Poisson regression model using BIC.

During the exploratory analysis, the age-wise crude rates of admission due to respiratory diseases among males and females (see Figure 3.6) showcased a potential non-linear impact of age on admission rates. Hence it was decided to consider an age polynomial within the linear predictor, and the modified predictor has the functional form:

$$\log(\mu_i^{Pois}) = \beta_0 + \sum_{k=1}^{r} \beta_{1,k} x_{i,1}^k + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} + \beta_{5,eeclass_i} + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} + o_i \tag{4.3}$$

where r is the degree of age polynomial that needs to be determined. For that, models with different degrees for the age polynomial were evaluated by considering the AIC and BIC values (see Table 4.6).

Degree	# of estimated parameters	BIC	AIC
r=1	33	24866	24551
r=2	34	24873	24548
r=3	35	24882	24547
r=4	36	24893	24549
r=5	37	24904	24550

Table 4.6: AIC and BIC values of the Poisson regression models with different degree ($r = 1, .., 5$) for the age polynomial.

The BIC favoured a simpler model with degree one. Using AIC, the model performance did not improve substantially for higher degrees. The reduction in AIC value is minimal for models with non-linear age terms. These results thus nullified the concern of the potential non-linear impact of age on admission rates.

In order to evaluate the potential interactions between covariates, a forward step-wise method using BIC was employed. In this case, we start from a base model (full model with all the covariates in this case) and consider adding one interaction term at a time. As before, the outcome of any particular step is the model that yields the lowest BIC value. The summary of the procedure is given in Table 4.7 with step-wise details provided in Table A.5.

Model	# of estimated parameters	BIC
Full model: ∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL + PLANTYP		
Full model	33	24866
+ SEX:EMPREL	35	24851

Table 4.7: Summary of the step-wise selection process for identifying relevant interaction terms of Poisson regression model using BIC.

Although the results suggested the inclusion of interaction terms between SEX and EMPREL variables within the predictor, the coefficient estimates for the interaction terms were not highly significant (see Table A.6) and hence were not included to avoid needless complexity. Thus the model fitting process for the Poisson regression model for admission rates related to respiratory diseases yielded a model with the linear predictor having all the covariates without any interaction terms.

The same functional form of the predictor was used for the NB regression model in which the admission counts, A_i, is assumed to follow a NB distribution with a dispersion parameter ϕ resulting in the model structure:

$$A_i \sim NB(\mu_i^{NB}, \phi) \tag{4.4}$$

with the mean μ_i^{NB} having the form

$$\begin{aligned} \log(\mu_i^{NB}) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} + \beta_{5,eeclass_i} \\ + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} + o_i \end{aligned} \tag{4.5}$$

The coefficient estimates for the NB regression model, along with corresponding significance, are given in Table 4.8. The results from the Poisson (Table 4.3) and NB regression models appear to be comparable in terms of the coefficient estimates and the significance of the estimates. A detailed comparison between the models and the analysis of the interpretations derived from the coefficient estimates are carried out later alongside the Bayesian hierarchical models.

Coefficient		Estimate	Std. Error	z value	Pr(> |z|)	Signi.
β_0	$\beta_{intercept}$	-5.0840	0.1307	-38.9030	< 2e-16	***
β_1	β_{age}	0.6125	0.0220	27.8330	< 2e-16	***
β_2	$\beta_{sex2-female}$	0.0818	0.0365	2.2400	0.0251	*
β_3	$\beta_{ur2-urban}$	-0.1292	0.0472	-2.7360	0.0062	**
$\beta_{4,region_a}$	$\beta_{region1-northeast}$	-0.2070	0.0902	-2.2950	0.0218	*
	$\beta_{region2-northcentral}$	-0.0576	0.0883	-0.6520	0.5142	
	$\beta_{region3-south}$	-0.0414	0.0865	-0.4780	0.6326	
	$\beta_{region4-west}$	-0.4827	0.0925	-5.2210	0.0000	***
	$\beta_{region5-unknown}$	0.7887	0.3289	2.3980	0.0165	*
$\beta_{5,eeclass_b}$	$\beta_{eeclass1-salarynon-union}$	-0.2777	0.0504	-5.5150	0.0000	***
	$\beta_{eeclass2-salaryunion}$	-0.0465	0.1292	-0.3600	0.7187	
	$\beta_{eeclass3-salaryother}$	-0.2330	0.0825	-2.8240	0.0048	**
	$\beta_{eeclass4-hourlynon-union}$	0.2240	0.0503	4.4530	0.0000	***
	$\beta_{eeclass5-hourlyunion}$	0.3397	0.0519	6.5500	0.0000	***
	$\beta_{eeclass6-hourlyother}$	0.1797	0.0795	2.2600	0.0238	*
	$\beta_{eeclass7-non-union}$	0.1221	0.0506	2.4150	0.0157	*
	$\beta_{eeclass8-union}$	-0.0849	0.0891	-0.9530	0.3405	
	$\beta_{eeclass9-unknown}$	-0.2233	0.0516	-4.3310	0.0000	***
$\beta_{6,eestatu_c}$	$\beta_{eestatu1-activefulltime}$	-0.4177	0.0569	-7.3430	0.0000	***
	$\beta_{eestatu2-activeparttimeorseasonal}$	-1.0819	0.1873	-5.7770	0.0000	***
	$\beta_{eestatu3-earlyretiree}$	-0.3025	0.0692	-4.3730	0.0000	***
	$\beta_{eestatu4-medicareeligibleretiree}$	-0.2171	0.1233	-1.7610	0.0782	.
	$\beta_{eestatu5-retiree(statusunknown)}$	-0.2757	0.2495	-1.1050	0.2691	
	$\beta_{eestatu6-COBRAcontinuee}$	0.3607	0.1303	2.7670	0.0057	**
	$\beta_{eestatu7-longtermdisability}$	1.2704	0.1321	9.6150	< 2e-16	***
	$\beta_{eestatu8-survivingspouse/depend}$	0.3360	0.1676	2.0050	0.0449	*
	$\beta_{eestatu9-unknown}$	0.3279	0.0732	4.4790	0.0000	***
$\beta_{7,emprel_d}$	$\beta_{emprel1-employee}$	-0.5558	0.0708	-7.8470	0.0000	***
	$\beta_{emprel2-spouse}$	-0.3081	0.0711	-4.3340	0.0000	***
	$\beta_{emprel3-child/other}$	0.8638	0.1368	6.3140	0.0000	***
$\beta_{8,plantyp_e}$	$\beta_{plantyp2-ComprehensivePlan}$	0.0884	0.0820	1.0780	0.2809	
	$\beta_{plantyp3-EPOPlan}$	0.0961	0.1492	0.6440	0.5193	
	$\beta_{plantyp4-HMOPlan}$	0.0320	0.0561	0.5700	0.5685	
	$\beta_{plantyp5-Non-CapPoSPlan}$	-0.1531	0.0670	-2.2870	0.0222	*
	$\beta_{plantyp6-PPOPlan}$	0.0908	0.0405	2.2410	0.0250	*
	$\beta_{plantyp7-CaporPartCapPoSPlan}$	0.0600	0.1342	0.4470	0.6547	
	$\beta_{plantyp8-CDHP}$	0.0757	0.0583	1.2980	0.1941	
	$\beta_{plantyp9-HDHP}$	-0.2898	0.0718	-4.0350	0.0001	***

Table 4.8: Coefficient estimates based on the NB regression model with the significance codes (‘***’,‘**’,‘*’, ‘.’,‘ ’) indicating the level of significance of the estimates within the intervals (0, 0.001], (0.001, 0.01], (0.01, 0.05], (0.05, 0.1], and (0.1, 1].

4.1.2 Bayesian regression models

This part of the thesis focuses on the development of Bayesian regression models for admission rates related to respiratory diseases. Within the Bayesian framework, we consider three different models: Bayesian Poisson (BP), Bayesian Poisson-Gamma (BP-G) and Bayesian Poisson Log-normal (BP-LN) models. Even though all three models have an underlying Poisson distributional assumption, the model structure varies. Estimation for these models was performed using Markov Chain Monte Carlo (MCMC) methodology, using the `WinBUGS` software by invoking it from `R` (see Section 2.3 for more details). As part of the model-fitting process, we run 30,000 iterations for a single chain. The output was then examined for convergence using the trace plots of the parameters, and we found that a burn-in period of 10,000 was sufficient. The remaining 20,000 realisations from the post-burn-in period are then utilised to obtain the posterior estimates. In order to aid the convergence of the chain, the coefficient estimates from the classical Poisson regression model were used as the initial values for the model parameters. The same combination of the number of iterations and burn-in period was used for all three models.

4.1.2.1 Bayesian Poisson regression model

The Bayesian Poisson (BP) regression model for admissions related to respiratory diseases has the following structure:

$$A_i|\lambda_i \sim Poi(\lambda_i e_i), \qquad \text{for } i = 1, .., n \tag{4.6}$$

where $\lambda_i = \exp(\boldsymbol{\beta}^\top \boldsymbol{x_i})$ or equivalently,

$$\begin{aligned} \log(\lambda_i) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} \\ + \beta_{5,eeclass_i} + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} \end{aligned} \tag{4.7}$$

which is similar to that of a classical regression model. The same set of covariates and the associated parameters as defined in Tables 4.1 and 4.2 are used for the BP

model as well. The fundamental difference is the prior assumptions assigned to the model parameters $\boldsymbol{\beta}$. For all the β parameters, a vague non-informative normal prior is assigned as shown below:

$$\beta_0 \sim N(0, 10^4)$$

$$\beta_1 \sim N(0, 10^4)$$

$$\beta_2 \sim N(0, 10^4)$$

$$\beta_3 \sim N(0, 10^4)$$

$$\beta_{4,region_k} \sim N(0, 10^4) \qquad \text{for } k = 1, ..., 5$$

$$\beta_{5,eeclass_l} \sim N(0, 10^4) \qquad \text{for } l = 1, ..., 9$$

$$\beta_{6,eestatu_m} \sim N(0, 10^4) \qquad \text{for } m = 1, ..., 9$$

$$\beta_{7,emprel_n} \sim N(0, 10^4) \qquad \text{for } n = 1, ..., 3$$

$$\beta_{8,plantyp_p} \sim N(0, 10^4) \qquad \text{for } p = 1, ..., 8.$$

A Bayesian variable selection was carried out for the BP model using the DIC measure. Following Spiegelhalter et al. (2002), the DIC measure is specified as $(D(\bar{\theta}) + 2pD)$ where $D(\bar{\theta})$ represents the deviance at the posterior mean and $pD = \overline{D(\theta)} - D(\bar{\theta})$ with $\overline{D(\theta)}$ being the posterior mean deviance. As in the case of the classical regression model, a backward step-wise approach was employed but instead of the AIC or BIC values, the DIC measure is used. The summary of the DIC based Bayesian variable selection process is shown in Table 4.9. At this point, it is also worth noting that in Bayesian analysis, the choice of prior impacts the posterior inferences, including the DIC. In this context, for the β parameters, we have adopted a highly vague normal prior with a large variance parameter. Even though small differences in the variance parameter may not have much impact, a much smaller variance parameter would have meant a more informative prior and could

have influenced the DIC value. In addition to the sensitivity to prior assumptions, there are other criticisms regarding the limitations of DIC. For example, the 'lack of invariance to reparameterisation' and the 'parameter focus' issue-that the DIC is influenced by the choice of the level of parameters in focus, which in turn determines the specification of model likelihood (Spiegelhalter et al. (2002), Millar (2009)). As it is beyond the scope of this thesis, for more discussion on the same, we refer to Spiegelhalter et al. (2014), Gibson et al. (2018) and Celeux et al. (2006). Nevertheless, due to its ease of implementation and adequacy in facilitating the comparison of Bayesian models to an acceptable level, DIC remains one of the widely accepted model selection criteria. We proceed with a highly vague prior assumption.

Model	Dbar	Dhat	pD	DIC
Step 1:∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL + PLANTYP				
Full model	24517.7	24484.9	32.8	**24550.5**
-AGE	25636.5	25604.3	32.2	25668.7
-REGION	24601.0	24572.7	28.3	24629.3
-SEX	24526.1	24494.6	31.5	24557.6
-UR	24531.7	24499.5	32.2	24563.9
-EECLASS	24665.6	24640.9	24.7	24690.3
-EESTATU	24757.5	24732.6	24.8	24782.3
-EMPREL	24604.2	24573.8	30.4	24634.7
-PLANTYP	24554.0	24527.7	26.3	24580.3

Table 4.9: Summary of the variable selection process for Bayesian Poisson regression model using DIC. Dbar = $\overline{D(\theta)}$ and Dhat = $D(\bar{\theta})$

Similar to the AIC and BIC approaches, the model with the lowest DIC value is preferred, and hence, the full model with all the covariates was adopted. The same covariates were considered for Bayesian Poisson Log-normal and Bayesian Poisson-Gamma models.

The coefficient estimates of the BP model are given in Table 4.10, and the trace plots for the different parameters over the 20,000 iterations are shown in Figure 4.1. The table contains the posterior mean and standard deviation estimates along with the Monte Carlo error (MCE) and posterior percentile (2.50%, 50%, 97.50%) values. The MCE value shows the variability arising as part of the underlying simulation

process and reflects the accuracy of the posterior estimates Lunn et al. (2000).

Coefficient	Description	Mean	SD	MC error	2.5%	50%	97.5%
β_0	(Intercept)	-5.103	0.112	0.008	-5.329	-5.096	-4.900
β_1	AGE	0.605	0.019	0.000	0.568	0.605	0.642
β_2	SEX 2-Female	0.096	0.030	0.001	0.036	0.096	0.156
β_3	UR 2-urban	-0.158	0.042	0.002	-0.239	-0.160	-0.077
	REGION1-Northeast	-0.156	0.082	0.005	-0.306	-0.162	0.014
	REGION2-Northcentral	-0.030	0.080	0.005	-0.175	-0.034	0.138
$\beta_{4,region_a}$	REGION3-South	-0.039	0.077	0.005	-0.180	-0.044	0.120
	REGION4-West	-0.476	0.087	0.005	-0.639	-0.477	-0.295
	REGION5-Unknown	0.701	0.300	0.021	0.062	0.722	1.236
	EECLASS1-Salary Non-Union	-0.283	0.047	0.002	-0.377	-0.282	-0.190
	EECLASS2-Salary Union	-0.055	0.127	0.008	-0.305	-0.052	0.187
	EECLASS3-Salary Other	-0.258	0.082	0.004	-0.421	-0.257	-0.102
	EECLASS4-Hourly Non-union	0.217	0.045	0.002	0.129	0.217	0.307
$\beta_{5,eeclass_b}$	EECLASS5-Hourly Union	0.322	0.048	0.002	0.229	0.323	0.414
	EECLASS6-Hourly Other	0.191	0.076	0.004	0.040	0.189	0.345
	EECLASS7-Non-union	0.122	0.046	0.002	0.033	0.122	0.209
	EECLASS8-Union	-0.097	0.086	0.005	-0.265	-0.099	0.075
	EECLASS9-Unknown	-0.159	0.045	0.001	-0.247	-0.159	-0.072
	EESTATU1-Active Full Time	-0.376	0.058	0.004	-0.491	-0.375	-0.263
	EESTATU2-Active Part Time or Seasonal	-1.049	0.174	0.012	-1.399	-1.054	-0.722
	EESTATU3-Early Retiree	-0.265	0.068	0.004	-0.396	-0.266	-0.128
	EESTATU4-Medicare Eligible Retiree	-0.195	0.111	0.006	-0.417	-0.191	0.017
$\beta_{6,eestatu_c}$	EESTATU5-Retiree (status unknown)	-0.341	0.263	0.021	-0.881	-0.328	0.107
	EESTATU6-COBRA Continuee	0.396	0.130	0.008	0.147	0.398	0.639
	EESTATU7-Long Term Disability	1.287	0.132	0.008	1.004	1.289	1.530
	EESTATU8-Surviving Spouse/Depend.	0.389	0.151	0.010	0.108	0.382	0.702
	EESTATU9-Unknown	0.154	0.068	0.003	0.023	0.154	0.286
	EMPREL1-Employee	-0.543	0.070	0.004	-0.674	-0.546	-0.393
$\beta_{7,emprel_d}$	EMPREL2-Spouse	-0.288	0.070	0.004	-0.415	-0.291	-0.144
	EMPREL3-Child/other	0.831	0.136	0.009	0.543	0.837	1.079
	PLANTYP2-Comprehensive Plan	0.122	0.073	0.003	-0.019	0.123	0.265
	PLANTYP3-EPO Plan	0.078	0.148	0.007	-0.228	0.078	0.366
	PLANTYP4-HMO Plan	0.051	0.050	0.001	-0.049	0.052	0.147
	PLANTYP5-Non-Cap PoS Plan	-0.156	0.059	0.002	-0.272	-0.155	-0.040
$\beta_{8,plantyp_e}$	PLANTYP6-PPO Plan	0.048	0.036	0.001	-0.021	0.049	0.118
	PLANTYP7-Cap or Part Cap PoS Plan	0.180	0.106	0.003	-0.037	0.183	0.381
	PLANTYP8-CDHP	0.004	0.053	0.002	-0.100	0.003	0.111
	PLANTYP9-HDHP	-0.327	0.067	0.001	-0.459	-0.327	-0.197

Table 4.10: Posterior estimates of the Bayesian Poisson model for admissions related to respiratory diseases in 2016.

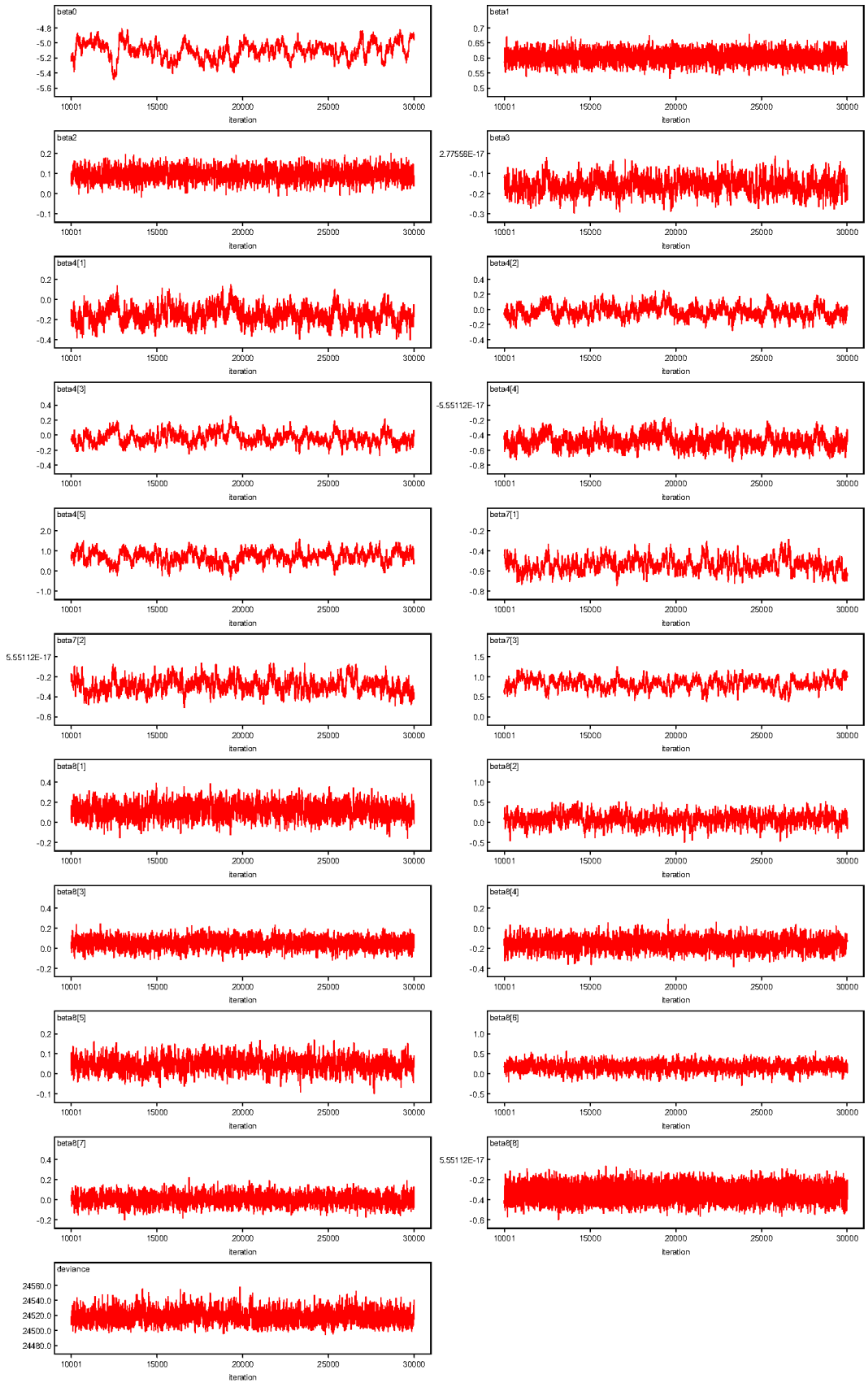


Figure 4.1: Trace plots for the parameters of the Bayesian Poisson model for admissions related to respiratory diseases in 2016.

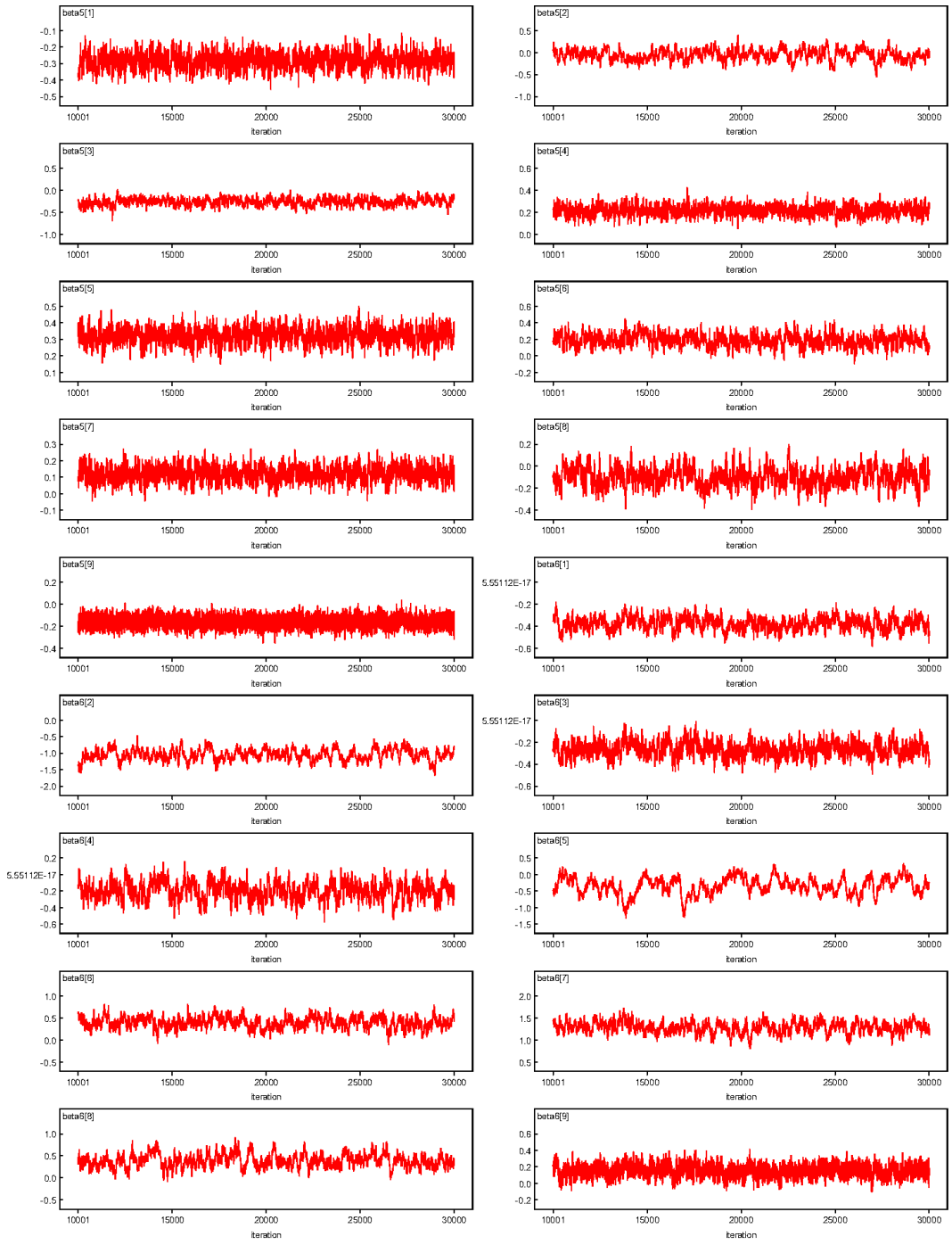


Figure 4.1: *Cont.*

The trace plots of the parameters of the Bayesian Poisson model clearly show that the model has converged. The plots also indicate that the choice of burn-in period was adequate. Additionally, the very low MCE for all the parameters also indicate that the number of iterations considered post-burn-in period was sufficient. A detailed discussion of the parameter estimates is carried out later in this chapter,

alongside the results from other models (see Section 4.1.3).

4.1.2.2 Bayesian Poisson Log-normal model

In the case of the Bayesian Poisson Log-normal model, the underlying distributional assumption is the same as that of the previous BP model, and thus we have:

$$A_i|\lambda_i \sim Poi(\lambda_i e_i), \qquad \text{for } i = 1, .., n. \tag{4.8}$$

We now assume a prior distribution for the rates λ_i, to facilitate capturing the variability and over-dispersion within the data. A Log-normal prior distribution is assigned to λ_i, resulting in $\lambda_i|\mu_i, \sigma^2 \sim Log - normal(\mu_i, \sigma^2)$ or equivalently:

$$\log(\lambda_i)|\mu_i, \sigma^2 \sim N(\mu_i, \sigma^2) \tag{4.9}$$

with

$$\begin{aligned}\mu_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} \\ + \beta_{5,eeclass_i} + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i}\end{aligned} \tag{4.10}$$

In addition to the vague priors specified for the β parameters in the BP model specification, a vague prior is assigned to the precision parameter $\tau = (1/\sigma^2)$ as $\tau \sim Gamma(1, 0.001)$ (Congdon (2019)). The posterior parameter estimates of the BP-LN model are given in Table 4.11 and the trace plots for the different parameters over the 20,000 iterations after the burn-in phase are shown in Figure 4.2.

Coefficient	Description	Mean	SD	MC error	2.50%	50%	97.50%
β_0	(Intercept)	-5.803	0.141	0.010	-6.096	-5.792	-5.554
β_1	AGE	0.626	0.022	0.001	0.582	0.627	0.670
β_2	SEX 2-Female	0.093	0.040	0.002	0.015	0.092	0.172
β_3	UR 2-urban	-0.069	0.054	0.003	-0.175	-0.070	0.040
	REGION1-Northeast	-0.191	0.098	0.007	-0.361	-0.199	0.016
	REGION2-Northcentral	-0.017	0.099	0.007	-0.195	-0.021	0.189
$\beta_{4,region_a}$	REGION3-South	0.024	0.096	0.007	-0.152	0.019	0.218
	REGION4-West	-0.454	0.101	0.007	-0.645	-0.456	-0.244
	REGION5-Unknown	0.638	0.365	0.026	-0.100	0.658	1.298
	EECLASS1-Salary Non-Union	-0.263	0.053	0.003	-0.368	-0.263	-0.158
	EECLASS2-Salary Union	-0.070	0.134	0.009	-0.348	-0.067	0.177
	EECLASS3-Salary Other	-0.243	0.089	0.006	-0.417	-0.245	-0.066
	EECLASS4-Hourly Non-union	0.236	0.053	0.003	0.129	0.237	0.340
$\beta_{5,eeclass_b}$	EECLASS5-Hourly Union	0.344	0.054	0.003	0.237	0.346	0.446
	EECLASS6-Hourly Other	0.150	0.081	0.005	-0.007	0.151	0.307
	EECLASS7-Non-union	0.140	0.052	0.003	0.036	0.141	0.240
	EECLASS8-Union	-0.090	0.093	0.005	-0.276	-0.089	0.089
	EECLASS9-Unknown	-0.204	0.052	0.003	-0.308	-0.202	-0.108
	EESTATU1-Active Full Time	-0.386	0.054	0.003	-0.489	-0.386	-0.273
	EESTATU2-Active Part Time or Seasonal	-1.113	0.190	0.015	-1.478	-1.110	-0.729
	EESTATU3-Early Retiree	-0.306	0.068	0.004	-0.444	-0.305	-0.173
	EESTATU4-Medicare Eligible Retiree	-0.184	0.126	0.007	-0.437	-0.183	0.057
$\beta_{6,eestatu_c}$	EESTATU5-Retiree (status unknown)	-0.374	0.276	0.018	-0.949	-0.356	0.121
	EESTATU6-COBRA Continuee	0.369	0.132	0.010	0.099	0.381	0.601
	EESTATU7-Long Term Disability	1.254	0.130	0.008	1.000	1.254	1.511
	EESTATU8-Surviving Spouse/Depend.	0.307	0.180	0.011	-0.075	0.311	0.646
	EESTATU9-Unknown	0.434	0.075	0.005	0.289	0.432	0.594
	EMPREL1-Employee	-0.523	0.074	0.005	-0.659	-0.526	-0.373
$\beta_{7,emprel_d}$	EMPREL2-Spouse	-0.312	0.078	0.005	-0.458	-0.318	-0.156
	EMPREL3-Child/other	0.836	0.147	0.010	0.544	0.846	1.104
	PLANTYP2-Comprehensive Plan	0.099	0.089	0.005	-0.083	0.102	0.272
	PLANTYP3-EPO Plan	0.071	0.136	0.009	-0.199	0.066	0.340
	PLANTYP4-HMO Plan	0.018	0.063	0.004	-0.111	0.019	0.142
	PLANTYP5-Non-Cap PoS Plan	-0.165	0.066	0.004	-0.297	-0.166	-0.033
$\beta_{8,plantyp_e}$	PLANTYP6-PPO Plan	0.128	0.042	0.002	0.044	0.128	0.209
	PLANTYP7-Cap or Part Cap PoS Plan	0.053	0.142	0.007	-0.241	0.056	0.326
	PLANTYP8-CDHP	0.088	0.060	0.003	-0.028	0.087	0.211
	PLANTYP9-HDHP	-0.292	0.074	0.005	-0.436	-0.289	-0.152
τ	Precision	0.894	0.050	0.004	0.797	0.896	0.988

Table 4.11: Posterior estimates of the BP-LN model for admissions related to respiratory diseases in 2016.

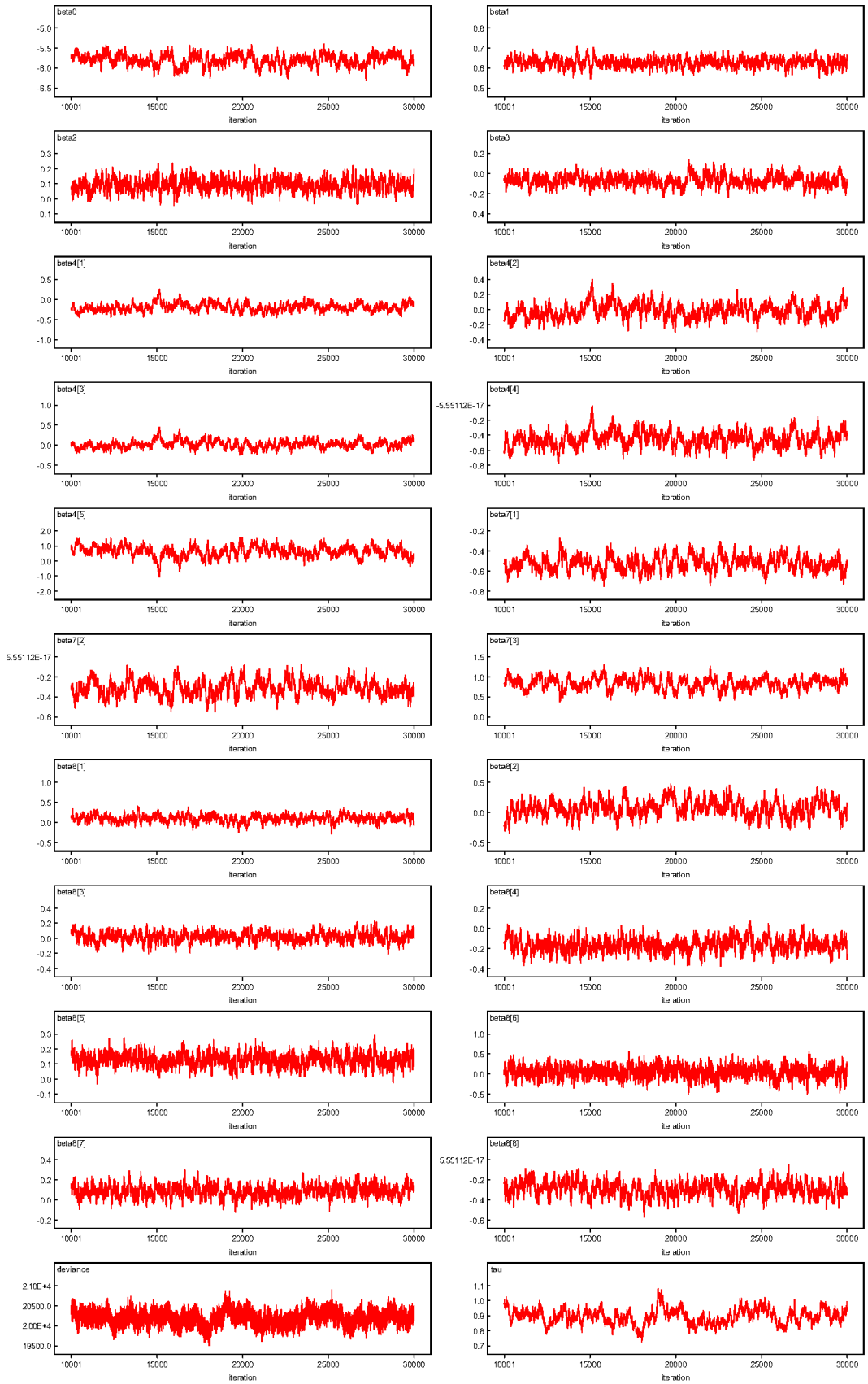


Figure 4.2: Trace plots for the parameters of BP-LN model for admissions related to respiratory diseases in 2016.

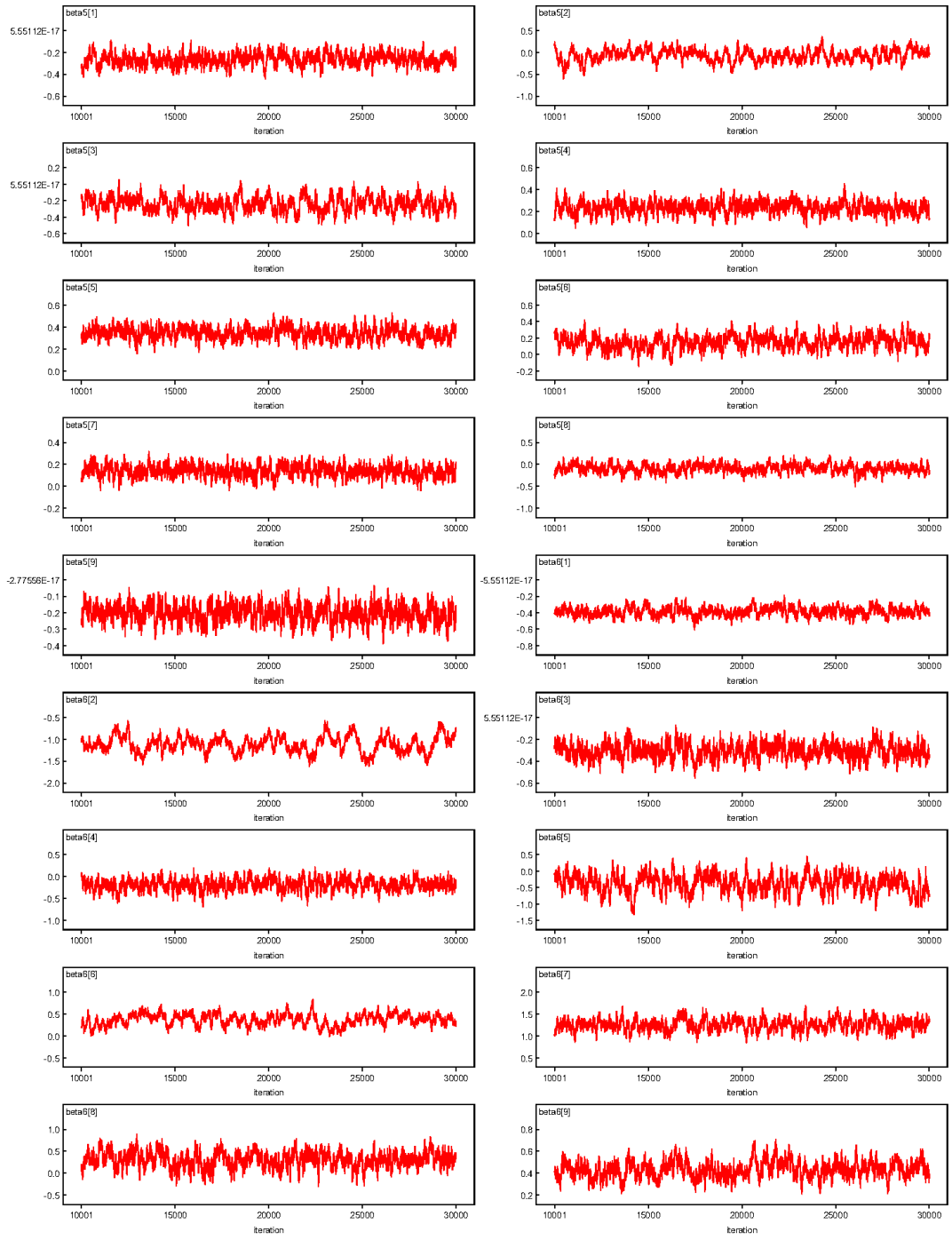


Figure 4.2: *Cont.*

The trace plots and the results of the BP-LN model suggest that the model has converged with low MCE for all the parameters.

4.1.2.3 Bayesian Poisson-Gamma model

Instead of the Log-normal prior assumption in the BP-LN model, for the BP-G model, a Gamma prior is assigned to the λ, thus yielding the following model structure:

$$A_i|\lambda_i \sim Poi(\lambda_i e_i), \qquad \text{for } i = 1, .., n \tag{4.11}$$

where $\mu_i = \lambda_i e_i$ with

$$\lambda_i|\phi, \xi_i \sim Gamma(\phi, \frac{\phi}{\xi_i}) \tag{4.12}$$

and ξ_i is defined as

$$\begin{aligned}\xi_i = \exp(\beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} \\ + \beta_{5,eeclass_i} + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i})\end{aligned} \tag{4.13}$$

with ϕ having a vague $Gamma(1, 0.001)$ prior. As detailed in Congdon (2019), this is equivalent to A_i having a negative binomial distributional assumption of the form

$$A_i \sim NB(\xi_i, \phi) \tag{4.14}$$

The posterior parameter estimates of the BP-G model are given in Table 4.12 and the trace plots for the different parameters over the 20,000 iterations are shown in Figure 4.3.

Coefficient	Description	Mean	SD	MC error	2.50%	50%	97.50%
β_0	(Intercept)	-5.318	0.106	0.009	-5.494	-5.304	-5.150
β_1	AGE	0.614	0.021	0.002	0.574	0.613	0.656
β_2	SEX 2-Female	0.078	0.038	0.003	0.009	0.077	0.152
β_3	UR 2-urban	-0.113	0.047	0.004	-0.196	-0.114	-0.013
	REGION1-Northeast	-0.124	0.079	0.006	-0.291	-0.118	0.008
	REGION2-Northcentral	0.024	0.072	0.006	-0.121	0.030	0.154
$\beta_{4,region_a}$	REGION3-South	0.048	0.071	0.006	-0.089	0.062	0.163
	REGION4-West	-0.403	0.086	0.007	-0.580	-0.395	-0.256
	REGION5-Unknown	0.455	0.276	0.023	0.026	0.406	0.963
	EECLASS1-Salary Non-Union	-0.276	0.047	0.004	-0.365	-0.275	-0.183
	EECLASS2-Salary Union	-0.056	0.171	0.014	-0.433	-0.024	0.243
	EECLASS3-Salary Other	-0.252	0.092	0.007	-0.430	-0.256	-0.077
	EECLASS4-Hourly Non-union	0.230	0.056	0.004	0.113	0.232	0.336
$\beta_{5,eeclass_b}$	EECLASS5-Hourly Union	0.344	0.062	0.005	0.234	0.343	0.469
	EECLASS6-Hourly Other	0.176	0.069	0.005	0.045	0.175	0.307
	EECLASS7-Non-union	0.125	0.054	0.004	0.020	0.126	0.233
	EECLASS8-Union	-0.083	0.079	0.006	-0.238	-0.079	0.058
	EECLASS9-Unknown	-0.208	0.050	0.003	-0.314	-0.207	-0.111
	EESTATU1-Active Full Time	-0.345	0.056	0.005	-0.445	-0.352	-0.232
	EESTATU2-Active Part Time or Seasonal	-1.130	0.163	0.014	-1.455	-1.097	-0.872
	EESTATU3-Early Retiree	-0.245	0.070	0.006	-0.359	-0.254	-0.101
	EESTATU4-Medicare Eligible Retiree	-0.095	0.091	0.007	-0.253	-0.100	0.102
$\beta_{6,eestatu_c}$	EESTATU5-Retiree (status unknown)	-0.579	0.233	0.019	-1.073	-0.557	-0.202
	EESTATU6-COBRA Continuee	0.393	0.163	0.014	0.107	0.385	0.672
	EESTATU7-Long Term Disability	1.317	0.136	0.011	1.090	1.321	1.583
	EESTATU8-Surviving Spouse/Depend.	0.305	0.202	0.017	-0.037	0.276	0.853
	EESTATU9-Unknown	0.378	0.072	0.005	0.231	0.378	0.520
	EMPREL1-Employee	-0.478	0.097	0.008	-0.623	-0.495	-0.277
$\beta_{7,emprel_d}$	EMPREL2-Spouse	-0.235	0.098	0.008	-0.373	-0.255	-0.020
	EMPREL3-Child/other	0.713	0.191	0.016	0.297	0.754	0.981
	PLANTYP2-Comprehensive Plan	0.093	0.071	0.006	-0.056	0.095	0.218
	PLANTYP3-EPO Plan	0.143	0.145	0.012	-0.105	0.120	0.410
	PLANTYP4-HMO Plan	0.021	0.052	0.004	-0.082	0.023	0.117
	PLANTYP5-Non-Cap PoS Plan	-0.172	0.070	0.005	-0.298	-0.178	-0.011
$\beta_{8,plantyp_e}$	PLANTYP6-PPO Plan	0.082	0.045	0.004	0.002	0.082	0.170
	PLANTYP7-Cap or Part Cap PoS Plan	0.094	0.101	0.008	-0.094	0.095	0.287
	PLANTYP8-CDHP	0.051	0.062	0.005	-0.078	0.045	0.168
	PLANTYP9-HDHP	-0.313	0.072	0.005	-0.452	-0.313	-0.164
ϕ	Dispersion	1.011	0.068	0.005	0.892	1.010	1.150

Table 4.12: Posterior estimates of the BP-G model for admissions related to respiratory diseases in 2016.

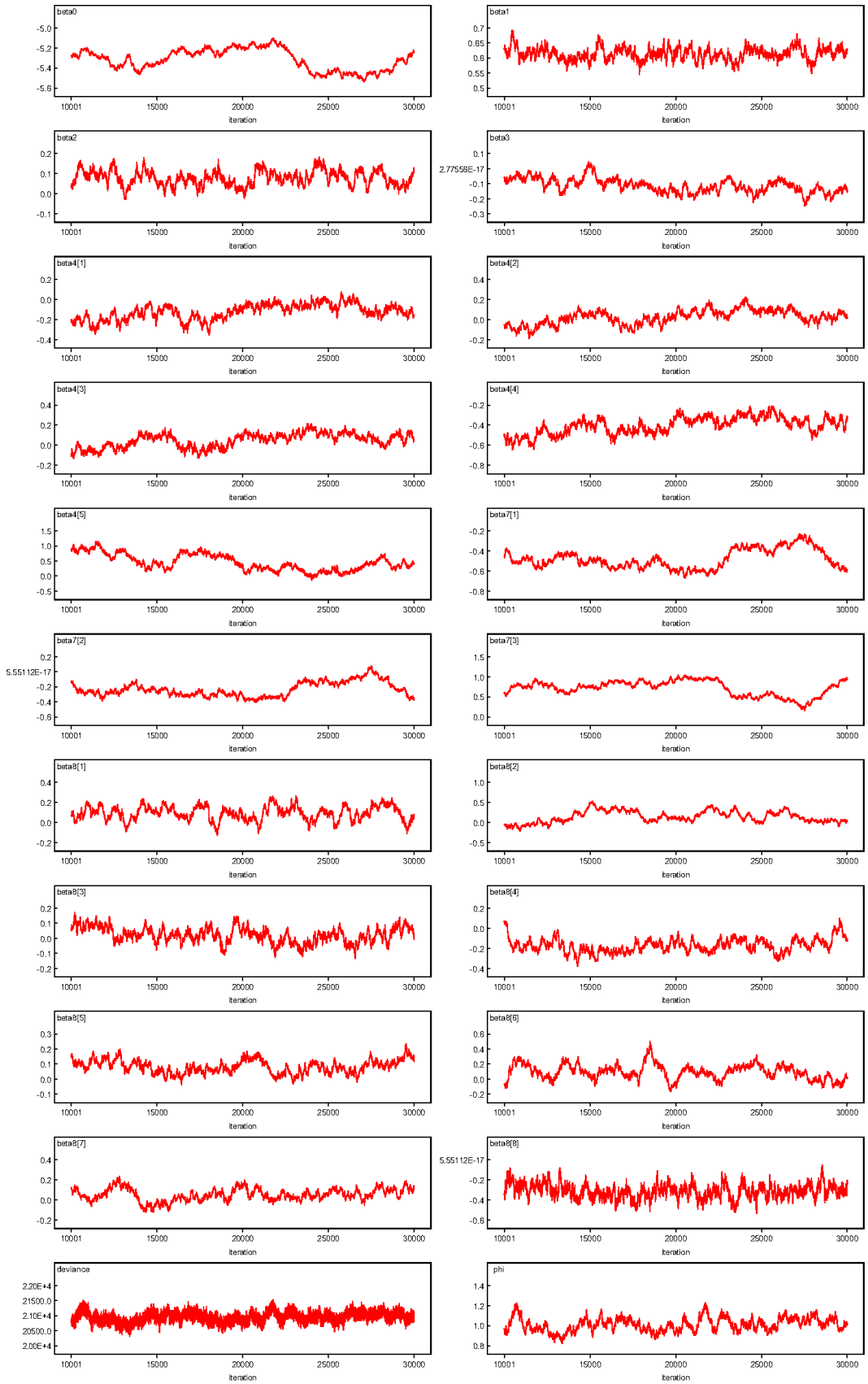


Figure 4.3: Trace plots for the parameters of the BP-G model for admissions related to respiratory diseases in 2016.

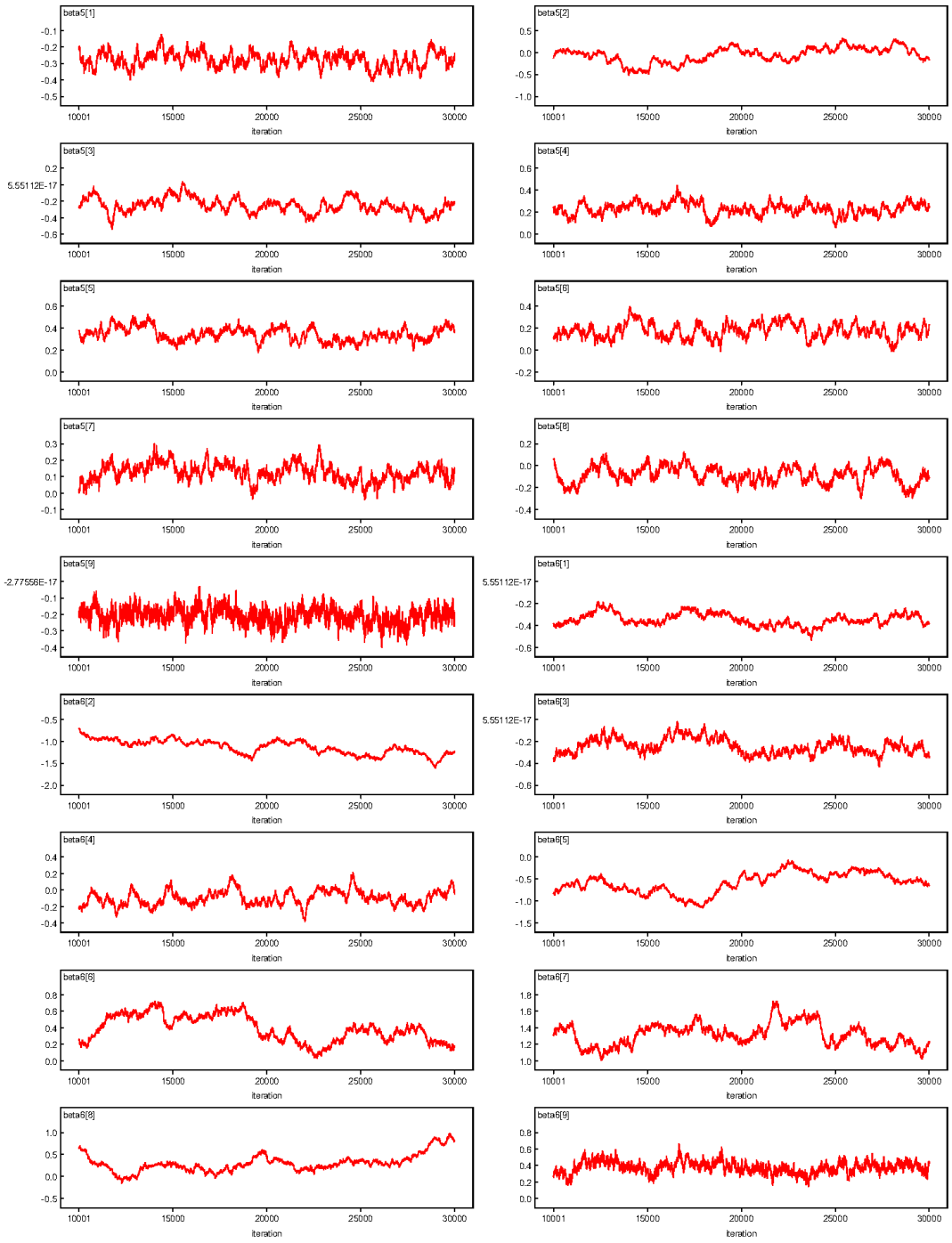


Figure 4.3: *Cont.*

The trace plots for the different parameters of the BP-G model indicate that the parameters have converged reasonably well with low MCE but the convergence is not as good as in the other models. A detailed evaluation of the various models discussed so far is carried out in the next part.

4.1.3 Comparative analysis of model outcomes

As part of the evaluation process, the model parameter estimates among different models are analysed and compared. The inferences obtained from the parameter estimates are also examined as part of this. Additionally, we consider the model fit by looking at the fitted values for specific risk profiles.

Table 4.13 shows the likelihood measures such as AIC, BIC and DIC for the different models. The AIC and BIC measures show that the NB regression model has a better fit compared to the Poisson regression model and is indicative of over-dispersion within the data. The DIC measure indicates that both the BP-LN and BP-G models outperform the BP model, with the BP-G model having a slightly better fit than the BP-LN model.

Evaluation criterion	Poisson	NB	BP	BP-LN	BP-G
AIC	24550.7	24109.9	-	-	-
BIC	24866.2	24435.0	-	-	-
DIC	-	-	24550.5	22852.3	22823.1

Table 4.13: AIC, BIC and DIC values for the different classical and Bayesian models for admissions related to respiratory diseases in the year 2016.

The parameter estimates from the different models, along with the associated variability measures, are shown in Table 4.14. The coefficient estimates of the Poisson and NB models are similar, also in terms of the level of significance for almost all the parameters. The estimates of the Poisson and NB models also seem comparable to the corresponding estimates from the Bayesian models, particularly for the highly significant parameters in the classical models.

As anticipated, the coefficient estimate for the age variable indicates that rates of admission increase with age. For all models, a sum-to-zero constraint was applied to all the categorical variables with more than two levels. The coefficient estimates for the various levels of the region covariate in the BP-G model as well as the other models imply that the admission rates for records with the region variable 'Unknown' are much higher than those for other regions. Among the other regions, the coefficient estimate for the West region (level=4) is distinctly negative compared

Model	poisson			NB			BP				BP-LN				BP-G			
Coefficient	Mean	SE	Signi	Mean	SE	Signi	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%
$\beta_{intercept}$	-5.053	0.126	***	-5.084	0.131	***	-5.103	0.112	-5.329	-4.900	-5.803	0.141	-6.096	-5.555	-5.318	0.106	-5.494	-5.150
β_{age}	0.604	0.019	***	0.612	0.022	***	0.605	0.019	0.568	0.642	0.626	0.022	0.582	0.670	0.614	0.021	0.574	0.656
$\beta_{sex2-female}$	0.095	0.031	**	0.082	0.037	*	0.096	0.030	0.036	0.155	0.093	0.040	0.015	0.172	0.078	0.038	0.009	0.152
$\beta_{ur2-urban}$	-0.160	0.041	***	-0.129	0.047	**	-0.158	0.042	-0.239	-0.077	-0.069	0.054	-0.175	0.040	-0.113	0.047	-0.196	-0.013
$\beta_{region1-northeast}$	-0.177	0.088	*	-0.207	0.090	*	-0.156	0.082	-0.306	0.014	-0.191	0.098	-0.361	0.016	-0.124	0.079	-0.291	0.008
$\beta_{region2-northcentral}$	-0.051	0.087		-0.058	0.088		-0.030	0.080	-0.175	0.138	-0.017	0.099	-0.195	0.189	0.024	0.072	-0.121	0.154
$\beta_{region3-south}$	-0.061	0.085		-0.041	0.086		-0.039	0.077	-0.180	0.120	0.024	0.096	-0.152	0.218	0.048	0.071	-0.089	0.163
$\beta_{region4-west}$	-0.498	0.091	***	-0.483	0.092	***	-0.476	0.087	-0.639	-0.295	-0.454	0.101	-0.645	-0.244	-0.403	0.086	-0.580	-0.256
$\beta_{region5-unknown}$	0.787	0.327	*	0.789	0.329	*	0.701	0.300	0.062	1.236	0.638	0.365	-0.100	1.298	0.455	0.276	0.026	0.963
$\beta_{eeclass1-salarynon-union}$	-0.282	0.046	***	-0.278	0.050	***	-0.283	0.047	-0.377	-0.190	-0.263	0.053	-0.368	-0.158	-0.276	0.047	-0.365	-0.183
$\beta_{eeclass2-salaryunion}$	-0.065	0.127		-0.047	0.129		-0.055	0.127	-0.305	0.187	-0.070	0.134	-0.348	0.177	-0.056	0.171	-0.433	0.243
$\beta_{eeclass3-salaryother}$	-0.254	0.080	**	-0.233	0.083	**	-0.258	0.082	-0.421	-0.102	-0.243	0.089	-0.417	-0.066	-0.252	0.092	-0.430	-0.077
$\beta_{eeclass4-hourlynon-union}$	0.218	0.045	***	0.224	0.050	***	0.217	0.045	0.129	0.307	0.236	0.053	0.129	0.340	0.230	0.056	0.113	0.336
$\beta_{eeclass5-hourlyunion}$	0.319	0.048	***	0.340	0.052	***	0.322	0.048	0.229	0.413	0.344	0.054	0.237	0.446	0.344	0.062	0.234	0.469
$\beta_{eeclass6-hourlyother}$	0.192	0.076	*	0.180	0.080	*	0.191	0.076	0.040	0.345	0.150	0.081	-0.007	0.307	0.176	0.069	0.045	0.307
$\beta_{eeclass7-non-union}$	0.121	0.045	**	0.122	0.051	*	0.122	0.046	0.033	0.209	0.140	0.052	0.036	0.240	0.125	0.054	0.020	0.233
$\beta_{eeclass8-union}$	-0.089	0.083		-0.085	0.089		-0.097	0.086	-0.265	0.075	-0.090	0.093	-0.276	0.089	-0.083	0.079	-0.238	0.058
$\beta_{eeclass9-unknown}$	-0.160	0.045	***	-0.223	0.052	***	-0.159	0.045	-0.247	-0.072	-0.204	0.052	-0.308	-0.108	-0.208	0.050	-0.314	-0.111
$\beta_{eestatu1-activefulltime}$	-0.386	0.054	***	-0.418	0.057	***	-0.376	0.058	-0.491	-0.263	-0.386	0.054	-0.489	-0.273	-0.345	0.056	-0.445	-0.232
$\beta_{eestatu2-activeparttimeorseasonal}$	-1.049	0.185	***	-1.082	0.187	***	-1.049	0.174	-1.399	-0.722	-1.113	0.190	-1.478	-0.730	-1.130	0.163	-1.455	-0.872
$\beta_{eestatu3-earlyretiree}$	-0.273	0.064	***	-0.303	0.069	***	-0.265	0.068	-0.396	-0.128	-0.306	0.068	-0.444	-0.173	-0.245	0.070	-0.359	-0.101
$\beta_{eestatu4-medicareeligibleretiree}$	-0.198	0.108	.	-0.217	0.123	.	-0.195	0.111	-0.417	0.016	-0.184	0.126	-0.437	0.057	-0.095	0.091	-0.253	0.102
$\beta_{eestatu5-retiree(statusunknown)}$	-0.302	0.242		-0.276	0.249		-0.341	0.263	-0.881	0.107	-0.374	0.276	-0.949	0.121	-0.579	0.233	-1.073	-0.205
$\beta_{eestatu6-COBRAcontinuee}$	0.386	0.128	**	0.361	0.130	**	0.396	0.130	0.147	0.639	0.369	0.132	0.099	0.601	0.393	0.163	0.107	0.672
$\beta_{eestatu7-longtermdisability}$	1.299	0.129	***	1.270	0.132	***	1.287	0.132	1.004	1.530	1.254	0.130	1.000	1.511	1.317	0.136	1.090	1.583
$\beta_{eestatu8-survivingspouse/depend}$	0.378	0.154	*	0.336	0.168	*	0.389	0.151	0.108	0.702	0.307	0.180	-0.075	0.646	0.305	0.202	-0.037	0.853
$\beta_{eestatu9-unknown}$	0.144	0.065	*	0.328	0.073	***	0.154	0.068	0.023	0.286	0.434	0.075	0.289	0.594	0.378	0.072	0.231	0.520
$\beta_{emprel1-employee}$	-0.552	0.069	***	-0.556	0.071	***	-0.543	0.070	-0.674	-0.393	-0.523	0.074	-0.659	-0.373	-0.478	0.097	-0.623	-0.277
$\beta_{emprel2-spouse}$	-0.298	0.070	***	-0.308	0.071	***	-0.288	0.070	-0.415	-0.144	-0.312	0.078	-0.458	-0.156	-0.235	0.098	-0.373	-0.020
$\beta_{emprel3-child/other}$	0.850	0.135	***	0.864	0.137	***	0.831	0.136	0.543	1.079	0.836	0.147	0.544	1.103	0.713	0.191	0.297	0.981
$\beta_{plantyp2-ComprehensivePlan}$	0.121	0.073	.	0.088	0.082		0.122	0.073	-0.019	0.265	0.099	0.089	-0.083	0.272	0.093	0.071	-0.056	0.218
$\beta_{plantyp3-EPOPlan}$	0.089	0.147		0.096	0.149		0.078	0.148	-0.228	0.366	0.071	0.136	-0.199	0.340	0.143	0.145	-0.105	0.410
$\beta_{plantyp4-HMOPlan}$	0.047	0.050		0.032	0.056		0.051	0.050	-0.049	0.147	0.018	0.063	-0.111	0.142	0.021	0.052	-0.082	0.117
$\beta_{plantyp5-Non-CapPoSPlan}$	-0.159	0.061	**	-0.153	0.067	*	-0.156	0.059	-0.272	-0.040	-0.165	0.066	-0.297	-0.033	-0.172	0.070	-0.298	-0.011
$\beta_{plantyp6-PPOPlan}$	0.043	0.035		0.091	0.040	*	0.048	0.036	-0.021	0.118	0.128	0.042	0.044	0.209	0.082	0.045	0.002	0.170
$\beta_{plantyp7-CaporPartCapPoSPlan}$	0.188	0.107	.	0.060	0.134		0.180	0.106	-0.037	0.381	0.053	0.142	-0.241	0.326	0.094	0.101	-0.094	0.287
$\beta_{plantyp8-CDHP}$	0.000	0.054		0.076	0.058		0.004	0.053	-0.100	0.111	0.088	0.060	-0.028	0.210	0.051	0.062	-0.078	0.168
$\beta_{plantyp9-HDHP}$	-0.329	0.068	***	-0.290	0.072	***	-0.327	0.067	-0.459	-0.197	-0.292	0.074	-0.436	-0.152	-0.313	0.072	-0.452	-0.165
deviance							24520	8.604	24500	24540	20190	168.3	19860	20520	20950	166.2	20630	21270
τ/ϕ				1.022	0.078						0.894	0.050	0.797	0.988	1.011	0.068	0.892	1.150

Table 4.14: Comparison of the model estimates of the Poisson, NB, BP, BP-LN and BP-G regression models for admissions related to respiratory diseases in 2016.

to the others, indicating lower admission rates in comparison to those regions.

Levels 1 and 2 in the UR variable represent rural and urban areas. Based on coefficient estimates, the admission rates due to respiratory diseases are higher for people in rural areas compared to people in urban areas. This is contrary to the expectation that people in rural areas are less subject to diseases related to respiratory diseases since they have access to cleaner air. This could be due to the underlying approach used to develop the 'UR' variable. The UR variable was created using MSA codes (see Section 3.3.1). If the individual's area of residence belonged to an MSA and had a corresponding MSA code, it was assumed that the person's residence belonged to an urban area. Alternatively, the individual was assumed to be from a rural area. However, even if an individual's residence did not belong to an MSA, this may not necessarily mean that the individual lives in a rural area, which might be causing the above-mentioned pattern.

The coefficient estimate for the gender covariate clearly shows that the admission rates for females are higher compared to males. This difference might be linked to potential variations in the prevalence of smoking habits, which is considered one of the main contributors to respiratory diseases (CDC (2021)). Additionally, it is worth noting that the coefficient estimates from the model with the SEX:EMPREL interaction term, identified while considering the relevant interaction terms (see Table A.6), show that when the interaction is taken into account, admission rates are lower among females.

In the case of all the models, according to the model estimates for the different levels of the EECLASS variable, it appears that individuals working on an hourly basis (levels 4, 5, and 6) have higher admission rates compared to those working on a salary basis (levels 1, 2, and 3). This difference could be attributed to the nature of work and the associated consequences arising from it. It could be the case that individuals working on an hourly basis are more likely to be carrying out manual labour rather than desk jobs. Furthermore, within the group of employees working on a salary or hourly basis, in both cases, unionised individuals have higher admis-

sion rates compared to non-unionised individuals. Studies indicate that unionised employees enjoy more benefits along with better utilisation of social insurance programs (Walters and Mishel (2003)). This implies that unionised employees are more likely to utilise the health plans to treat any instances of diseases, leading to more admissions. If so, this trend should prevail for admissions related to other diseases as well, which will be investigated later in this chapter.

As for the EESTATU (employment status) variable, the estimates from all the models suggest that the admission rates are considerably high for individuals with 'Long-term disability' (level=7), which is justifiable. Consecutively, the posterior estimate under the BP-G model for 'Consolidated Omnibus Budget Reconciliation Act (COBRA) continuee' (level=6) also indicates higher rates of admission for individuals belonging to that category compared to other levels of the EESTATU variable. COBRA is the law that facilitates the option for employees to continue with the health insurance plan under certain circumstances even after they leave employment. Another interesting observation is that the individuals with employment status 'Active part-time or seasonal' had a considerably negative estimate, implying lower admission rates. For the classical Poisson and NB regression models, the estimates of coefficients of almost all the levels of employment status variable were significant with the exception of the 'retiree-status unknown' (level=5) level, showcasing a clear impact of the different levels of EESTATU variable on the rates of admission related to respiratory diseases.

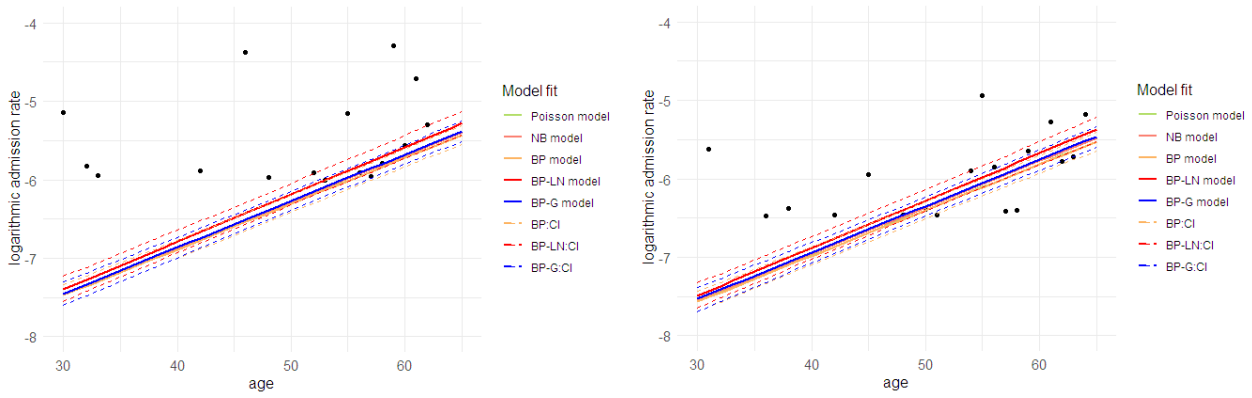
Regarding the coefficient estimates of the EMPREL (employee relation) covariate, the pattern is consistent over different models and indicates that the 'Child/other' level (level=3) exhibits higher rates of admission in comparison to other levels. The data under consideration is for the age group 30-65, and hence, the individuals with the EMPREL variable 'Child/other' belong to this age group. The definition of dependent varies for different insurers or employers. In general, apart from spouse/partner (considered a separate level), an unmarried child who meets certain eligibility requirements or elderly parents are considered as dependents. Since a

precise definition of the 'Child/ other' level is not available, it is not straightforward to make robust inferences using these estimates. One possible explanation could be that the 'Child/other' category mainly comprises elderly dependents. However, as mentioned before, there exist limitations to verify the same.

According to the model estimates, the admission rates are lowest for individuals in the 'High-Deductible Health Plan' (HDHP) (level=9). This might be due to the fact that the HDHP plan incentivises individuals to be more proactive in managing their health and seek care only when essential or have more restrictive coverage.

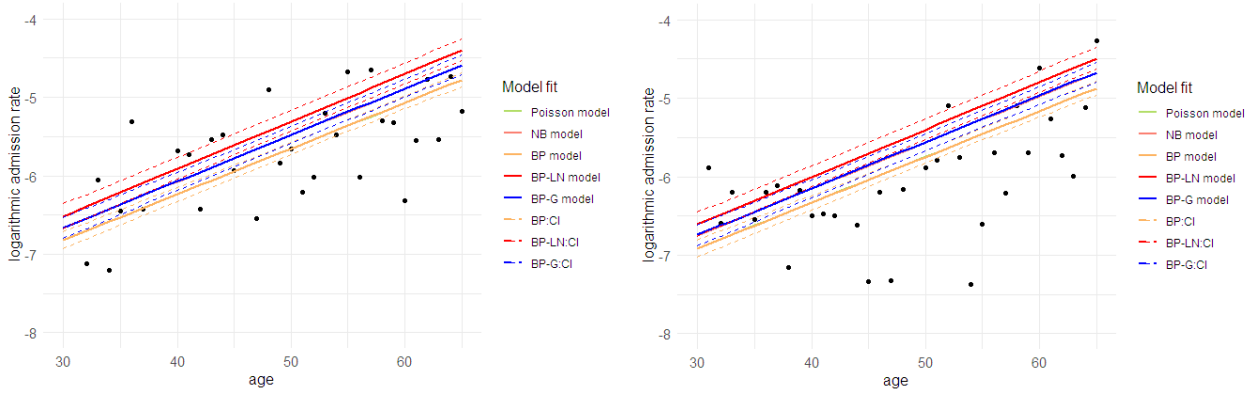
As mentioned before, in order to evaluate and compare the fit of the different models, the fitted curves from the different models are plotted over the crude rates for specific risk profiles (see Figures 4.4 and 4.5). Fitted curves from each of the models correspond to the logarithmic admission rates, which is, in fact, the linear predictor. For the Bayesian models, the 95% credible interval (CI) is also plotted alongside the fitted curves from each of the models. In general, $100(1-\alpha)\%$ credible intervals correspond to the intervals that contain $100(1-\alpha)\%$ of the posterior distribution of a parameter of interest and in this case, we consider 95% equal-tailed credible intervals.

For all the risk profiles, the fitted curves for all the models appear almost alike, particularly those of the Poisson and BP models, which are overlapping in all cases. A similar trend is observed for the NB and BP-G models as well. The fit of the BP-LN model is slightly different compared to all the other models. The CI interval seems relatively narrow for all the models, even though the intervals do not encompass the crude rates for all the ages. The excessive granularity of the chosen risk profiles could be the reason for this.



(a) (b)

Figure 4.4: Observed logarithmic admission rates and fitted rates under different models along with CI intervals for Bayesian models. Risk profile: UR-urban; REGION-south; EECLASS-salary non-union; EESTATU-active fill-time; EMPREL-employee; PLANTYP- PPO. (**a**) females; (**b**) males.



(a) (b)

Figure 4.5: Observed logarithmic admission rates and fitted rates under different models along with CI intervals for Bayesian models. Risk profile: UR-urban; REGION-south; EECLASS-unknown; EESTATU-unknown; EMPREL-employee; PLANTYP- PPO. (**a**) females; (**b**) males.

To better understand the fit of the models, the fitted rates of admission over some high-level (less granular) risk profiles were also considered. In order to achieve this, the predicted admission counts were aggregated over the risk profiles and divided by the associated exposure. We have $A_i \sim Poi(\lambda_i e_i)$ where for a model $m_t, t = 1, .., 5$,

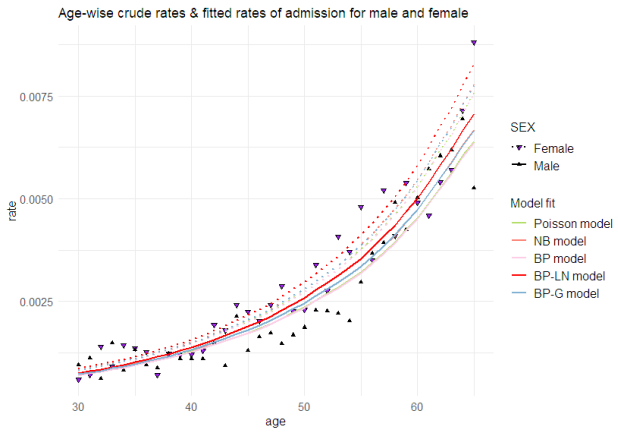
$\lambda_i^{m_t} = \exp(\boldsymbol{\beta}_{\boldsymbol{m_t}}^{\top} \boldsymbol{x_i})$. The predicted admission counts $\hat{A}_i^{m_t}$ is given by

$$\hat{A}_i^{m_t} = E[A_i | \lambda_i^{m_t}] = \exp(\boldsymbol{\beta}_{\boldsymbol{m_t}}^{\top} \boldsymbol{x_i}) \times e_i \tag{4.15}$$

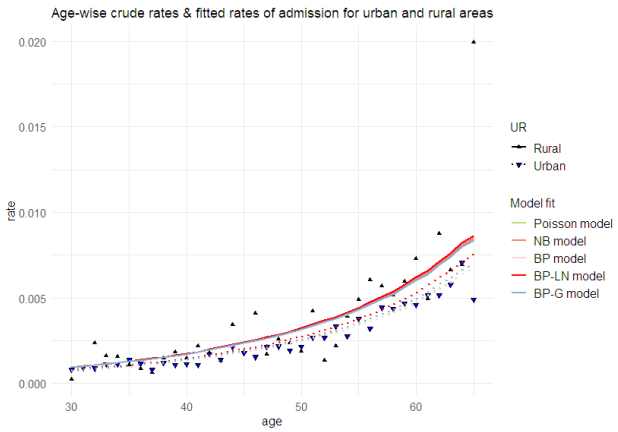
The age-wise fitted rates over different levels of the SEX variable are then estimated by

$$\hat{\lambda}_{age_k, sex_l}^{m_t} = \frac{\sum_{i=1}^{n} \hat{A}_{\{i\ ;\ age=k, sex=l\}}^{m_t}}{\sum_{i=1}^{n} e_{\{i\ ;\ age=k, sex=l\}}}$$

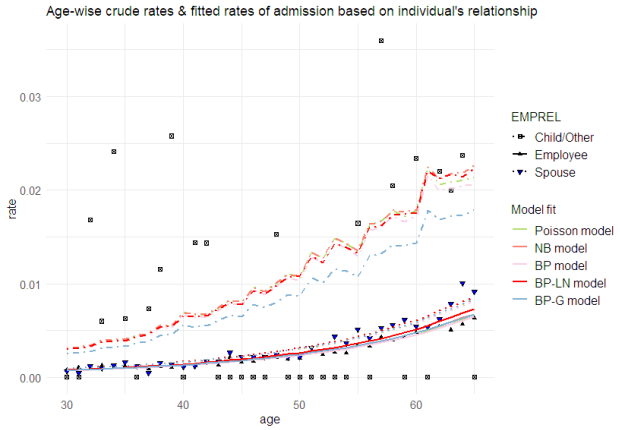
The fitted and crude rates of admission over the different levels of the SEX, UR and EMPREL variables are shown in Figure4.6 where all models seem to have a reasonably good fit. A slight variation is observed in the fitted rates from the BP-G model for 'Child/ other' (level =3) compared to that of the other models. Apart from this, an almost similar pattern is observed among different models in all instances.



(a)



(b)



(c)

Figure 4.6: Age-wise crude rates and fitted rates of admission over different variables. (**a**) SEX; (**b**) UR; (**c**) EMPREL.

4.1.4 Extending the models for combined data from multiple years

The models discussed so far were developed using the data from the year 2016. In this part, we extend those models for the combined data from all the available years (2016-2019) for admissions related to respiratory diseases. As previously mentioned, the aggregated risk profile data for all the years comprises 425,202 records. As part of these, the YEAR variable is introduced as an additional categorical covariate into the models. The linear predictor defined for the earlier models (Equations 4.7, 4.10 and 4.13) will be modified to include the year covariate, thus yielding:

$$\begin{aligned}\exp(\boldsymbol{\beta}^\top \boldsymbol{x_i}) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} + \beta_{5,eeclass_i} \\ + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} + \beta_{9,year_i}\end{aligned} \tag{4.16}$$

The YEAR categorical variable has four levels $\{2016, 2017, 2018, 2019\}$ with $\{\beta_{9,1}, ..., \beta_{9,4}\}$ being the corresponding set of coefficients. For the Bayesian models, as in the case of other β parameters, a vague normal prior was assigned to the β parameters associated with the year covariate:

$$\beta_{9,year_f} \sim N(0, 10^4), \qquad \text{for } f = 1, ..., 4.$$

For all the Bayesian models, 40,000 iterations were considered, within which 15,000 were discarded as burn-in period. Similar to other categorical variables, a sum-to-zero constraint was applied to the coefficient estimates of the different levels of the YEAR variable. The coefficient estimates from the various models are given in Table 4.15, and the trace plots for the BP, BP-LN and BP-G models are given in Figures A.1, A.2 and A.3 respectively.

Similar to the models for 2016 data, the coefficient estimates from various models are comparable. Subject to slight variations in the level of significance in classical

Model	Poisson			NB			BP				BP-LN				BP-G			
Coefficient	Mean	SE	Signi	Mean	SE	Signi	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%
$\beta_{intercept}$	-5.156	0.086	***	-5.174	0.088	***	-5.185	0.082	-5.350	-5.036	-5.894	0.106	-6.132	-5.704	-5.289	0.059	-5.396	-5.189
β_{age}	0.648	0.010	***	0.653	0.011	***	0.649	0.010	0.629	0.668	0.667	0.012	0.643	0.689	0.653	0.011	0.633	0.674
$\beta_{sex2-female}$	0.104	0.016	***	0.097	0.019	***	0.104	0.016	0.073	0.135	0.106	0.020	0.065	0.145	0.096	0.019	0.057	0.132
$\beta_{ur2-urban}$	-0.165	0.021	***	-0.142	0.024	***	-0.164	0.021	-0.205	-0.124	-0.107	0.024	-0.154	-0.059	-0.147	0.024	-0.194	-0.100
$\beta_{region1-northeast}$	-0.029	0.073		-0.052	0.074		-0.008	0.076	-0.147	0.153	-0.023	0.092	-0.182	0.187	0.057	0.059	-0.052	0.170
$\beta_{region2-northcentral}$	0.106	0.072		0.103	0.073		0.128	0.075	-0.008	0.283	0.143	0.091	-0.028	0.359	0.208	0.058	0.118	0.326
$\beta_{region3-south}$	0.081	0.072		0.094	0.072		0.102	0.074	-0.032	0.256	0.163	0.091	0.001	0.383	0.200	0.055	0.108	0.310
$\beta_{region4-west}$	-0.312	0.074	***	-0.319	0.074	***	-0.292	0.076	-0.431	-0.133	-0.296	0.092	-0.462	-0.069	-0.213	0.062	-0.317	-0.086
$\beta_{region5-unknown}$	0.154	0.283		0.174	0.284		0.070	0.294	-0.553	0.600	0.013	0.359	-0.872	0.649	-0.253	0.225	-0.690	0.114
$\beta_{eeclass1-salarynon-union}$	-0.340	0.023	***	-0.336	0.025	***	-0.341	0.023	-0.385	-0.296	-0.327	0.026	-0.378	-0.276	-0.340	0.023	-0.387	-0.296
$\beta_{eeclass2-salaryunion}$	-0.208	0.059	***	-0.197	0.061	**	-0.211	0.063	-0.337	-0.097	-0.206	0.062	-0.323	-0.085	-0.188	0.045	-0.282	-0.114
$\beta_{eeclass3-salaryother}$	-0.296	0.038	***	-0.287	0.040	***	-0.296	0.039	-0.376	-0.220	-0.302	0.045	-0.394	-0.216	-0.273	0.041	-0.345	-0.178
$\beta_{eeclass4-hourlynon-union}$	0.278	0.021	***	0.292	0.024	***	0.278	0.021	0.237	0.320	0.306	0.026	0.257	0.356	0.290	0.023	0.244	0.336
$\beta_{eeclass5-hourlyunion}$	0.343	0.024	***	0.349	0.026	***	0.343	0.025	0.296	0.393	0.360	0.028	0.305	0.415	0.350	0.025	0.297	0.400
$\beta_{eeclass6-hourlyother}$	0.239	0.035	***	0.234	0.037	***	0.238	0.034	0.169	0.305	0.220	0.039	0.136	0.289	0.216	0.039	0.126	0.292
$\beta_{eeclass7-non-union}$	0.112	0.025	***	0.112	0.027	***	0.114	0.025	0.066	0.163	0.113	0.028	0.057	0.165	0.111	0.025	0.060	0.159
$\beta_{eeclass8-union}$	-0.037	0.042		-0.026	0.044		-0.035	0.042	-0.119	0.048	-0.032	0.045	-0.118	0.054	-0.021	0.050	-0.133	0.067
$\beta_{eeclass9-unknown}$	-0.091	0.023	***	-0.140	0.026	***	-0.090	0.023	-0.135	-0.046	-0.132	0.026	-0.185	-0.082	-0.144	0.025	-0.191	-0.094
$\beta_{eestatu1-activefulltime}$	-0.431	0.030	***	-0.457	0.032	***	-0.427	0.026	-0.479	-0.376	-0.437	0.034	-0.503	-0.369	-0.439	0.021	-0.479	-0.400
$\beta_{eestatu2-activeparttimeorseasonal}$	-0.826	0.077	***	-0.856	0.078	***	-0.828	0.076	-0.977	-0.676	-0.843	0.073	-0.997	-0.707	-0.816	0.079	-0.958	-0.684
$\beta_{eestatu3-earlyretiree}$	-0.338	0.035	***	-0.372	0.038	***	-0.335	0.032	-0.398	-0.271	-0.380	0.039	-0.457	-0.305	-0.363	0.029	-0.424	-0.305
$\beta_{eestatu4-medicareeligibleretiree}$	-0.252	0.056	***	-0.270	0.065	***	-0.250	0.052	-0.360	-0.152	-0.263	0.069	-0.402	-0.133	-0.234	0.070	-0.373	-0.107
$\beta_{eestatu5-retiree(statusunknown)}$	-0.243	0.169		-0.229	0.174		-0.263	0.144	-0.524	0.046	-0.293	0.171	-0.641	0.034	-0.361	0.093	-0.532	-0.201
$\beta_{eestatu6-COBRAcontinuee}$	0.209	0.068	**	0.187	0.069	**	0.216	0.068	0.084	0.350	0.168	0.068	0.038	0.302	0.213	0.057	0.109	0.328
$\beta_{eestatu7-longtermdisability}$	1.337	0.063	***	1.311	0.065	***	1.343	0.067	1.206	1.470	1.292	0.066	1.163	1.427	1.308	0.062	1.187	1.416
$\beta_{eestatu8-survivingspouse/depend}$	0.439	0.074	***	0.429	0.080	***	0.434	0.075	0.290	0.578	0.408	0.087	0.230	0.569	0.417	0.079	0.279	0.582
$\beta_{eestatu9-unknown}$	0.105	0.036	**	0.258	0.040	***	0.110	0.033	0.044	0.175	0.349	0.042	0.263	0.431	0.275	0.038	0.196	0.343
$\beta_{emprel1-employee}$	-0.646	0.034	***	-0.646	0.035	***	-0.645	0.033	-0.713	-0.580	-0.625	0.034	-0.691	-0.561	-0.648	0.024	-0.701	-0.604
$\beta_{emprel2-spouse}$	-0.394	0.034	***	-0.407	0.035	***	-0.393	0.034	-0.461	-0.329	-0.423	0.034	-0.488	-0.357	-0.412	0.025	-0.466	-0.366
$\beta_{emprel3-child/other}$	1.040	0.066	***	1.053	0.067	***	1.038	0.065	0.914	1.173	1.047	0.064	0.922	1.175	1.060	0.045	0.978	1.154
$\beta_{plantyp2-ComprehensivePlan}$	0.201	0.037	***	0.181	0.042	***	0.203	0.037	0.129	0.273	0.211	0.044	0.124	0.296	0.175	0.038	0.094	0.244
$\beta_{plantyp3-EPOPlan}$	0.028	0.072		0.031	0.073		0.023	0.068	-0.109	0.159	-0.002	0.071	-0.138	0.142	0.033	0.074	-0.110	0.166
$\beta_{plantyp4-HMOPlan}$	-0.017	0.025		-0.016	0.028		-0.016	0.024	-0.065	0.031	-0.027	0.029	-0.083	0.031	-0.017	0.029	-0.078	0.037
$\beta_{plantyp5-Non-CapPoSPlan}$	0.001	0.030		0.022	0.033		0.002	0.030	-0.057	0.058	0.007	0.035	-0.062	0.079	0.020	0.032	-0.044	0.079
$\beta_{plantyp6-PPOPlan}$	-0.002	0.018		0.038	0.020	.	-0.001	0.017	-0.035	0.032	0.068	0.021	0.027	0.109	0.041	0.022	-0.006	0.078
$\beta_{plantyp7-CaporPartCapPoSPlan}$	0.026	0.054		-0.109	0.070		0.025	0.054	-0.080	0.129	-0.114	0.073	-0.254	0.027	-0.108	0.063	-0.229	0.000
$\beta_{plantyp8-CDHP}$	-0.022	0.026		0.026	0.029		-0.021	0.026	-0.071	0.030	0.028	0.030	-0.032	0.087	0.030	0.029	-0.033	0.086
$\beta_{plantyp9-HDHP}$	-0.215	0.030	***	-0.173	0.032	***	-0.214	0.030	-0.272	-0.157	-0.171	0.035	-0.240	-0.106	-0.174	0.032	-0.236	-0.112
$\beta_{year1-2016}$	0.059	0.013	***	0.058	0.016	***	0.059	0.013	0.032	0.085	0.063	0.017	0.030	0.096	0.060	0.016	0.027	0.090
$\beta_{year2-2017}$	0.000	0.014		-0.005	0.017		0.000	0.014	-0.027	0.029	-0.012	0.018	-0.048	0.021	-0.006	0.017	-0.039	0.029
$\beta_{year3-2018}$	0.000	0.013		0.000	0.015		0.000	0.013	-0.025	0.025	0.006	0.017	-0.027	0.038	0.001	0.015	-0.025	0.030
$\beta_{year4-2019}$	-0.058	0.014	***	-0.054	0.016	***	-0.058	0.014	-0.085	-0.032	-0.056	0.017	-0.090	-0.025	-0.056	0.016	-0.088	-0.027
deviance							97460	8.442	97440	97480	79550	352.8	78880	80270	82240	321.7	81610	82890
τ/ϕ				0.875	0.033						0.815	0.023	0.773	0.865	0.872	0.026	0.823	0.929

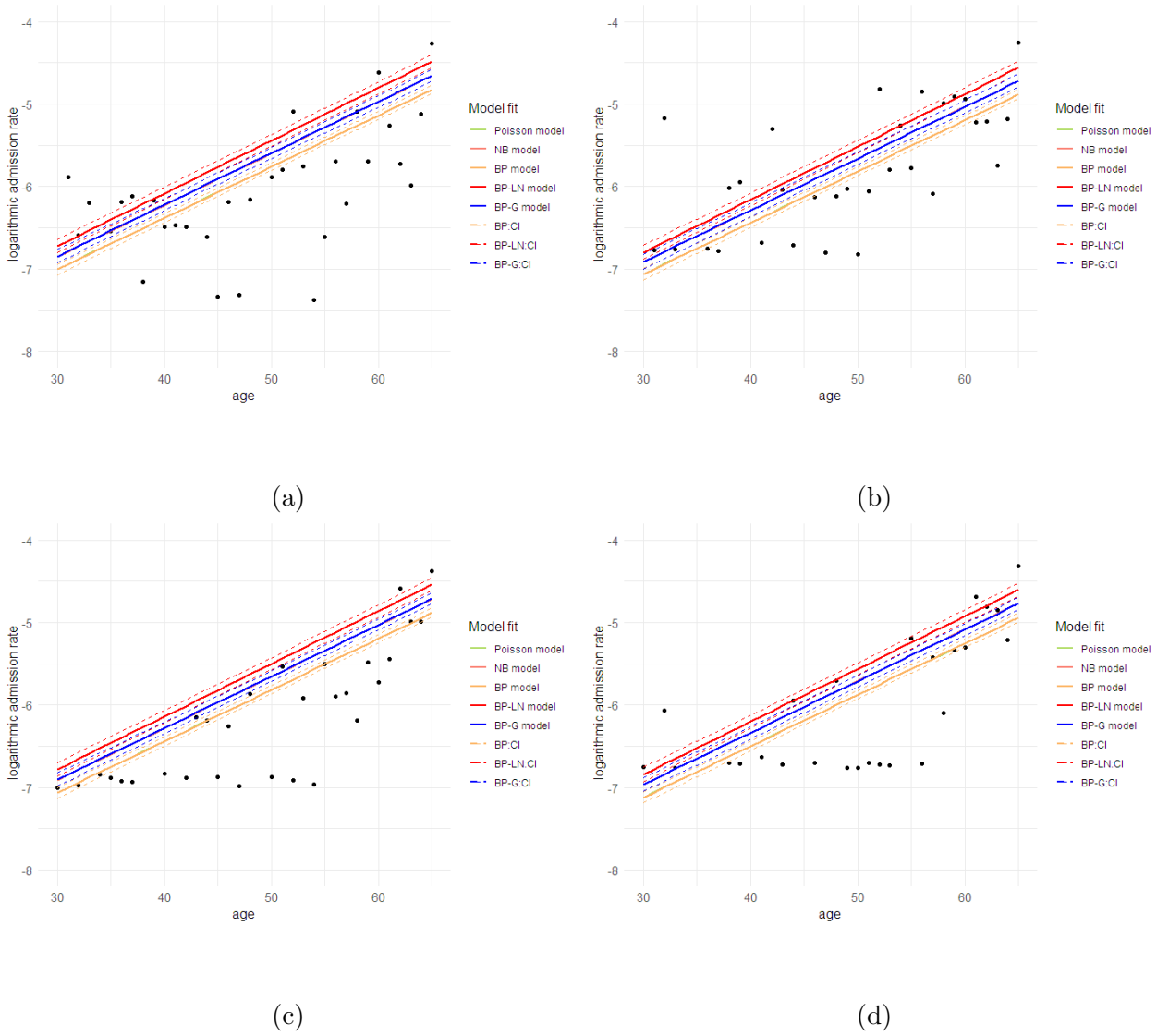
Table 4.15: Comparison of the model estimates of the Poisson, NB, BP, BP-LN and BP-G regression models for admissions related to respiratory diseases in the years 2016-2019.

models and sampling error for the Bayesian models, the quantitative order of the estimates for different levels of categorical variables does not vary to a great extent between models. Furthermore, the patterns perceived from the coefficient estimates of the models for 2016 data are generally observed here as well, implying that the patterns are consistent over different years. As for the YEAR covariate, the positive estimate for 2016 indicates that the admission rates were higher in 2016 compared to other years. Furthermore, the admission rate in 2019 is notably low, hinting at a potential downturn in the prevalence of chronic respiratory diseases. However, such a trend can only be definitively ascertained with further investigation as more data emerges. Also, the trace plots from all the Bayesian models show reasonable convergence for all the parameters.

The AIC and BIC values for Poisson and NB regression models, as well as the DIC values for the Bayesian models, are provided in Table 4.16. The different likelihood measures indicate the same pattern in the fit of various models, as observed for the 2016 data. Among the classical regression models, the NB model outperformed the Poisson model, and the BP-G model exhibited the best fit among the Bayesian models.

Evaluation criterion	Poisson	NB	BP	BP-LN	BP-G
AIC	97494.53	95695.38	-	-	-
BIC	97889.10	96100.91	-	-	-
DIC	-	-	97494.20	90460.40	89884.10

Table 4.16: AIC, BIC and DIC values for the different classical and Bayesian models for admissions related to respiratory diseases in the year 2016.



(a) (b)

(c) (d)

Figure 4.7: Observed logarithmic admission rates and fitted rates under different models along with CI intervals for Bayesian models for different years. Risk profile: SEX-male; UR-urban; REGION-south; EECLASS-unknown; EESTATU-unknown; EMPREL-employee; PLANTYP- PPO. (**a**) 2016; (**b**) 2017; (**c**) 2018; (**d**) 2019.

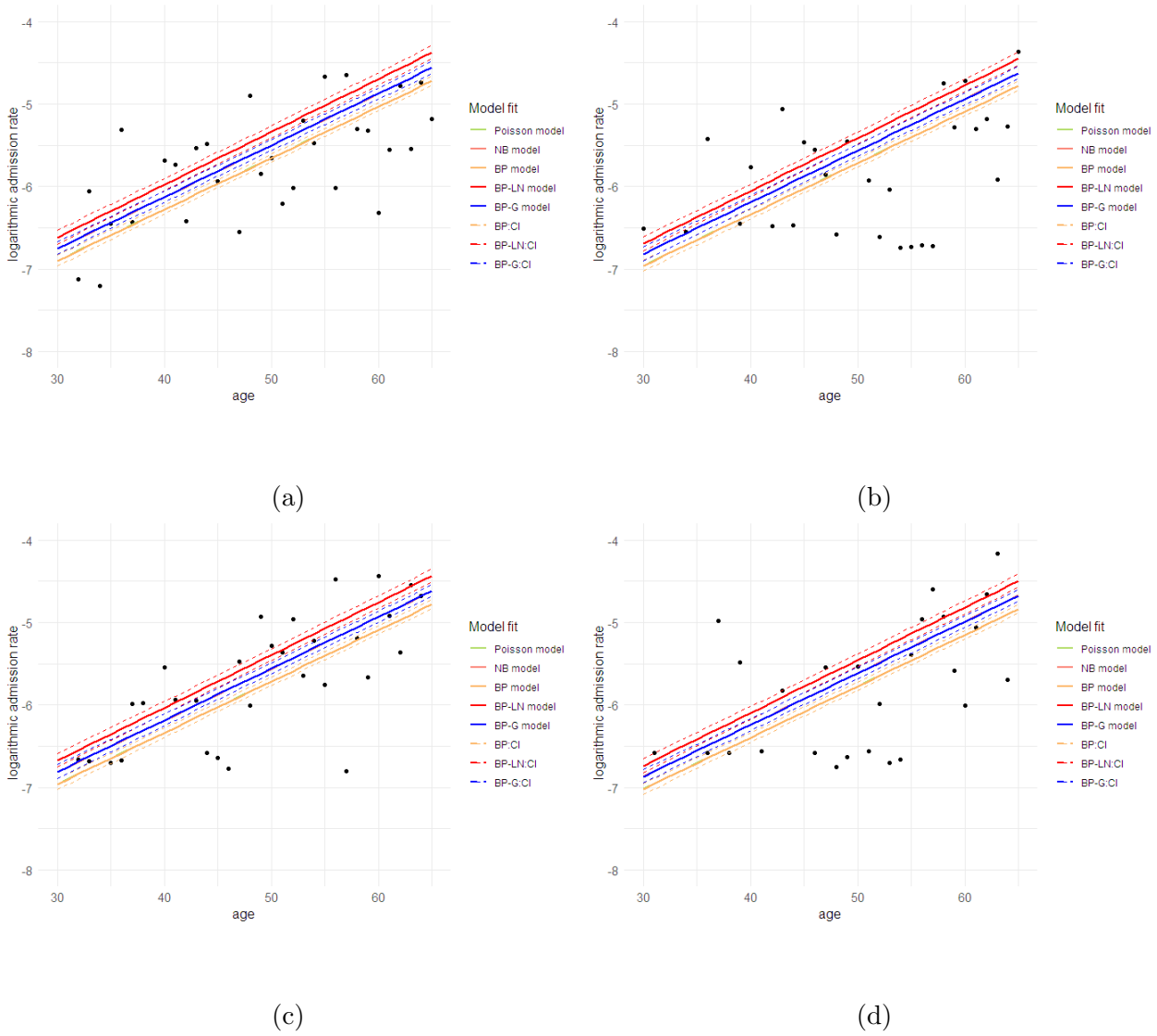


Figure 4.8: Observed logarithmic admission rates and fitted rates under different models along with CI intervals for Bayesian models for different years. Risk profile: SEX-female; UR-urban; REGION-south; EECLASS-unknown; EESTATU-unknown; EMPREL-employee; PLANTYP- PPO. (**a**) 2016; (**b**) 2017; (**c**) 2018; (**d**) 2019.

4.2 Admissions Related to Neoplasms

The crude rates of admission due to neoplasms over different ages indicate a marked difference between males and females (see Figure 3.5). Hence, the admission rates were modelled separately for males and females. As in the case of admissions related to respiratory diseases, risk profile level data was considered, which comprises

425,202 records, among which males and females make up 199,044 and 226,158 records, respectively. The combined data from all available years are considered for model development. The procedure for constructing the models for males and females is outlined here.

Initially, a full Poisson regression model with all the covariates was considered. A step-wise variable selection was then carried out using AIC and BIC measures. The next step was to determine the degree of age polynomial considered in order to account for any non-linear age effect as hinted from the crude rates. Finally, the relevant interaction terms were identified. The best-identified model structure was then used for all the models, both classical and Bayesian.

4.2.1 Rates of admission among males

The exploratory analysis described in Section 3.4 revealed that although the rates of admission due to neoplasm among males are increasing with age, they are lower compared to females in the initial ages (age 30 to 55) and followed by higher rates in the older ages. As stated in Chapter 3, this could be attributed to the prevalence of breast cancer among females in that age range. The distribution of admission related to neoplasms among males over the years 2016-2019 is shown in Table 4.17.

YEAR	# of admissions	Exposure
2016	2,671	778,350
2017	2,075	638,977
2018	3,028	958,146
2019	2,594	834,501

Table 4.17: Number of admissions related to neoplasms among males and the total exposure in different years.

4.2.1.1 Model formulation

Initially, a full Poisson model with all the covariates was considered. The following functional form was considered for the admission counts A_i^C:

$$A_i^C \sim Poi(\lambda_i e_i) \tag{4.17}$$

where

$$\begin{aligned}\log(\mu_i^{Pois}) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} + \beta_{5,eeclass_i} \\ + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} + \beta_{9,year_i} + o_i\end{aligned} \tag{4.18}$$

which is effectively the same model structure as that of the Poisson regression model defined for admissions related to respiratory diseases (Equation 4.2) with the additional YEAR covariate. The complete set of covariates and the corresponding coefficients are detailed in Tables 4.1 and 4.2. As before, a sum-to-zero constraint was applied for categorical variables with more than two levels and standardisation was applied to the age variable. The parameter estimates from the full Poisson regression model are shown in Table 4.18. As anticipated, the coefficient estimates for the age variable indicate that the rates of admission due to cancer among males are increasing with age. Also, individuals from urban areas have higher admission rates compared to individuals from rural areas. The summary of the the backward step-wise variable selection carried out using the AIC and BIC measures for the Poisson regression model are given in Tables 4.19 and Tables 4.20.

Coefficient	Estimate	Std. Error	z value	Pr(> |z|)	Signi.
$\beta_{intercept}$	-5.0876	0.0987	-51.5510	< 2e-16	***
β_{age}	0.8825	0.0136	64.7850	< 2e-16	***
$\beta_{ur2-urban}$	0.0289	0.0291	0.9930	0.3209	
$\beta_{region1-northeast}$	-0.2009	0.0637	-3.1530	0.0016	**
$\beta_{region2-northcentral}$	-0.1563	0.0631	-2.4770	0.0132	*
$\beta_{region3-south}$	-0.2521	0.0622	-4.0530	0.0001	***
$\beta_{region4-west}$	-0.3042	0.0644	-4.7230	0.0000	***
$\beta_{region5-unknown}$	0.9135	0.2419	3.7770	0.0002	***
$\beta_{eeclass1-salarynon-union}$	0.0167	0.0245	0.6840	0.4941	
$\beta_{eeclass2-salaryunion}$	0.0620	0.0612	1.0140	0.3106	
$\beta_{eeclass3-salaryother}$	0.0641	0.0402	1.5940	0.1109	
$\beta_{eeclass4-hourlynon-union}$	0.0864	0.0283	3.0590	0.0022	**
$\beta_{eeclass5-hourlyunion}$	0.0925	0.0315	2.9340	0.0033	**
$\beta_{eeclass6-hourlyother}$	0.0254	0.0476	0.5320	0.5945	
$\beta_{eeclass7-non-union}$	0.0878	0.0317	2.7750	0.0055	**
$\beta_{eeclass8-union}$	-0.0020	0.0500	-0.0400	0.9684	
$\beta_{eeclass9-unknown}$	-0.4331	0.0309	-13.9950	< 2e-16	***
$\beta_{eestatu1-activefulltime}$	-0.5297	0.0499	-10.6240	< 2e-16	***
$\beta_{eestatu2-activeparttimeorseasonal}$	-0.6493	0.0978	-6.6390	0.0000	***
$\beta_{eestatu3-earlyretiree}$	-0.3448	0.0553	-6.2370	0.0000	***
$\beta_{eestatu4-medicareeligibleretiree}$	-0.6734	0.1783	-3.7780	0.0002	***
$\beta_{eestatu5-retiree(statusunknown)}$	0.0926	0.1989	0.4660	0.6414	
$\beta_{eestatu6-COBRAcontinuee}$	0.2143	0.0856	2.5050	0.0123	*
$\beta_{eestatu7-longtermdisability}$	1.3051	0.0840	15.5280	< 2e-16	***
$\beta_{eestatu8-survivingspouse/depend}$	0.1231	0.2436	0.5050	0.6133	
$\beta_{eestatu9-unknown}$	0.4622	0.0565	8.1860	0.0000	***
$\beta_{emprel1-employee}$	-0.3365	0.0635	-5.3040	0.0000	***
$\beta_{emprel2-spouse}$	-0.1766	0.0641	-2.7570	0.0058	**
$\beta_{emprel3-child/other}$	0.5131	0.1254	4.0930	0.0000	***
$\beta_{plantyp2-ComprehensivePlan}$	0.0404	0.0518	0.7800	0.4356	
$\beta_{plantyp3-EPOPlan}$	0.1103	0.0856	1.2890	0.1975	
$\beta_{plantyp4-HMOPlan}$	-0.0412	0.0328	-1.2570	0.2087	
$\beta_{plantyp5-Non-CapPoSPlan}$	0.1431	0.0385	3.7170	0.0002	***
$\beta_{plantyp6-PPOPlan}$	0.0221	0.0231	0.9570	0.3384	
$\beta_{plantyp7-CaporPartCapPoSPlan}$	-0.3406	0.0820	-4.1540	0.0000	***
$\beta_{plantyp8-CDHP}$	0.0640	0.0329	1.9430	0.0520	.
$\beta_{plantyp9-HDHP}$	0.0020	0.0346	0.0560	0.9550	
$\beta_{year1-2016}$	0.0419	0.0170	2.4620	0.0138	*
$\beta_{year1-2017}$	-0.0104	0.0185	-0.5660	0.5717	
$\beta_{year1-2018}$	-0.0055	0.0163	-0.3370	0.7359	
$\beta_{year1-2019}$	-0.0260	0.0171	-1.5170	0.1292	

Table 4.18: Coefficient estimates based for the Poisson regression model with all the covariates for admissions related to neoplasms among males.

Model	# of estimated parameters	AIC
Step 1∼ AGE + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
- UR	34	56108
Full model	35	56109
- YEAR	32	56110
- PLANTYP	28	56130
- REGION	31	56135
- EMPREL	33	56166
- EECLASS	27	56332
- EESTATU	27	57221
- AGE	34	61123
Step 2∼ AGE + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
- YEAR	31	56109
- PLANTYP	27	56128
- REGION	30	56134
- EMPREL	32	56165
- EECLASS	26	56331
- EESTATU	26	57219
- AGE	33	61122

Table 4.19: Summary of the variable selection process for Poisson regression model for admissions related to cancer among males using AIC.

The AIC based approach suggested a model without the UR variable. In contrast, BIC suggested excluding the PLANTYP, YEAR, REGION and UR variables in succession. As the aim is also to analyse the impact of different risk factors on the rates of admission and also considering the disparity in the outcome from AIC and BIC-based approaches, it was decided to retain all the variables within the model.

Model	# of estimated parameters	BIC
Step 1 AGE + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
- PLANTYP	28	56415
- YEAR	32	56436
- REGION	31	56452
- UR	34	56455
Full model	35	56466
- EMPREL	33	56503
- EECLASS	27	56607
- EESTATU	27	57496
- AGE	34	61470
Step 2 AGE + UR + REGION + EECLASS + EESTATU + EMPREL + YEAR		
- YEAR	25	56386
- REGION	24	56401
- UR	27	56403
- EMPREL	26	56452
- EECLASS	20	56567
- EESTATU	20	57424
- AGE	27	61421
Step 3 AGE + UR + REGION + EECLASS + EESTATU + EMPREL		
- REGION	21	56372
- UR	24	56375
- EMPREL	23	56424
- EECLASS	17	56537
- EESTATU	17	57404
- AGE	24	61388
Step 4 AGE+ UR + EECLASS + EESTATU + EMPREL		
- UR	20	56360
- EMPREL	19	56410
- EECLASS	13	56530
- EESTATU	13	57393
- AGE	20	61383
Step 5 AGE + EECLASS + EESTATU + EMPREL		
- EMPREL	18	56398
- EECLASS	12	56518
- EESTATU	12	57381
- AGE	19	61372

Table 4.20: Summary of the variable selection process for Poisson regression model for admissions related to cancer among males using BIC.

The next step of the model formulation involved identifying the degree of the age

polynomial. The modified predictor with the age polynomial was of the form

$$\log(\mu_i^{Pois}) = \beta_0 + \sum_{k=1}^{r} \beta_{1,k} x_{i,1}^k + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} + \beta_{5,eeclass_i} + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} + \beta_{9,year_i} + o_i \tag{4.19}$$

where r is the degree of age polynomial that needs to be determined. The AIC and BIC values of models with different degrees for the age polynomial are shown in Table 4.21. Considering both AIC and BIC, the degree was set as three since the difference between the AIC value of the models with degrees three and four was negligible.

Degree	# of estimated parameter	BIC	AIC
r=1	35	56466.3	56109.3
r=2	36	56448.6	56081.4
r=3	37	56433.8	56056.4
r=4	38	56442.9	56055.3
r=5	39	56453.9	56056.0

Table 4.21: AIC and BIC values of the Poisson regression models for admissions due to cancer among males with different degrees ($r = 1, .., 5$) for the age polynomial.

For identifying potential interactions between different variables, a forward step-wise method using BIC was undertaken, the summary of which is shown in Table 4.22 and the step-wise details are provided in Table A.7. The BIC recommends the model with the interaction term between EECLASS and EESTATU variables. From a practical standpoint, due to the lack of in-depth understanding of the implications associated with various levels of the two variables combined with the inherent complexity in interpreting all potential interaction terms, the aforementioned interaction term was not included. Moreover, the number of parameters for such a model almost triples compared to the full model with an age polynomial. Consequently, the full model with the age polynomial was adopted as the optimal model.

Model	# of estimated parameters	BIC
Step 1: ∼ polynomial(AGE, degree = 3) + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP		
Full model w/ age polynomial	37	56433.8
+ EECLASS: EESTATU	101	55425.4

Table 4.22: Summary of the step-wise selection process for identifying relevant interaction terms of Poisson regression model for cancer related admissions in males using BIC.

The same functional form of the linear predictor, as shown in Equation 4.19, was also implemented in the rest of the models. Similar to the case of admissions related to respiratory diseases, in addition to the Poisson regression model, we consider the NB, BP, BP-LN and BP-G models for admissions due to neoplasms in males. In the case of Bayesian models, the prior specification of the model parameters was done following the same scheme mentioned for the models for admissions due to respiratory diseases.

4.2.1.2 Evaluation and comparison of model performance

In order to confirm the convergence, for all the Bayesian models, instead of a single chain, two separate chains or sets of samples were examined. For each of these chains, 30,000 iterations were considered, with the initial 10,000 samples being discarded as the burn-in phase. Also, for each chain, a separate set of initial values or starting points were assigned to the model parameters. The model parameters are then estimated using the post-burnin samples (20,000) from both chains collectively. The likelihood-based measures for different models shown in Table 4.23 indicate that among the classical models, the NB model outperforms the Poisson regression model. The DIC values show that the BP-LN model has the best fit, followed by the BP-G model. This is contrary to the pattern observed in the case of the models for admissions due to respiratory diseases.

Evaluation criterion	Poisson	NB	BP	BP-LN	BP-G
AIC	56,056.4	55,320.7			
BIC	56,433.8	55,708.4			
DIC			56,056.2	52,630.1	52,731.8

Table 4.23: AIC, BIC and DIC values for the different classical and Bayesian models for admissions related to neoplasms in males.

The coefficient estimates of the different models are shown in Table 4.24 and the trace plots for the BP, BP-LN and BP-G models are given in Figures 4.9, A.4 and A.5 respectively. The trace plots indicate that the model parameters of all the Bayesian models have converged in a satisfactory manner. The coefficient estimates from the different models do not vary to a great extent. Even in instances where the values slightly differ, in most cases, the significance codes for the classical models and the quantile values of the Bayesian models indicate that the estimates are not prominent. Since the BP-LN model had the best fit, the following discussion of coefficient estimates is carried out, relying primarily on the BP-LN model.

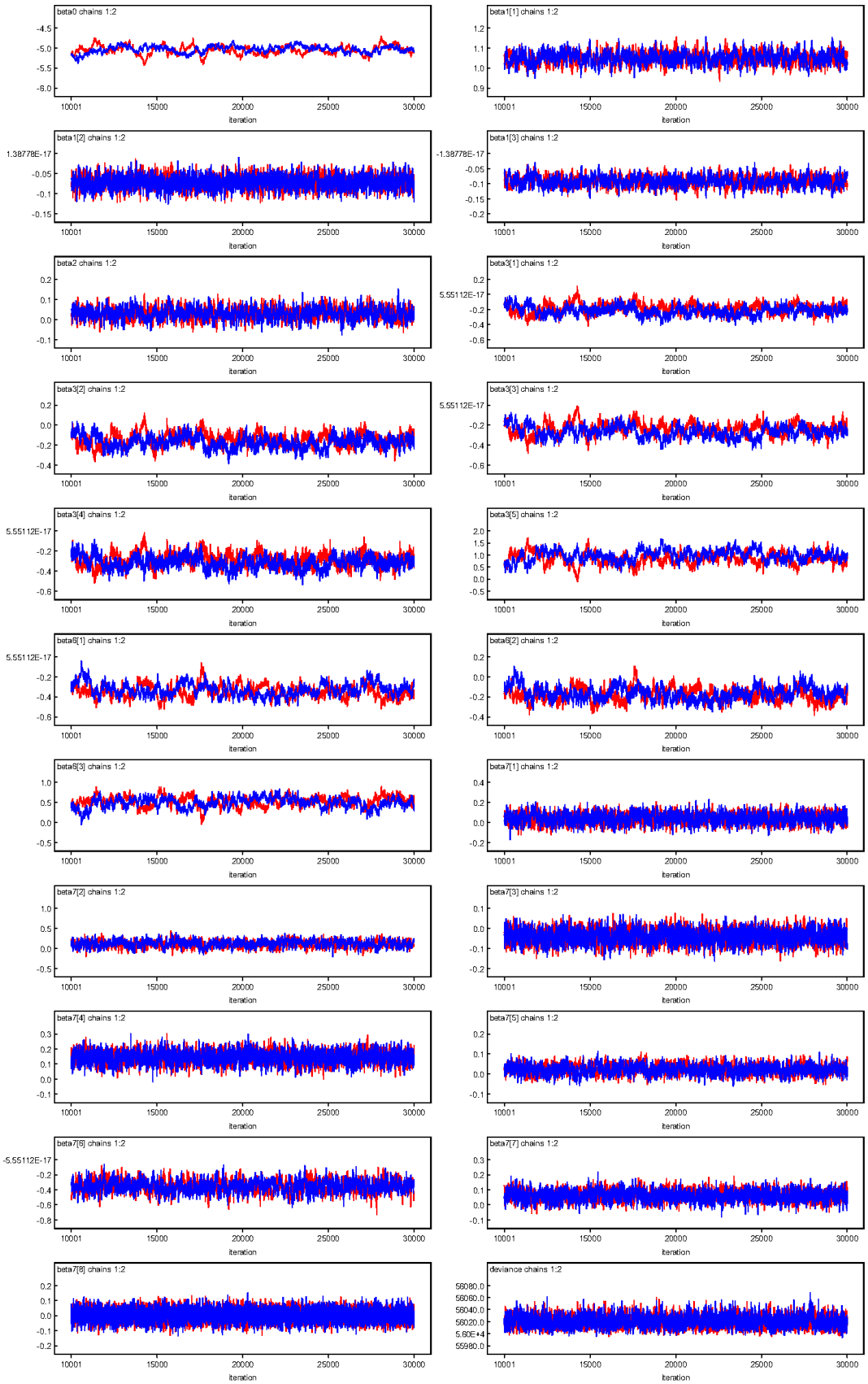


Figure 4.9: Trace plots for the parameters of the BP model for admissions related to neoplasms among males.

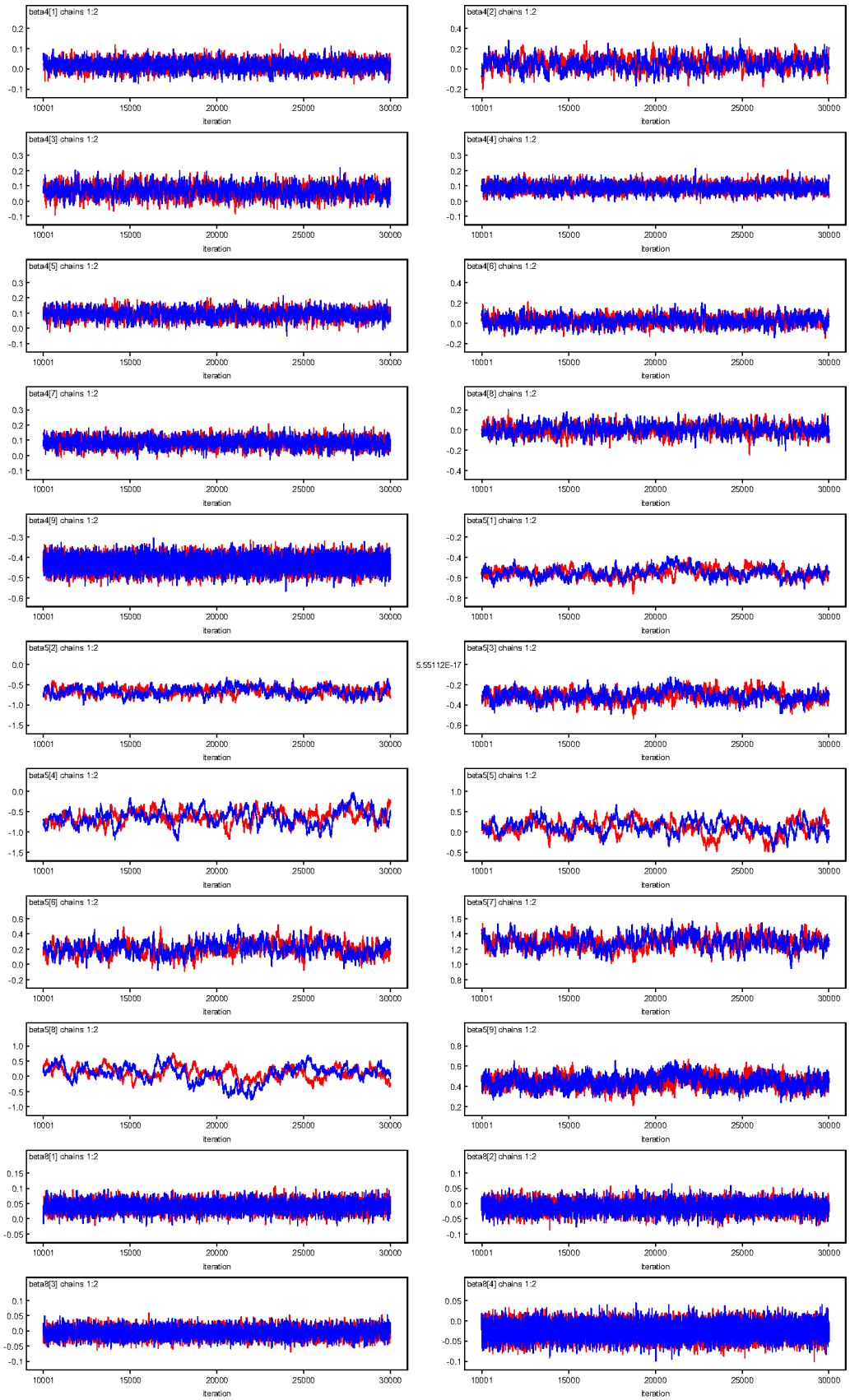


Figure 4.9: *Cont.*

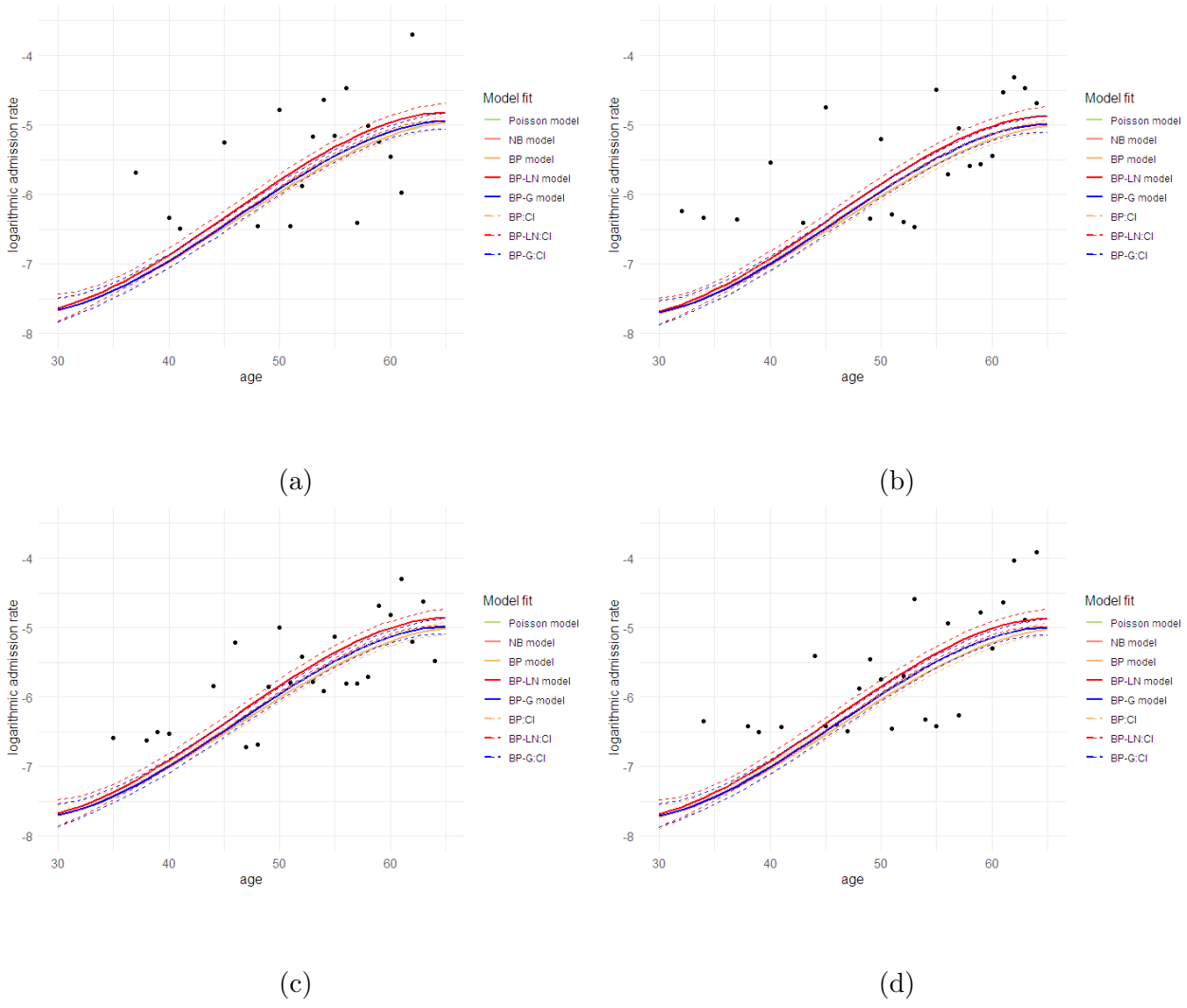
Model	Poisson			NB			BP				BP-LN				BP-G			
Coefficient	Mean	SE	Signi	Mean	SE	Signi	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%
$\beta_{intercept}$	-5.034	0.100	***	-5.105	0.103	***	-5.046	0.098	-5.248	-4.869	-5.744	0.108	-5.960	-5.532	-5.222	0.095	-5.372	-5.040
β_{age}	1.047	0.031	***	1.082	0.035	***	1.049	0.030	0.990	1.108	1.115	0.036	1.048	1.186	1.092	0.029	1.040	1.152
β_{age^2}	-0.070	0.015	***	-0.090	0.017	***	-0.071	0.015	-0.101	-0.041	-0.100	0.018	-0.135	-0.065	-0.088	0.018	-0.125	-0.055
β_{age^3}	-0.091	0.017	***	-0.102	0.019	***	-0.092	0.017	-0.124	-0.058	-0.106	0.021	-0.147	-0.065	-0.107	0.015	-0.139	-0.079
$\beta_{ur2-urban}$	0.030	0.029		0.092	0.033	**	0.030	0.029	-0.027	0.089	0.131	0.033	0.065	0.196	0.091	0.037	0.013	0.165
$\beta_{region1-northeast}$	-0.202	0.064	**	-0.220	0.066	***	-0.202	0.065	-0.326	-0.069	-0.212	0.071	-0.341	-0.066	-0.151	0.057	-0.261	-0.041
$\beta_{region2-northcentral}$	-0.155	0.063	*	-0.149	0.065	*	-0.155	0.064	-0.277	-0.027	-0.131	0.071	-0.260	0.018	-0.079	0.054	-0.184	0.019
$\beta_{region3-south}$	-0.252	0.062	***	-0.227	0.064	***	-0.252	0.063	-0.369	-0.124	-0.184	0.071	-0.315	-0.037	-0.162	0.055	-0.267	-0.063
$\beta_{region4-west}$	-0.302	0.064	***	-0.314	0.066	***	-0.302	0.066	-0.426	-0.168	-0.309	0.071	-0.441	-0.160	-0.248	0.056	-0.352	-0.137
$\beta_{region5-unknown}$	0.911	0.242	***	0.910	0.247	***	0.911	0.246	0.406	1.368	0.835	0.269	0.267	1.327	0.640	0.204	0.260	1.024
$\beta_{eeclass1-salarynon-union}$	0.017	0.024		0.016	0.028		0.017	0.025	-0.033	0.066	0.044	0.030	-0.014	0.103	0.012	0.027	-0.040	0.069
$\beta_{eeclass2-salaryunion}$	0.056	0.061		0.055	0.065		0.052	0.064	-0.076	0.178	0.035	0.067	-0.101	0.169	0.044	0.062	-0.071	0.155
$\beta_{eeclass3-salaryother}$	0.066	0.040	.	0.074	0.043	.	0.067	0.040	-0.010	0.145	0.065	0.045	-0.026	0.150	0.087	0.040	0.003	0.169
$\beta_{eeclass4-hourlynon-union}$	0.088	0.028	**	0.122	0.032	***	0.088	0.028	0.033	0.144	0.133	0.033	0.068	0.199	0.124	0.030	0.065	0.189
$\beta_{eeclass5-hourlyunion}$	0.091	0.032	**	0.110	0.035	**	0.091	0.032	0.028	0.154	0.120	0.036	0.048	0.191	0.116	0.034	0.046	0.176
$\beta_{eeclass6-hourlyother}$	0.025	0.048		0.015	0.050		0.025	0.047	-0.065	0.119	-0.005	0.052	-0.109	0.097	0.015	0.044	-0.065	0.106
$\beta_{eeclass7-non-union}$	0.087	0.032	**	0.089	0.035	*	0.086	0.032	0.024	0.147	0.089	0.037	0.014	0.160	0.086	0.034	0.020	0.158
$\beta_{eeclass8-union}$	0.001	0.050		0.005	0.055		0.004	0.052	-0.096	0.108	0.000	0.058	-0.116	0.113	0.000	0.052	-0.095	0.103
$\beta_{eeclass9-unknown}$	-0.432	0.031	***	-0.486	0.034	***	-0.431	0.031	-0.492	-0.371	-0.479	0.034	-0.545	-0.411	-0.483	0.033	-0.548	-0.419
$\beta_{eestatu1-activefulltime}$	-0.557	0.050	***	-0.594	0.052	***	-0.553	0.048	-0.644	-0.455	-0.559	0.052	-0.658	-0.448	-0.574	0.035	-0.636	-0.509
$\beta_{eestatu2-activeparttimeorseasonal}$	-0.667	0.098	***	-0.697	0.099	***	-0.667	0.100	-0.867	-0.483	-0.693	0.091	-0.878	-0.514	-0.730	0.076	-0.873	-0.594
$\beta_{eestatu3-earlyretiree}$	-0.318	0.055	***	-0.357	0.058	***	-0.314	0.053	-0.419	-0.208	-0.332	0.060	-0.447	-0.215	-0.333	0.042	-0.416	-0.251
$\beta_{eestatu4-medicareeligibleretiree}$	-0.644	0.178	***	-0.646	0.183	***	-0.627	0.181	-0.996	-0.283	-0.696	0.185	-1.089	-0.359	-0.567	0.118	-0.781	-0.333
$\beta_{eestatu5-retiree(statusunknown)}$	0.121	0.199		0.062	0.210		0.113	0.181	-0.271	0.453	0.015	0.207	-0.401	0.406	-0.061	0.127	-0.305	0.242
$\beta_{eestatu6-COBRAcontinuee}$	0.206	0.086	*	0.166	0.087	.	0.206	0.088	0.041	0.385	0.156	0.083	-0.009	0.318	0.172	0.084	0.036	0.340
$\beta_{eestatu7-longtermdisability}$	1.294	0.084	***	1.253	0.086	***	1.292	0.088	1.118	1.462	1.229	0.086	1.061	1.396	1.278	0.076	1.145	1.427
$\beta_{eestatu8-survivingspouse/depend}$	0.122	0.244		0.088	0.245		0.104	0.242	-0.496	0.523	0.045	0.219	-0.412	0.444	0.071	0.151	-0.214	0.396
$\beta_{eestatu9-unknown}$	0.442	0.057	***	0.725	0.060	***	0.446	0.056	0.340	0.558	0.836	0.060	0.718	0.952	0.744	0.049	0.649	0.839
$\beta_{emprel1-employee}$	-0.342	0.064	***	-0.311	0.064	***	-0.336	0.066	-0.450	-0.198	-0.287	0.063	-0.408	-0.161	-0.292	0.090	-0.447	-0.124
$\beta_{emprel2-spouse}$	-0.178	0.064	**	-0.214	0.065	**	-0.173	0.066	-0.291	-0.038	-0.231	0.065	-0.355	-0.102	-0.198	0.095	-0.354	-0.023
$\beta_{emprel3-child/other}$	0.520	0.126	***	0.525	0.127	***	0.509	0.130	0.240	0.735	0.518	0.126	0.267	0.755	0.490	0.184	0.149	0.798
$\beta_{plantyp2-ComprehensivePlan}$	0.041	0.052		0.048	0.059		0.042	0.051	-0.057	0.141	0.058	0.060	-0.064	0.173	0.034	0.061	-0.084	0.154
$\beta_{plantyp3-EPOPlan}$	0.112	0.086		0.108	0.087		0.114	0.084	-0.061	0.269	0.083	0.091	-0.099	0.262	0.101	0.096	-0.097	0.291
$\beta_{plantyp4-HMOPlan}$	-0.041	0.033		-0.033	0.037		-0.040	0.033	-0.103	0.023	-0.035	0.039	-0.111	0.044	-0.028	0.036	-0.099	0.047
$\beta_{plantyp5-Non-CapPoSPlan}$	0.145	0.039	***	0.138	0.043	**	0.146	0.039	0.069	0.221	0.115	0.046	0.025	0.201	0.151	0.046	0.062	0.236
$\beta_{plantyp6-PPOPlan}$	0.022	0.023		0.076	0.027	**	0.023	0.023	-0.022	0.069	0.112	0.029	0.055	0.168	0.086	0.028	0.034	0.144
$\beta_{plantyp7-CaporPartCapPoSPlan}$	-0.343	0.082	***	-0.498	0.099	***	-0.350	0.082	-0.511	-0.193	-0.506	0.109	-0.726	-0.301	-0.522	0.118	-0.771	-0.312
$\beta_{plantyp8-CDHP}$	0.062	0.033	.	0.107	0.036	**	0.062	0.034	-0.004	0.129	0.116	0.039	0.041	0.192	0.117	0.039	0.030	0.193
$\beta_{plantyp9-HDHP}$	0.002	0.035		0.054	0.038		0.003	0.035	-0.064	0.071	0.058	0.040	-0.020	0.139	0.060	0.040	-0.016	0.140
$\beta_{year1-2016}$	0.041	0.017	*	0.033	0.020		0.041	0.017	0.008	0.074	0.035	0.023	-0.010	0.079	0.033	0.019	-0.004	0.070
$\beta_{year2-2017}$	-0.011	0.018		-0.009	0.021		-0.011	0.018	-0.047	0.026	-0.016	0.024	-0.062	0.029	-0.008	0.020	-0.048	0.031
$\beta_{year3-2018}$	-0.005	0.016		-0.008	0.019		-0.005	0.016	-0.038	0.027	-0.005	0.020	-0.044	0.035	-0.008	0.018	-0.042	0.029
$\beta_{year4-2019}$	-0.025	0.017		-0.016	0.020		-0.025	0.017	-0.059	0.009	-0.015	0.021	-0.056	0.026	-0.017	0.021	-0.057	0.026
deviance							56020	8.666	56000	56040	46670	292.1	46090	47240	48590	249.1	48100	49080
τ/ϕ				1.152	0.060						0.925	0.041	0.847	1.009	1.139	0.052	1.042	1.247

Table 4.24: Comparison of the model estimates of the Poisson, NB, BP, BP-LN and BP-G regression models for admissions due to neoplasms among males.

The coefficient estimates associated with the age polynomial from all the models indicate varying trends over different ages. The positive linear coefficient and negative coefficients for quadratic and cubic terms indicate that the increase in admission rates with age slows down or even decreases at older ages. The plot of the crude rates of admission related to neoplasms in males (see Figure 3.5) did reveal a decrease in admissions rate at older ages for males. Additionally, the coefficient estimate indicates a higher admission rates for individuals from urban areas than rural areas.

As in the case of respiratory diseases, cancer-related admissions are lowest among males from west region. The admissions rate is considerably high for males with region 'Unknown' (level =5). Contrary to this, in the case of the EECLASS variable, the admission rates are lowest for the 'Unknown' level (level=9). Similar to respiratory diseases, for neoplasms in males, the admission rates are higher among individuals with long-term disability (EESTATU level=7). Furthermore, the pattern observed in the coefficient estimates for the EMPREL variable is also similar to that of respiratory diseases. For the PLANTYP variable, individuals with the capitated or partially capitated plan types (level=7) have considerably lower admission rates compared to other plan types. The coefficient estimates for the different levels of the YEAR covariate indicate that the admission rates do not vary to a great extent over different years for any given risk profile.

For a better understanding of the models' fit, the logarithmic admission rates obtained from the different models were considered for specific risk profiles. Two separate risk profiles were considered over different years and are shown in Figures 4.10 and 4.11. Compared to the risk profile shown in 4.11, the models fit better the risk profile depicted in 4.10.



(a) (b)

(c) (d)

Figure 4.10: Observed logarithmic admission rates due to neoplasms and fitted rates under different models along with CI intervals for Bayesian models for different years. Risk profile: SEX-male; UR-urban; REGION-south; EECLASS-salary non-union; EESTATU-active fill-time; EMPREL-employee; PLANTYP- PPO. (**a**) 2016; (**b**) 2017; (**c**) 2018; (**d**) 2019.

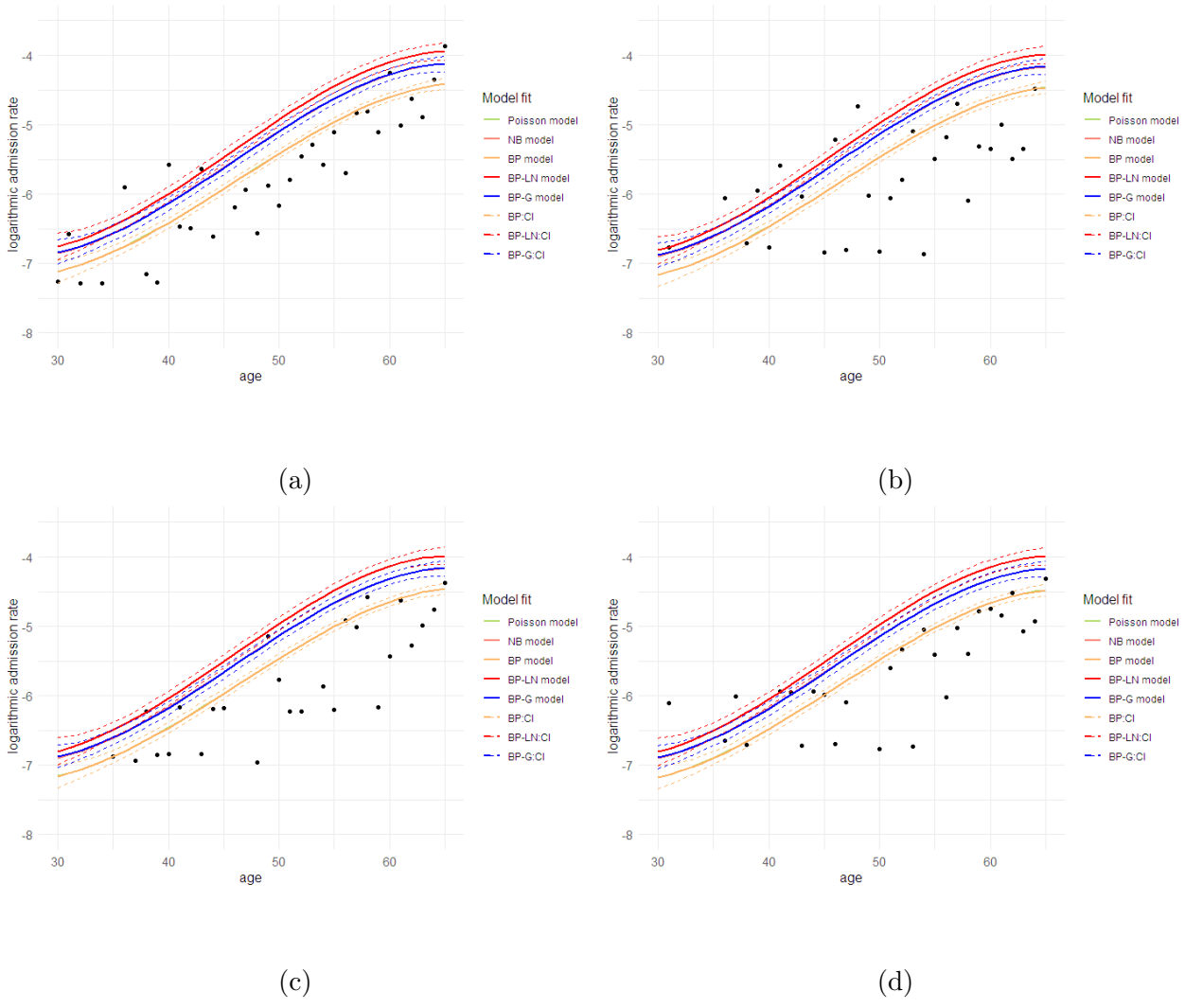


Figure 4.11: Observed logarithmic admission rates due to neoplasms and fitted rates under different models along with CI intervals for Bayesian models for different years. Risk profile: SEX-male; UR-urban; REGION-south; EECLASS-unknown; EESTATU-unknown; EMPREL-employee; PLANTYP- PPO.(**a**) 2016; (**b**) 2017; (**c**) 2018; (**d**) 2019.

4.2.2 Rates of admission among females

The steps undertaken for formulating the models for admissions due to neoplasms among males were undertaken sequentially for females as well. The details of which are provided in Appendix A.4. The optimal model structure was identified by considering likelihood-based measures while also giving due consideration to other aspects, such as complexity. The model structure thus identified was the same as that of the models for cancer-related admissions in males. Thus, the linear predictor for all the

models was of the form

$$\exp(\boldsymbol{\beta}^\top \boldsymbol{x_i}) = \beta_0 + \sum_{k=1}^{r} \beta_{1,k} x_{i,1}^k + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_{4,region_i} + \beta_{5,eeclass_i} + \beta_{6,eestatu_i} + \beta_{7,emprel_i} + \beta_{8,plantyp_i} + \beta_{9,year_i} \tag{4.20}$$

were $r = 3$. For the Bayesian models, 30,000 iterations from two chains were considered, and the initial 10,000 iterations from each chain were discarded as a burn-in period. Among the Bayesian models, based on DIC, the BP-G model has better fit than the other Bayesian models (see Table 4.25). This is contrary to what was observed previously in the case models for admissions among males but is inline with that of the models for admissions related to respiratory diseases.

Evaluation criterion	Poisson	NB	BP	BP-LN	BP-G
AIC	77,352.0	76,453.7			
BIC	77,734.2	76,846.2			
DIC			77,352.1	73,773.9	73,728.1

Table 4.25: AIC, BIC and DIC values for the different classical and Bayesian models for admissions related to neoplasms in females.

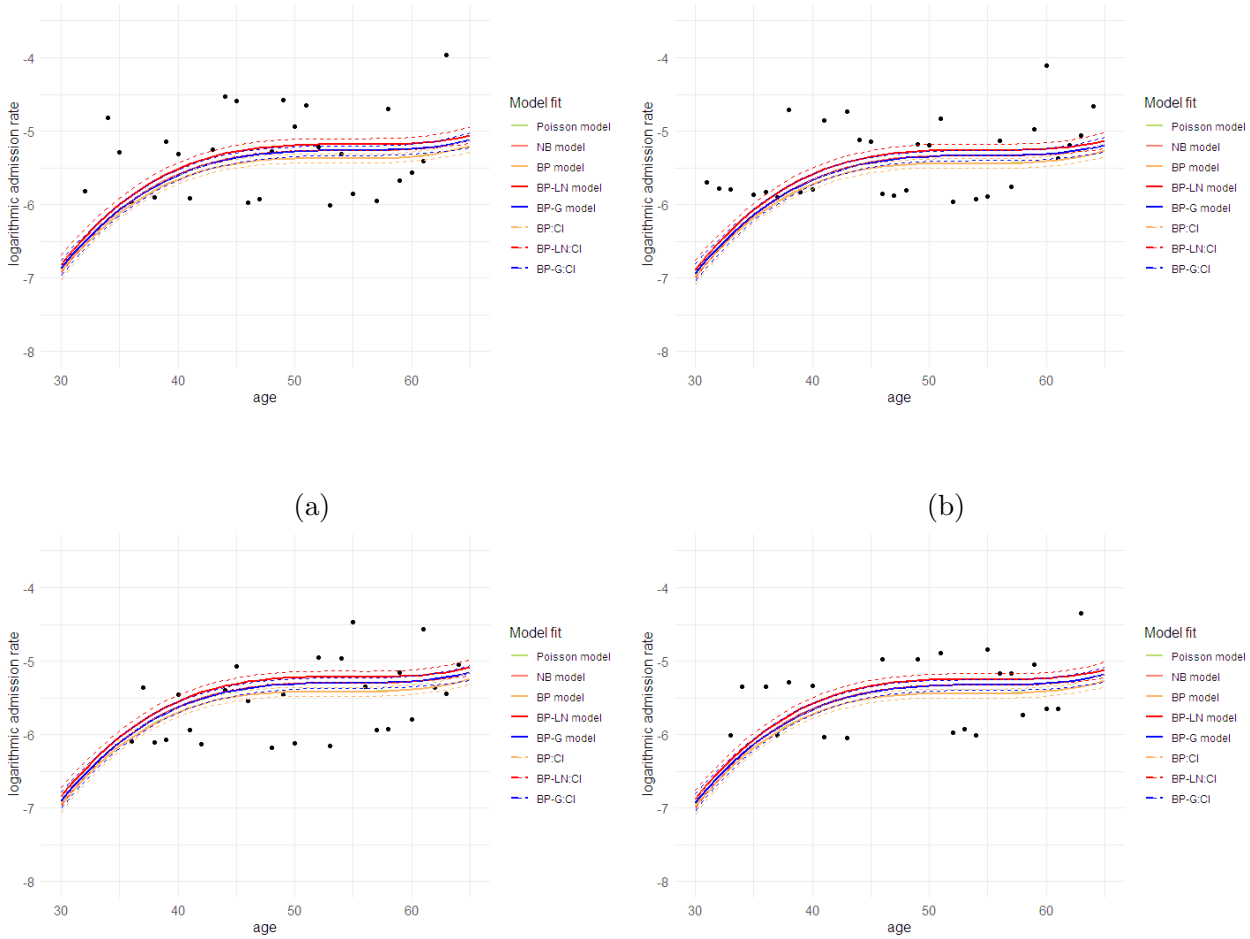
The trace plots for the BP, BP-LN and BP-G models (see Figures A.6, A.7 and A.8) showed that the parameters have converged for all the models. The coefficient estimates of the different models are shown in Table 4.26.

The coefficient estimates for the age polynomial terms are reflective of the decrease in rates of admission in the older ages for females. An interesting observation is that the coefficient estimate for the urban area is positive, indicating higher rates of admission among females from urban areas than from rural areas. Although a similar pattern was observed for males (see Table 4.24), the difference between urban and rural was minimal, and the estimate was not prominent for some of the models. For the rest of the categorical variables, the pronounced patterns observed between the different levels of the variables from the models for males are also visible within the models for females. As before, we also consider the fitted curves for a specific

Model	poisson			NB			BP				BP-LN				BP-G			
Coefficient	Mean	SE	Signi	Mean	SE	Signi	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%	Mean	SD	2.50%	97.50%
$\beta_{intercept}$	-4.522	0.087	***	-4.624	0.089	***	-4.514	0.086	-4.695	-4.355	-5.085	0.096	-5.283	-4.906	-4.713	0.079	-4.865	-4.547
β_{age}	0.056	0.021	**	0.088	0.025	***	0.057	0.021	0.015	0.099	0.087	0.026	0.036	0.137	0.094	0.019	0.055	0.132
β_{age^2}	-0.164	0.012	***	-0.182	0.014	***	-0.165	0.012	-0.188	-0.142	-0.188	0.014	-0.216	-0.160	-0.181	0.012	-0.204	-0.155
β_{age^3}	0.136	0.013	***	0.126	0.014	***	0.135	0.013	0.110	0.161	0.127	0.015	0.099	0.156	0.124	0.012	0.103	0.149
$\beta_{ur2-urban}$	0.140	0.025	***	0.227	0.028	***	0.139	0.025	0.089	0.188	0.251	0.031	0.190	0.312	0.237	0.025	0.192	0.295
$\beta_{region1-northeast}$	-0.108	0.060	.	-0.138	0.061	*	-0.113	0.062	-0.226	0.011	-0.120	0.066	-0.243	0.020	-0.132	0.055	-0.244	-0.009
$\beta_{region2-northcentral}$	-0.199	0.060	***	-0.203	0.061	***	-0.205	0.061	-0.317	-0.080	-0.195	0.063	-0.311	-0.059	-0.193	0.057	-0.308	-0.074
$\beta_{region3-south}$	-0.192	0.059	**	-0.149	0.060	*	-0.197	0.060	-0.307	-0.075	-0.116	0.063	-0.230	0.020	-0.139	0.054	-0.245	-0.032
$\beta_{region4-west}$	-0.346	0.061	***	-0.376	0.062	***	-0.351	0.062	-0.463	-0.222	-0.373	0.066	-0.492	-0.231	-0.369	0.056	-0.484	-0.251
$\beta_{region5-unknown}$	0.844	0.231	***	0.866	0.234	***	0.866	0.237	0.387	1.294	0.804	0.246	0.268	1.244	0.834	0.211	0.377	1.259
$\beta_{eeclass1-salarynon-union}$	-0.081	0.021	***	-0.073	0.024	**	-0.081	0.022	-0.123	-0.038	-0.057	0.025	-0.106	-0.007	-0.074	0.024	-0.125	-0.027
$\beta_{eeclass2-salaryunion}$	0.008	0.059		0.002	0.060		0.005	0.059	-0.107	0.126	-0.013	0.062	-0.139	0.110	-0.015	0.066	-0.144	0.101
$\beta_{eeclass3-salaryother}$	0.049	0.032		0.072	0.034	*	0.051	0.032	-0.012	0.113	0.070	0.036	0.001	0.143	0.071	0.038	-0.005	0.147
$\beta_{eeclass4-hourlynon-union}$	0.082	0.023	***	0.081	0.026	**	0.083	0.023	0.038	0.127	0.094	0.028	0.039	0.149	0.086	0.027	0.032	0.139
$\beta_{eeclass5-hourlyunion}$	0.130	0.029	***	0.153	0.031	***	0.129	0.029	0.072	0.186	0.155	0.031	0.095	0.217	0.153	0.032	0.092	0.215
$\beta_{eeclass6-hourlyother}$	0.170	0.035	***	0.170	0.037	***	0.170	0.035	0.099	0.237	0.158	0.039	0.080	0.235	0.168	0.037	0.104	0.251
$\beta_{eeclass7-non-union}$	0.099	0.025	***	0.091	0.028	**	0.099	0.026	0.047	0.149	0.097	0.030	0.038	0.154	0.090	0.027	0.040	0.145
$\beta_{eeclass8-union}$	0.023	0.042		0.046	0.045		0.024	0.043	-0.062	0.106	0.039	0.046	-0.053	0.126	0.061	0.050	-0.032	0.169
$\beta_{eeclass9-unknown}$	-0.479	0.025	***	-0.542	0.028	***	-0.479	0.025	-0.529	-0.430	-0.542	0.027	-0.595	-0.490	-0.538	0.027	-0.591	-0.485
$\beta_{eestatu1-activefulltime}$	-0.392	0.041	***	-0.430	0.042	***	-0.390	0.040	-0.465	-0.311	-0.417	0.043	-0.503	-0.334	-0.376	0.045	-0.465	-0.304
$\beta_{eestatu2-activeparttimeorseasonal}$	-0.546	0.076	***	-0.594	0.077	***	-0.546	0.075	-0.692	-0.400	-0.609	0.079	-0.770	-0.453	-0.560	0.068	-0.718	-0.444
$\beta_{eestatu3-earlyretiree}$	-0.284	0.048	***	-0.307	0.050	***	-0.281	0.048	-0.375	-0.187	-0.312	0.052	-0.414	-0.213	-0.259	0.047	-0.353	-0.178
$\beta_{eestatu4-medicareeligibleretiree}$	-0.372	0.077	***	-0.390	0.082	***	-0.377	0.078	-0.526	-0.225	-0.372	0.086	-0.551	-0.210	-0.314	0.074	-0.489	-0.190
$\beta_{eestatu5-retiree(statusunknown)}$	-0.438	0.258	.	-0.450	0.261	.	-0.443	0.262	-0.938	0.034	-0.463	0.277	-1.015	0.078	-0.917	0.257	-1.298	-0.435
$\beta_{eestatu6-COBRAcontinuee}$	0.352	0.073	***	0.307	0.074	***	0.352	0.072	0.214	0.499	0.291	0.079	0.132	0.442	0.364	0.068	0.224	0.481
$\beta_{eestatu7-longtermdisability}$	1.167	0.082	***	1.115	0.083	***	1.168	0.084	0.999	1.323	1.086	0.084	0.918	1.244	1.180	0.086	0.984	1.339
$\beta_{eestatu8-survivingspouse/depend}$	-0.116	0.101		-0.193	0.106	.	-0.113	0.103	-0.320	0.084	-0.209	0.105	-0.410	-0.005	-0.115	0.063	-0.241	0.006
$\beta_{eestatu9-unknown}$	0.628	0.046	***	0.942	0.048	***	0.630	0.046	0.544	0.719	1.006	0.049	0.911	1.101	0.998	0.052	0.901	1.099
$\beta_{emprel1-employee}$	-0.327	0.047	***	-0.302	0.047	***	-0.330	0.045	-0.417	-0.242	-0.281	0.049	-0.375	-0.182	-0.290	0.044	-0.376	-0.208
$\beta_{emprel2-spouse}$	-0.506	0.047	***	-0.524	0.047	***	-0.509	0.045	-0.596	-0.422	-0.527	0.049	-0.621	-0.432	-0.513	0.044	-0.600	-0.433
$\beta_{emprel3-child/other}$	0.832	0.092	***	0.825	0.092	***	0.839	0.087	0.667	1.009	0.808	0.095	0.619	0.991	0.802	0.085	0.650	0.969
$\beta_{plantyp2-ComprehensivePlan}$	0.089	0.047	.	0.111	0.050	*	0.089	0.047	-0.002	0.179	0.114	0.050	0.016	0.211	0.108	0.045	0.015	0.193
$\beta_{plantyp3-EPOPlan}$	0.098	0.072		0.098	0.073		0.101	0.074	-0.051	0.239	0.076	0.076	-0.077	0.222	0.085	0.066	-0.039	0.218
$\beta_{plantyp4-HMOPlan}$	-0.077	0.026	**	-0.088	0.029	**	-0.077	0.026	-0.127	-0.026	-0.082	0.031	-0.142	-0.023	-0.087	0.034	-0.149	-0.020
$\beta_{plantyp5-Non-CapPoSPlan}$	0.102	0.030	***	0.119	0.033	***	0.103	0.030	0.045	0.164	0.106	0.036	0.037	0.175	0.125	0.032	0.058	0.187
$\beta_{plantyp6-PPOPlan}$	-0.050	0.019	**	0.029	0.021		-0.050	0.018	-0.086	-0.013	0.055	0.023	0.010	0.100	0.033	0.024	-0.014	0.076
$\beta_{plantyp7-CaporPartCapPoSPlan}$	-0.115	0.057	*	-0.358	0.074	***	-0.118	0.056	-0.229	-0.010	-0.366	0.081	-0.527	-0.209	-0.366	0.070	-0.491	-0.216
$\beta_{plantyp8-CDHP}$	0.004	0.027		0.065	0.030	*	0.004	0.027	-0.051	0.056	0.069	0.031	0.008	0.131	0.074	0.031	0.012	0.131
$\beta_{plantyp9-HDHP}$	-0.051	0.029	.	0.023	0.032		-0.052	0.029	-0.109	0.006	0.028	0.035	-0.037	0.097	0.029	0.032	-0.034	0.092
$\beta_{year1-2016}$	0.043	0.014	**	0.041	0.017	*	0.043	0.014	0.016	0.071	0.043	0.018	0.008	0.077	0.041	0.017	0.010	0.075
$\beta_{year2-2017}$	-0.022	0.015		-0.027	0.017		-0.022	0.015	-0.051	0.008	-0.030	0.019	-0.067	0.005	-0.029	0.018	-0.062	0.007
$\beta_{year3-2018}$	0.002	0.013		0.006	0.016		0.001	0.014	-0.025	0.028	0.010	0.016	-0.023	0.042	0.007	0.017	-0.027	0.038
$\beta_{year4-2019}$	-0.023	0.014		-0.020	0.016		-0.023	0.014	-0.051	0.005	-0.023	0.017	-0.055	0.012	-0.020	0.017	-0.053	0.013
deviance							77320	8.58	77300	77330	66590	310.1	65990	67210	68440	308.3	67820	69050
τ/ϕ				1.482	0.068						1.220	0.047	1.132	1.318	1.468	0.070	1.333	1.617

Table 4.26: Comparison of the model estimates of the Poisson, NB, BP, BP-LN and BP-G regression models for admissions due to neoplasms among females.

risk profile to evaluate the fit of the models (see Figure 4.12). The risk profile was considered over different years. The plots indicate that in all the years, the models have a satisfactory fit. In all instances, the Poisson and BP models had overlapping curves and similarly for the NB and BP-G models.



(a) (b)

(c) (d)

Figure 4.12: Observed logarithmic admission rates due to neoplasms and fitted rates under different models along with CI intervals for Bayesian models for different years. Risk profile: SEX-female; UR-urban; REGION-south; EECLASS-salary non-union; EESTATU-active fill-time; EMPREL-employee; PLANTYP- PPO. (**a**) 2016; (**b**) 2017; (**c**) 2018; (**d**) 2019.

4.3 Chapter Summary

In conclusion, numerous classical and Bayesian models were developed in this chapter for modelling the admission counts related to respiratory diseases and neoplasms.

In all instances, the NB models provided a better fit to the data compared to the corresponding Poisson models. Among the Bayesian models, the BP-G model gave the best fit, followed by the BP-LN model in terms of the DIC values, with the exception of models for neoplasms-related admissions in males. In that context, the BP-LN model showed a slightly better fit than the BP-G model. The improvement or variation in the fit of the models is not very evident when the fitted curves are plotted for specific risk profiles due to the granularity. In most instances, the fitted curves for the Poisson and the BP models appear to be overlapping and likewise for the NB and BP-G models. A number of aspects, including predictive performance, also need to be taken into consideration prior to upholding a specific model over alternative models. Such an analysis is carried out at the end of the next chapter, where the predictive performance of classical, Bayesian and NN models is compared. In the next chapter, we employ deep learning methodologies for developing network-based models for modelling admission rates.

Chapter 5

Neural Network Models for Modelling Admission Rates

The focus of this chapter is the development of a suite of neural network-based predictive models that can accurately predict admission rates related to critical illnesses, such as respiratory diseases. As mentioned in Chapter 2, the extensive adoption of deep learning approaches in recent years is evident in the insurance context, as in other research areas. Additionally, in Chapter 2 we also briefly described the structure of a FFNN, which serves as the core design for all the models discussed in this chapter. In what follows, we provide an overview of the key aspects considered in this chapter, parts of which have also been discussed in Jose et al. (2022).

We start by investigating the efficiency of the Poisson and negative binomial Combined Actuarial Neural Network (CANN) models (Schelldorfer and Wuthrich (2019)) for predicting admission rates using the same admission counts data for respiratory diseases that we considered in the previous chapter while developing the Bayesian models. In particular, we begin by considering Poisson NN models, including a CANN model, and develop modifications based on early stopping and dropout techniques that improve their performances. Subsequently, motivated by the suitability of the negative binomial distribution for addressing over-dispersion, we also develop negative binomial NNs and compare their predictive performance with that of the

Poisson models. NN-based models are trained by minimising the corresponding deviance loss functions and are compared using the testing data loss. It is also worth noting that while the use of machine learning approaches and CANN models, under both Poisson and negative binomial distributional assumptions, has been explored for data-driven applications in the field of non-life insurance, to the best of our knowledge, this is the first instance which considers employing such methods for morbidity modelling in an insurance context.

Furthermore, we consider the bias-regularised version of the negative binomial NN and CANN models by modifying the intercept of their output layers following the approach of Wüthrich (2020). Additionally, we also address bias issues on a population level by extracting the last hidden layer of the NN and CANN model, fitting the corresponding negative binomial regression models and, therefore, controlling the portfolio or population level bias by adjusting the intercept.

Moreover, following the setup of Richman and Wüthrich (2020), we determine a nagging predictor in the case of the negative binomial NN and CANN models for taking advantage of the randomness of neural network calibrations to provide more stable predictions than those under a single neural network run. Moreover, for providing reliable comparisons between the performance of the regression, NN and CANN models, a k-fold validation is carried out to allow us to evaluate the models' predictive ability when a range of different data configurations is considered for training and prediction purposes. The aforementioned modelling approaches are then extended for other diseases, such as neoplasms.

5.1 Feature Pre-processing for Network Models

As previously mentioned, the development of different neural network methodologies was first carried out using the admission data for respiratory diseases. We consider the individual-level data in the year 2016, which comprises 2,050,100 records. The details of the different variables in the data set are given in Table 5.1. The EGE-

OLOC variable was chosen over the REGION variable as it gives more granular information regarding an individual's location of residence.

Variable	Description	Comment	Categories
ENROLID	Unique ID for individual	ID variable	-
AGE	Age of the last birthday of the individual	$\in[30, 65]$	-
SEX	Gender of the individual	Factor w/2 categories	1: Male, 2: Female
UR	Urban/ rural indicator based on individual's residence	Factor w/2 categories	1: Rural, 2: Urban
REGION	Geographical region of residence	Factor w/5 categories	1: Northeast, 2: North Central, 3: South, 4: West, 5: Unknown
EGEOLOC	Geographic location based on postal code of individual's residence	Factor w/53 categories	See Table A.3
EECLASS	Employee classification	Factor w/9 categories	1: Salary Non-union, 2: Salary Union, 3: Salary Other, 4: Hourly Non-union, 5: Hourly Union, 6: Hourly Other, 7: Non-union, 8: Union, 9: Unknown
EESTATU	Status of employment	Factor w/9 categories	1: Active Full Time, 2: Active Part Time or Seasonal, 3: Early Retiree, 4: Medicare Eligible Retiree, 5: Retiree (status unknown), 6: Comprehensive Omnibus Budget Reconciliation Act (COBRA) Continuee, 7: Long-Term Disability, 8: Surviving Spouse/Depend, 9: Unknown
EMPREL	Relation to the primary beneficiary	Factor w/3 categories	1: Employee, 2: Spouse, 3: Child/Other

Table 5.1: Description of variables in the admission data set.

Variable	Description	Comment	Categories
PLANTYP	Type of health plan individual is part of	Factor w/8 categories	2: Comprehensive Plan, 3: Exclusive Provider Organization Plan, 4: Health Maintenance Organization Plan, 5: Non-Capitated (Non-Cap) Point-of-Service, 6: Preferred Provider Organization Plan, 7: Capitated (Cap) or Partially Capitated (PartCap) Point-of-Service Plan, 8: Consumer-Driven Health Plan, 9: High-Deductible Health Plan
EXPOSURE	Period of enrollment- yearly exposure	$\in(0,1]$	-

Table 5.1: *Cont.*

In addition to the data considerations detailed in Chapter 3, before developing the neural network models, some further feature pre-processing was undertaken. The full data set was randomly split into a learning set $\mathcal{D}$ and a testing set $\mathcal{T}$. The learning set and testing set, thus created using a 90:10 split, comprised 1,845,090 and 205,010 records, respectively. All models were fitted using the learning data set, and the performance of the models was evaluated on the testing set under the assumption that the underlying modelling assumptions hold for both sets of data.

As discussed in Ferrario et al. (2020), the proper working of the gradient descent methods (GDMs) used for neural network model fitting requires the different feature components to be on an identical scale. In order to adjust the scaling, for the continuous and binary variables such as AGE, SEX and UR, a min-max scaler was adopted, which transforms the variable to a scale of $[-1,1]$ using the formula:

$$x_j \mapsto x_j^* = \frac{2(x_j - m_j)}{M_j - m_j} - 1 \in [-1,1] \tag{5.1}$$

where m_j and M_j represent the minimum and maximum values of variable x_j. For the binary variables SEX and UR, the values would be replaced by either -1 or 1. For

all the remaining categorical variables, since they are nominal in nature, a dummy encoding was used. Under dummy encoding, a categorical variable with l levels would be represented using a $(l-1)$ dimensional feature vector, with a reference category and value 1 being used to identify the actual level for a data record (see Equation 5.2). An alternative option is to use one-hot encoding, which would have resulted in an (l) dimensional feature vector without a reference level (Equation 5.3). For a categorical variable x_j, with l categories $c_1, c_2, .., c_l$, treating c_l as the reference level, dummy encoding is given by

$$x_j \mapsto \left(\mathbb{1}_{\{x_j=c_1\}}, ..., \mathbb{1}_{\{x_j=c_{l-1}\}}\right)^T \in \mathbb{R}^{l-1} \tag{5.2}$$

whereas one-hot encoding would take the form

$$x_j \mapsto \left(\mathbb{1}_{\{x_j=c_1\}}, ..., \mathbb{1}_{\{x_j=c_l\}}\right)^T \in \mathbb{R}^{l}. \tag{5.3}$$

As both approaches increase the dimension of the feature space, for neural network models, data embedding was implemented using an embedding layer, which facilitates a lower dimensional representation of the categorical variables. The embedding layer approach was proposed by Bengio et al. (2000) and was adopted in an insurance context by Richman and Wüthrich (2021). The embedding layer maps a categorical variable x_j with l levels to a low dimensional real-valued vector of dimension v $((v < l)$ (i.e. $f : x_j \mapsto \mathbb{R}^v)$. The value of v, which is treated as a hyper-parameter, needs to be decided, taking into consideration that it impacts the complexity of the model. In the current context, we have chosen $v = 2$ for creating the embedding layers for categorical variables (see Figure 5.1). Code for data preparation, feature pre-processing and other supporting functions for recording model performance is given in Listing A.1.

5.2 Network-based Models

Various regression models for modelling the admission counts data were discussed in Chapter 4. For the ease of the reader and to demonstrate the development of neural network models with underlying distributional assumptions similar to the regression models, the structure of the regression models is once again briefly explained here. For the Poisson GLM, we assume that for $i = 1, \ldots, n$, n being the number of records in the learning data set $\mathcal{D}$, admission numbers, A_i, follow a Poisson distribution with

$$A_i \sim Poisson(\lambda_i e_i) \tag{5.4}$$

where the mean ($\mu_i^{Pois} = \lambda_i e_i$) depends on the policyholder's characteristics $\boldsymbol{x_i}$ through $\lambda_i = \exp(\boldsymbol{\beta}^\top \boldsymbol{x_i})$, and the exposure e_i. By choosing the logarithmic link function, which is, in fact, with the canonical link function for the Poisson GLM, we have a predictor of the form

$$\mu_i^{Pois} : \mathcal{X} \mapsto \mathbb{R}_+, \qquad (o_i, \boldsymbol{x_i}) \mapsto \log(\mu_i^{Pois}) = o_i + \boldsymbol{\beta}^\top \boldsymbol{x_i} = o_i + \langle \boldsymbol{\beta}, \boldsymbol{x_i} \rangle, \tag{5.5}$$

where $\mathcal{X} \subset \mathbb{R}^q$ is the feature space with $\boldsymbol{x}_i = (1, x_{i,1}, \ldots, x_{i,q})^\top$ giving the feature information. $o_i = \log(e_i)$ is the offset term, and $\boldsymbol{\beta}^\top = (\beta_0, \beta_1, .., \beta_q)$ is the unknown vector of coefficients to be estimated.

Similarly, for the negative binomial regression model, admission numbers, A_i, $i = 1, \ldots, n$, are assumed to follow a negative binomial distribution with a dispersion parameter $\phi > 0$:

$$A_i \sim NB(\mu_i^{NB}, \phi) \tag{5.6}$$

with

$$\mathbb{E}[Y_i] = \mu_i^{NB} \qquad \text{and} \qquad \mathbb{V}(Y_i) = \mu_i^{NB} + \phi \mu_i^{NB^2}. \tag{5.7}$$

For a logarithmic link function, the predictor μ_i^{NB} has the form

$$\mu_i^{NB} : \mathcal{X} \mapsto \mathbb{R}_+ \qquad \mu_i^{NB} = \exp(o_i + \langle \boldsymbol{\beta}, \boldsymbol{x_i} \rangle). \tag{5.8}$$

5.2.1 Neural network model

In what follows, we consider the basic feed-forward neural network (FFNN) model. In particular, a FFNN constitutes an input layer, one or more hidden layers, and an output layer. The feature space $\mathcal{X}$ acts as the input layer and the hidden layers are composed of a set number of neurons in each layer. The output from a particular hidden layer serves as the input for the next layer. The output of a neuron depends on the linear combination of the previous layer's output and the choice of activation function assigned to the layer that it is part of (see Section 5.3 for more details on activation function). The number of hidden layers $d \in \mathbb{N}$ is treated as a hyper-parameter and is also referred to as the depth of the network. The last layer of the architecture is the output layer, which is connected to the last hidden layer. In a neural network, each layer is a function of the previous layer. The m^{th} hidden layer $\mathbf{z}^{(m)}, 1 \leq m \leq d$ with dimension $q_m \in \mathbb{N}$ can be defined as

$$\mathbf{z}^{(m)} : \mathbb{R}^{q_{m-1}} \to \mathbb{R}^{q_m}, \quad \mathbf{z} \mapsto \mathbf{z}^{(m)}(\mathbf{z}) = (1, z_1^{(m)}(\mathbf{z}), \ldots, z_{q_m}^{(m)}(\mathbf{z}))^\top, \tag{5.9}$$

inclusive of the intercept component, where the neurons $z_j^{(m)}, 1 \leq j \leq q_m$, are given by

$$z_j^{(m)}(\mathbf{z}) = \psi\left(\langle \boldsymbol{\beta}_j^{(m)}, \mathbf{z} \rangle\right), \tag{5.10}$$

with $\psi : \mathbb{R} \to \mathbb{R}$, being the activation function and $\boldsymbol{\beta}_j^{(m)} \in \mathbb{R}^{q_{m-1}+1}$ the network parameters. The hidden layer, $\mathbf{z}^{(m)}$ depends on network parameters $\left(\boldsymbol{\beta}_1^{(m)}, \ldots, \boldsymbol{\beta}_{q_m}^{(m)}\right) \in \mathbb{R}^{q_m}$. For $q_0 = q$, with q being the dimension of the feature space $\mathcal{X}$, the network parameter $\boldsymbol{\beta} = \left(\boldsymbol{\beta}_1^{(1)}, \ldots, \boldsymbol{\beta}_{q_d}^{(d)}, \boldsymbol{\beta}^{(d+1)}\right) \in \mathbb{R}^r$ will have dimension r where

$$r = \sum_{m=1}^{d} q_m \left(q_{m-1} + 1\right) + (q_d + 1). \tag{5.11}$$

A diagrammatic representation of a feed-forward neural network with an embedding layer and three hidden layers with 20,15,10 neurons in each layer, are shown in Figure 5.1.

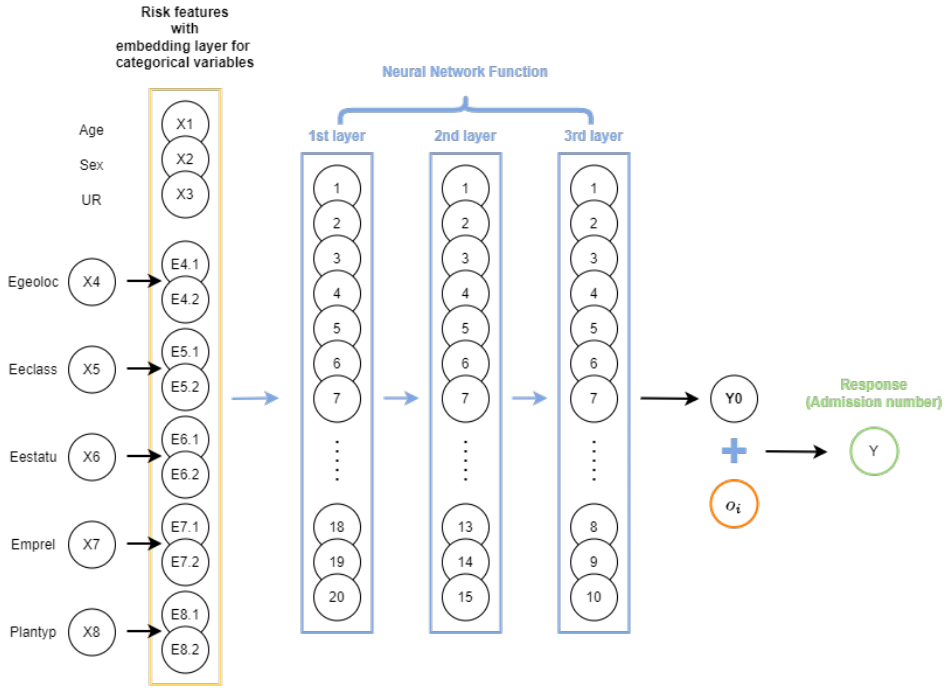


Figure 5.1: A sample representation of a feed-forward neural network with embedding layers and three hidden layers with 20,15,10 neurons in each layer.

Under a neural network regression model, the predictors μ^{Pois} and μ^{NB} of the traditional Poisson and negative binomial regression models are replaced by the neural network predictors μ^{PoisNN} and μ^{NBNN}. To illustrate the structure of the network-based model, we will refer to the models under the Poisson distributional assumption. For an FFNN model with depth d under the Poisson assumption, the predictor is of the form

$$(o_i, \mathbf{x}_i) \mapsto \log(\mu_i^{PoisNN}) = o_i + \langle \boldsymbol{\beta}^{(d+1)}, (\mathbf{z}^{(d)} \circ \cdots \circ \mathbf{z}^{(1)})(\mathbf{x}_i) \rangle, \tag{5.12}$$

for $i = 1, \ldots, n$, where $\boldsymbol{\beta}^{(d+1)} \in \mathbb{R}^{q_d+1}$ are the weights that map the neurons of the last hidden layer $\mathbf{z}^d$ to the output layer $\mathbb{R}_+$. The code for implementing a sample Poisson NN model is given in Listing A.2 and the pseudo code explaining the steps involved are given in Algorithm A.1.

5.2.2 CANN model

As mentioned earlier, the CANN model has an additional regression function nested into the model predictor using a skip connection. Additionally, it is worth noting that the CANN model and its extensions are part of the Residual Neural Network (ResNet) family, which employs skip connections, as discussed in He et al. (2016). These skip connections enable the model to address the vanishing gradient effects commonly associated with deep NNs. Under the Poisson distributional assumption, the model predictor of the CANN model with depth $d \in \mathbb{N}$ has the form

$$(o_i, \mathbf{x}_i) \mapsto \log(\mu_i^{PoisCANN}) = o_i + \langle \boldsymbol{\beta}^{Pois}, \mathbf{x}_i \rangle + \langle \boldsymbol{\beta}^{(d+1)}, (\mathbf{z}^{(d)} \circ \cdots \circ \mathbf{z}^{(1)})(\mathbf{x}_i) \rangle, \quad (5.13)$$

with $i = 1, \ldots, n$ and parameter vector $\boldsymbol{\beta} = \left(\boldsymbol{\beta}^{Pois}, \boldsymbol{\beta}^{(d+1)}\right)^\top \in \mathbb{R}^{q_0+r}$. The $\boldsymbol{\beta}^{Pois}$ vector represents the parameters associated with the skip connection. The three terms on the right-hand side of Equation 5.13 represent the offset, regression function/skip connection, and the network function, respectively. A schematic representation of a sample CANN model with three hidden layers and 20,15,10 neurons in each layer is shown in Figure 5.2.

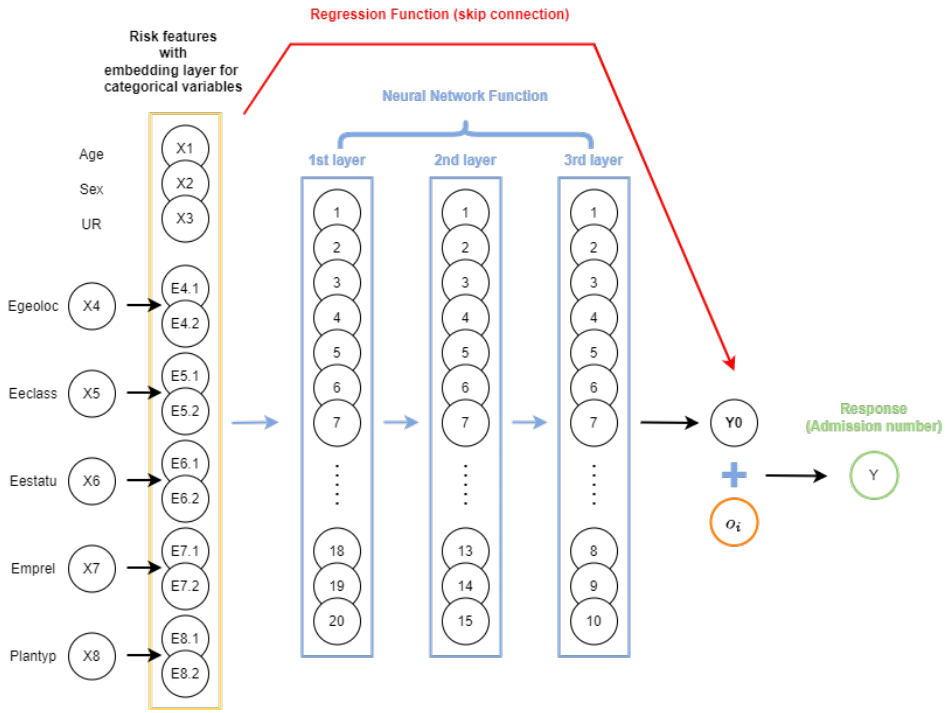


Figure 5.2: An illustration of a sample CANN model with three hidden layers and 20,15,10 neurons in each layer.

As shown in Figure 5.2, feature space $\mathcal{X}$ is directly linked to the output layer. Different variants of CANN exist depending on whether the weights in the regression part are updated or not whilst training the model (Schelldorfer and Wuthrich (2019)). Owing to the ease of implementation and following the literature, we focus on the variant in which the weights of the regression component are kept fixed as the iterated weighted least squares (IWLS) estimate $\hat{\boldsymbol{\beta}}^{Pois}$ from the regression model. This particular variant of the CANN can be implemented by replacing the offset term o_i in Equation 5.13 with

$$o_i^{CANN} = \log\left(e_i \exp\langle \hat{\boldsymbol{\beta}}^{Pois}, \boldsymbol{x_i}\rangle\right). \tag{5.14}$$

The approach can be enacted indifferently under both Poisson and negative binomial distribution assumptions by adjusting the likelihood function under each model and altering the sample code given in Listing A.3. For the models under the Poisson distribution assumption, with y_i being the response variable and μ_i the mean (μ_i^{Pois} for regression, μ_i^{PoisNN} for FFNN and $\mu_i^{PoisCANN}$ for CANN), the log-likelihood is

of the form

$$l(\mu; y) = \sum_{i=1}^{n} \{y_i \ln(\mu_i) - \mu_i - \ln \Gamma(y_i + 1)\}. \tag{5.15}$$

Likewise, for the negative binomial models, under the `NB2` parameterisation (Hardin and Hilbe (2007)), the log-likelihood is given by

$$l(\mu; y, \phi) = \sum_{i=1}^{n} \Big\{ y_i \ln\Big(\frac{\phi\mu_i}{1+\phi\mu_i}\Big) - \frac{1}{\phi}\ln(1+\phi\mu_i) + \ln\Gamma\Big(y_i + \frac{1}{\phi}\Big) - \ln\Gamma\Big(y_i + 1\Big) - \ln\Gamma\Big(\frac{1}{\phi}\Big)\Big\}. \tag{5.16}$$

The details of the fitting of the different models discussed so far are discussed in detail in the next section.

5.3 Model Fitting

The different data considerations and exploratory analysis described in the Chapter 3, as well as the development of models described in the previous section, were carried out using the programming language `R` using `RStudio IDE` (R Core Team (2021) and RStudio Team (2021)). The two main packages utilised for developing the neural-network-based models are `keras` and `TensorFlow` packages, details of which can be found in the respective manuals Allaire and Chollet (2021) and Allaire and Tang (2021). The important snippets of code developed for implementing NN and CANN models, as well as the different model improvement approaches, are given in the Appendix.

5.3.1 Hyper-parameters

The development of the network-based models was carried out in different stages. The first step of constructing the models involves determining the hyper-parameters, such as the number of hidden layers, choice of activation function, and the gradient descent method (GDM) used for model training. In terms of the aspects mentioned above, following earlier work by Ferrario et al. (2020) in a similar context, the

following assumptions were made:

- Number of hidden layers: the number of hidden layers was kept at three.
- Activation function: for hidden layers, the hyperbolic tangent function, $\psi(x) = \tanh(x)$, was used. Any alternative non-linear activation function would work. The motivation behind a non-linear activation function is that a non-linear activation function allows for a non-linear model space, reducing the number of nodes needed and allowing the network to automatically capture the interaction effect of different features. For the output layer, an exponential function was used, which is the inverse of the link function ($g(.) = \ln()$) and, therefore, is in line with the underlying distributional assumption.
- Gradient descent method: the neural network training utilises a gradient descent optimisation algorithm for estimating the model weights. Ferrario et al. (2020) compared different GDMs in terms of performance and identified the Nesterov-accelerated adaptive moment estimation (Nadam) method as performing better compared to other similar methods. Hence, we also adapted the Nadam as the choice of GDM. An overview of the different GDMs is given in Ruder (2016), and additional details regarding the 'Nadam' method could be found in Dozat (2016).
- Validation set: the training of neural network models requires further splitting the learning set into a training set, $\mathcal{D}^{(-)}$, and a validation data set, $\mathcal{V}$. The validation data set is used as the evaluation set during the iterative process for estimating the model weights. In other words, $\mathcal{V}$ tracks possible over-fitting of the model to $\mathcal{D}^{(-)}$. For the network-based models discussed here, and an 80:20 split was used for the training and validation data sets. Once the training is complete, the final performance of the fitted model is assessed using the testing set.
- Loss function: the loss function is the objective function that the GDM algorithm minimises in order to estimate the model weights (Goodfellow et al.

(2016)). Numerous options exist in terms of the choice of the loss function. For instance, mean squared error (MSE), mean absolute error (MAE), and deviance loss are some of the popular choices of loss functions used in a regression problem. For our context, we adapted deviance loss as the loss function. The motivation behind the particular choice is that minimising the deviance loss is equivalent to maximising the corresponding log-likelihood function, which gives the MLE. The deviance loss is defined as the difference between the log-likelihood of the saturated or full model and the fitted model, and for a data set $\mathcal{A}$ the Poisson deviance loss is given by

$$\mathcal{L}_{\mathcal{A}}(\boldsymbol{\beta}) = 2\sum_{i\in\mathcal{A}}\Big(y_i \log y_i - y_i - y_i \log \hat{\mu}_i + \hat{\mu}_i\Big), \tag{5.17}$$

where $\hat{\mu}_i$ denotes the fitted mean and y_i the observed number of admissions for $i = 1, \ldots, n$. Similarly, for models under the negative binomial distributional assumption, the deviance loss function has the form:

$$\begin{aligned}\mathcal{L}_{\mathcal{A}}(\boldsymbol{\beta}, \phi) = 2\sum_{i\in\mathcal{A}}\Big(y_i \log y_i &- (y_i + \phi)\log(y_i + \phi) \\ &- y \log \hat{\mu}_i + (y_i + \phi)\log(\hat{\mu}_i + \phi)\Big).\end{aligned} \tag{5.18}$$

The GDM estimates the model weights by minimising the deviance loss $\mathcal{L}_{\mathcal{D}}$ for the learning set, and the performance of the thus fitted models can be compared using the deviance loss $\mathcal{L}_{\mathcal{T}}$ for the testing set. As described earlier, the GDM updates model weights with an improved choice in an iterative manner. The iterative updation of model weights under the GDM could be represented as

$$\boldsymbol{\beta}_{[t]} \mapsto \boldsymbol{\beta}_{[t+1]} = \boldsymbol{\beta}_{[t]} - \rho_{t+1}\nabla_{\boldsymbol{\beta}}\mathcal{L}(\mathcal{D}_s^{(-)}, \boldsymbol{\beta}_{[t]}), \tag{5.19}$$

where $[t]$ indicates the algorithmic step, and $\rho_{t+1} > 0$ gives the learning rate and $\mathcal{D}_s^{(-)}$ are the mini-batches or batches (see next section for more details on batches). The learning rate determines the size of each step and influences the speed of movement

towards the optimal model weights. Since the primary focus of this thesis is the predictive modelling of admission rates, more details relating to the functionalities of the neural network model training are not discussed further (see, e.g., Russell (2010); Goodfellow et al. (2016)).

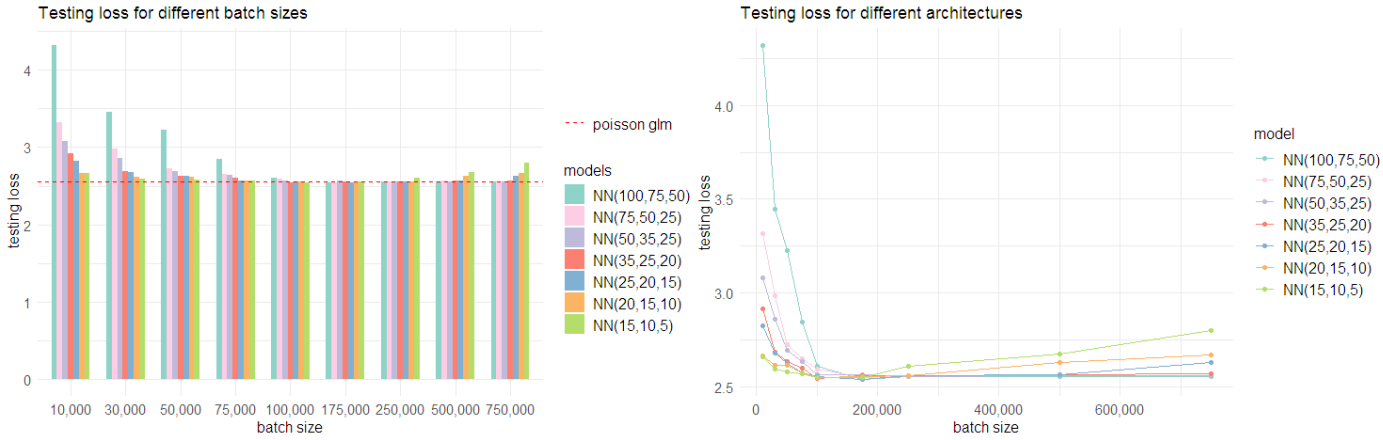
5.3.2 Batch size and epochs

In addition to the model choices discussed above, the other main model attributes that need to be determined are the number of neurons in each layer, batch size, and epochs. Due to the computational burden of considering a large data set at once, during the training of a neural network, the data in the training set $\mathcal{D}^{(-)}$ are considered in smaller batches $(\mathcal{D}_1^{(-)}, \ldots, \mathcal{D}_S^{(-)})$ created randomly, having approximately the same size $b \in \mathbb{N}$. The batch size b refers to the size of the smaller batches created. Epochs give the number of times that the full learning data set is iteratively considered during training (Ferrario et al. (2020)). The ideal choice of batch size must be determined in conjunction with the epochs as it determines the total number of GDM steps undertaken during model training, which impacts the model performance. Due to the size of the data in hand, we determined the batch size and epochs using trial and error. Two approaches, outlined below, were undertaken to determine a reasonable choice of batch size, epochs, and network architecture. The two approaches were considered using a Poisson NN with seven different model architectures of varying complexity. For all seven architectures, three hidden layers were used (100,75,50), with (75,50,25), (50,35,25), (35,25,20), (25,20,15), (20,15,10), and (15,10,5) neurons in each of the three hidden layers. The choice of the batch size epoch and network architecture thus identified from the exercise was then adopted for Poisson CANN and the network-based models under negative binomial distribution assumption.

Approach 1: batch size is varied, keeping the epochs fixed; then, for the batch size giving the best performance, different epochs are considered.

- Step 1: Initially, for different model architectures of varying complexities,

different batch sizes were considered, keeping the number of epochs fixed to 1000. All considered models involve three layers, with a different number of nodes in each layer. The different model architectures were fitted using batch sizes of 10,000, 30,000, 50,000, 75,000, 100,000, 175,000, 250,000, 500,000, and 750,000. The performances of the models were compared using the testing (out-of-sample) loss $\mathcal{L}_T$, i.e., the deviance loss (Equation 5.17) under the testing data set. The results are shown in Table A.13 and illustrated in Figure 5.3. The tables also show the learning (in-sample) deviance loss, $\mathcal{L}_D$, and the average fitted mean $\hat{\mu}$ for the full data set under the considered models.



(**a**) (**b**)

Figure 5.3: Performance of the different models under the initial step of approach 1, under varying batch sizes: (**a**) testing loss for the Poisson NN models under different architectures. The red line shows the testing loss for the Poisson regression model; (**b**) change in the testing loss for the Poisson NN models under different architectures as the batch size increases.

All models, irrespective of their level of complexity, performed well with a batch size of 175,000. As anticipated, complex models had a higher testing loss with smaller batch sizes due to over-fitting. In general, the testing loss presented a decreasing trend for all considered models as the batch size increased from 10,000 to 175,000. For batch sizes greater than 175,000, both testing loss and learning loss for simpler models started to rise. This indicates under-fitting and shows that for batch sizes greater than 175,000, the complexity of simpler models with fewer neurons in the hidden layers is insufficient to

fit the data effectively. Hence, this analysis suggested choosing a batch size of 175,000. Nevertheless, as all models had a comparable testing loss for batch size 175,000, three of the simpler models (NN (25,20,15), NN (20,15,10), and NN (15,10,5)) were considered further for identifying the optimal number of epochs.

- Step 2: in order to find the optimal number of epochs, the NN (25,20,15), NN (20,15,10), and NN (15,10,5) models were fit using different choices of epochs (100, 250, 500, 1000, 1500, and 2000), keeping the batch size fixed at 175,000. The results of these models are given in Table 5.2.

Model	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
NN (25,20,15)	100	2.8021	2.6751	0.0033
NN (25,20,15)	250	2.7207	2.5843	0.0029
NN (25,20,15)	500	2.6910	2.5668	0.0028
NN (25,20,15)	1000	2.6713	2.5621	0.0028
NN (25,20,15)	1500	2.6509	2.5480	0.0029
NN (25,20,15)	2000	2.6212	2.5770	0.0028
NN (20,15,10)	100	2.8200	2.6968	0.0038
NN (20,15,10)	250	2.7433	2.6092	0.0031
NN (20,15,10)	500	2.6983	2.5707	0.0029
NN (20,15,10)	1000	2.6719	2.5634	0.0027
NN (20,15,10)	1500	2.6505	2.5582	0.0028
NN (20,15,10)	2000	2.6420	2.5872	0.0028
NN (15,10,5)	100	3.3765	3.2795	0.0097
NN (15,10,5)	250	2.8166	2.6936	0.0035
NN (15,10,5)	500	2.7434	2.6171	0.0033
NN (15,10,5)	1000	2.6704	2.5641	0.0028
NN (15,10,5)	1500	2.6517	**2.5407**	0.0027
NN (15,10,5)	2000	2.6457	2.5723	0.0028

Table 5.2: The testing loss, learning loss and average fitted mean of the Poisson neural network models with (25,20,15), (20,15,10), and (15,10,5) architectures for different choices of epochs and batch size of 175,000.

For the models considered in step 2, the testing loss was lowest when the number of epochs was 1500. Hence, the combination of the batch size equal to 175,000 and 1500 epochs was deemed optimal under this approach.

Approach 2: epochs number is varied, keeping the batch size fixed; then, for the epochs number giving the best performance, different batch sizes are considered. The same steps as those followed in Approach 1 were carried out under this approach as well but in an alternative order. Model architectures, batch sizes, and numbers of epochs are also the same as in Approach 1.

- Step 1: we now initially alter the number of epochs, keeping the batch size fixed at 30,000. The results of this step are given in Table A.14. For all considered model architectures, except for NN (100,75,50), the testing loss was lower than that for the Poisson regression model, when the number of epochs was 250. Moreover, the testing loss was lowest for all considered model architectures when the number of epochs was 250 (see Figure 5.4). Hence, the number of epochs was chosen as 250.

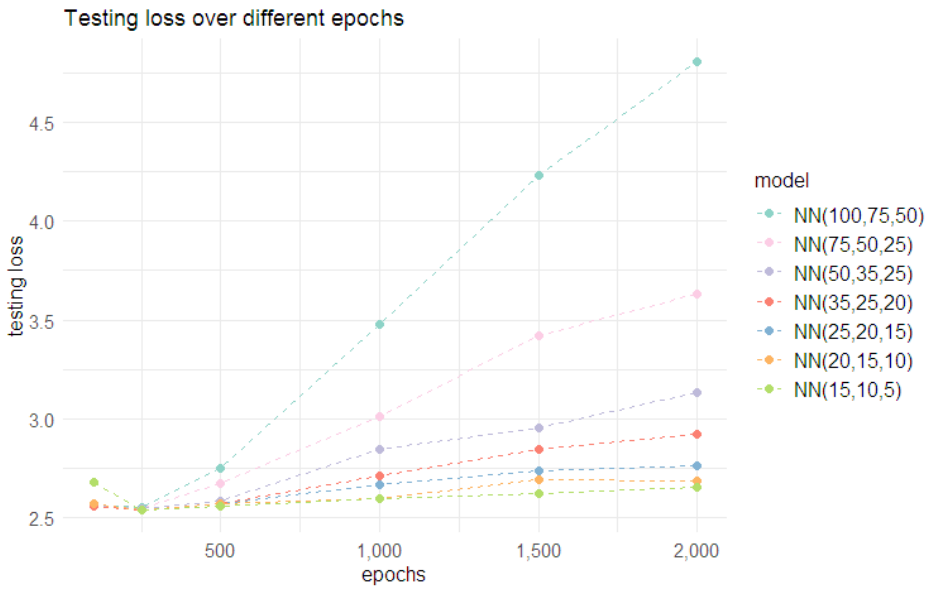


Figure 5.4: Change in the testing (out-of-sample) loss for Poisson neural network models with different architectures as epochs increase.

- Step 2: For all model architectures other than (100,75,50), different batch sizes were considered with the number of epochs fixed at 250. The results are given in Table A.15 and illustrated in Figure 5.5. With a batch size of 30,000, all models, except for NN (15,10,5), had testing losses lower than that for the Poisson regression model. When the batch size increased to 50,000, the NN

(50,35,25) also had a similar testing loss. However, a batch size of 30,000 was chosen, as all models performed well under this choice. The combination of a batch size of 30,000 and 250 epochs was deemed optimal under the second approach.

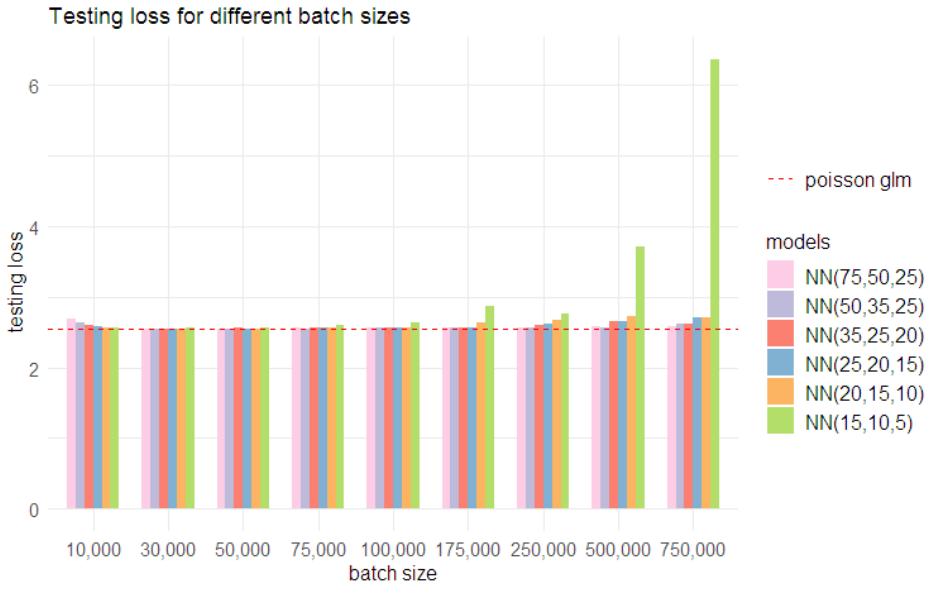


Figure 5.5: Change in the deviance testing (out-of-sample) loss for the Poisson neural network models with different architectures as batch size increases.

5.3.2.1 Comparison of approaches

The two approaches yielded different combinations of batch sizes and epochs. Under Approach 1, the best batch size and epoch number combination was (175,000, 1500), whereas Approach 2 identified (30,000, 250) as the best combination. In order to compare these combinations, results from 50 separate calibrations with different starting points for GDM were considered for the NN (30,25,20), NN (25,20,15), NN (20,15,10), and NN (15,10,5) models (see Figure 5.6). The motivation behind considering different calibrations is the inherent randomness in the results of neural network models. Several aspects of the neural network model fitting contribute to this randomness and are discussed in detail in Section 5.5.2. Different calibrations were implemented by altering the seed value for the random number generator, which determines the initial value of model weights under GDM (see Section 5.5.2 for more details).

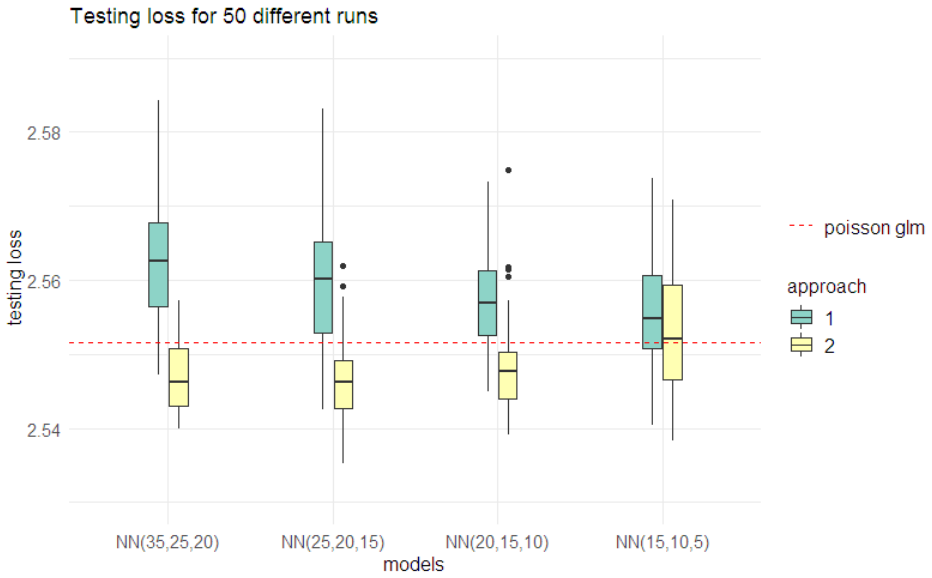


Figure 5.6: Performance from 50 different calibrations for Poisson NN models under different architectures on testing data set, for the best combination of batch sizes and epochs identified in Approach 1 (175,000, 1500) and Approach 2 (30,000, 250). The horizontal red line shows the deviance loss value for the Poisson regression model.

The graphs clearly show that the combination of a batch size of 30,000 and 250 epochs performs better than a batch size of 175,000 and 1500 epochs, in terms of testing loss. The results also indicate that both (25,20,10) and (20,15,10) architectures have similar predictive performances. Hence, we consider both these architectures while investigating additional model improvements discussed in the following section.

5.4 Model Improvements

5.4.1 Approaches for preventing over-fitting

One of the most significant aspects that need to be addressed while training a neural-network model is the over-fitting of the model to the learning set, $\mathcal{D}$, which may potentially affect the predictive performance of the model. Three of the most commonly used approaches for preventing over-fitting were considered here, i.e., regularisation, early stopping, and dropout. For comparing the different improvement

approaches, Poisson NN models with (25,20,10) and (20,15,10) architectures were used.

5.4.1.1 Regularisation

Under this approach, a penalty function is considered for the loss function, controlled by a regularisation parameter that is added to the network parameters. Following Ferrario et al. (2020), the modified loss function for the Poisson NN is given by

$$\mathcal{L}_{\mathcal{A}}(\boldsymbol{\beta};\eta) = 2\sum_{i\in\mathcal{A}}\Big(y_i \log y_i - y_i - y_i \log \hat{\mu}_i + \hat{\mu}_i\Big) + \eta||\boldsymbol{\theta}_-||_p^p \tag{5.20}$$

where η is the regularisation parameter and $\boldsymbol{\theta}_-$ is the subset of network parameters considered for regularisation; $||\boldsymbol{\theta}_-||_p$ is the ℓ^p-norm and gives $\big(\sum_{j\in\boldsymbol{\theta}_-}|\theta_j|^p\big)^{1/p}$. Ridge regularisation ($p = 2$) was selected over LASSO (least absolute shrinkage and selection operator regression) regularisation ($p = 1$), as the former penalises the parameters depending upon their values and not on the same scale (Ferrario et al. (2020)). Following the literature, Hastie et al. (2009) and Ferrario et al. (2020), regularisation was applied to all network parameters except for the intercepts and the last output layer. The main criticism that regularisation faces is that it is heavily influenced by the choice of the regularisation parameter. For each of the (25,20,10) and (20,15,10) architectures, four choices of the regularisation parameter η were considered ($\eta = 10^{-1}$, $\eta = 10^{-3}$, $\eta = 10^{-5}$ and $\eta = 10^{-8}$) and the results are shown in Table 5.3 (see Listing A.4 for the sample code).

Model	η value	Learning Loss	Testing Loss	Average Fitted Mean
NN (25,20,15)	$\eta = 0$	2.6614	**2.5354**	0.0027
NN (25,20,15)	$\eta = 10^{-1}$	2.8052	2.6787	0.0029
NN (25,20,15)	$\eta = 10^{-3}$	2.8045	2.6767	0.0027
NN (25,20,15)	$\eta = 10^{-5}$	2.6827	2.5536	0.0028
NN (25,20,15)	$\eta = 10^{-8}$	2.6614	**2.5354**	0.0027
NN (20,15,10)	$\eta = 0$	2.6584	2.5380	0.0027
NN (20,15,10)	$\eta = 10^{-1}$	2.8049	2.6781	0.0028
NN (20,15,10)	$\eta = 10^{-3}$	2.8045	2.6767	0.0027
NN (20,15,10)	$\eta = 10^{-5}$	2.6817	2.5550	0.0028
NN (20,15,10)	$\eta = 10^{-8}$	2.6585	2.5380	0.0027

Table 5.3: Testing loss, learning loss and average fitted mean of the Poisson neural network models ((20,15,10), and (25,20,15) architectures) with regularisation ($\eta = 10^{-1}, \eta = 10^{-3}, \eta = 10^{-5}, \eta = 10^{-8}$), and without regularisation ($\eta = 0$).

From the results, it is evident that for both model architectures, the testing loss decreased (improvement in predictive performance) when the η value decreased from 10^{-1} to 10^{-8}. Nevertheless, for $\eta = 10^{-8}$, both architectures had the same testing loss as that of the model without regularisation, indicating non-improvement of the predictive performance under regularisation. In order to further assess the effectiveness of regularisation while accounting for the inherent randomness in the results, multiple calibrations of both architectures were considered with $\eta = 10^{-8}$ (see Section 5.4.1.4).

5.4.1.2 Early stopping

Generally, a neural network model starts to over-fit after a particular number of epochs. This is related to work in Section 5.3.2 on identifying the ideal number of epochs for a given batch size. The logic behind the early-stopping approach is to identify the ideal number of epochs above which the model starts to over-fit. In other words, the aim is to identify the number of epochs beyond which the validation loss starts to increase since an increase in validation loss indicates over-fitting of the model. Numerous ways of implementing early stopping exist, the details of which are discussed by Prechelt (1998). Our implementation employs a callback approach, which initially lets the model train for a large epoch. When the training is over, the

model weights that gave the lowest value for the loss function on the validation set, are retrieved.

The early-stopping approach was implemented for both (25,20,15) and (20,15,10) architectures with a batch size of 30,000 and 1000 epochs. As anticipated, the models with early stopping performed better (see Table 5.4). For both model architectures, the ideal number of epochs that gave the best validation loss was around 250 (231 for (25,20,15) and 272 for (20,15,10)). Both models gave similar results as that of the already identified choice of 30,000 batch size and 250 epochs. The results indicate that the early-stopping approach is desirable over the version without early stopping (see Listing A.5 for sample code). In the current context, since the early stopping was implemented using the callback approach, this overcomes the hurdle of identifying the optimal number of epochs and allows to have an arbitrarily large number of epochs (e.g., 500 epochs).

Model	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
NN (25,20,15)	early stopped	2.6654	2.5456	0.0025
NN (25,20,15)	1000 epochs	2.5635	2.6704	0.0028
NN (20,15,10)	early stopped	2.6623	**2.5403**	0.0025
NN (20,15,10)	1000 epochs	2.5865	2.6216	0.0027

Table 5.4: The testing loss, learning loss and average fitted mean of the Poisson neural network models ((20,15,10), and (25,20,15) architectures) with and without early stopping.

5.4.1.3 Dropout

Under this approach, for each step of the gradient descent (50 GDM steps within each of the 250 epochs), each neuron is dropped with a probability p, independently of other neurons creating a thinned network. The model weights are shared among the different thinned networks considered and make up the final unthinned neural net. In other words, for a GDM step involving thinned network, the gradient of the weights of the dropped neurons is zero Srivastava et al. (2014). See Figure 5.7 for a sample representation of the dropout process for a NN (20,15,10) architecture.

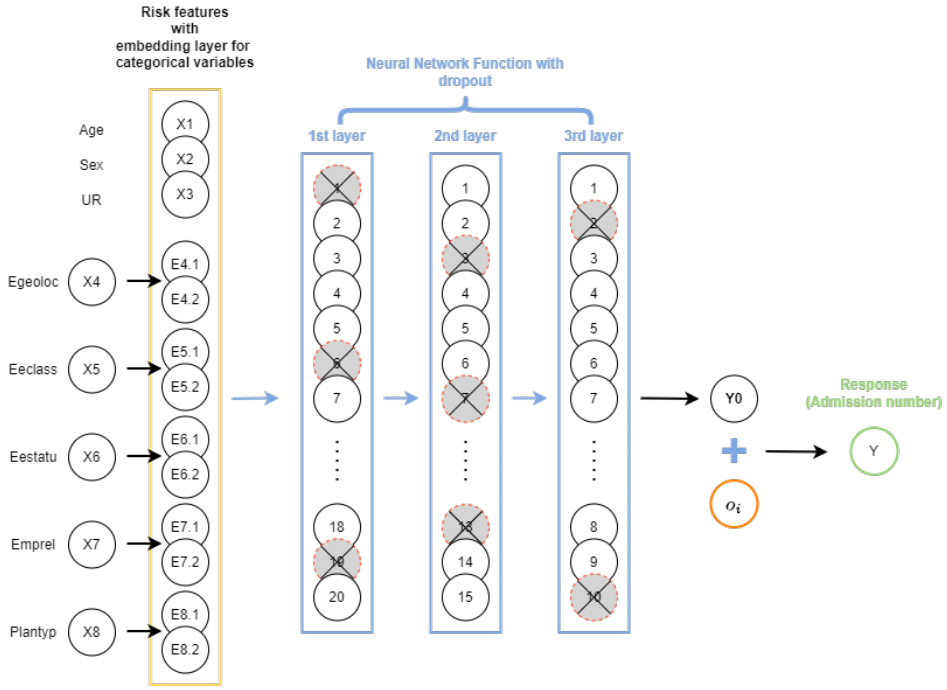


Figure 5.7: An illustration of dropout process for a sample NN model with three hidden layers and 20,15,10 neurons in each layer. The gray crossed-out circles in each layer represent the neurons that were randomly dropped with probability p.

A fixed dropout rate was used for neurons in all three layers. Dropout rates of $1\%, 2\%, 5\%$, and 10% were considered for both model architectures. The results are shown in Table 5.5. When the dropout rate was 2%, the testing loss for the (25,20,15) architecture was similar to that of the implementation without dropout, whereas for the (20,15,10) architecture, the testing loss decreased (see Listing A.6 for sample code). The average fitted mean for all the models with dropout is slightly different from the data indicating bias at the portfolio or population level (see Section 5.5.1 for more details on the population level bias of NN models and approaches for addressing it). Since the dropout rate of 2% performed better in comparison to $1\%, 5\%$, and 10%, the former was adopted for further comparison where the randomness in the model results is also taken into consideration (see Section 5.4.1.4).

Model	Dropout Rate	Learning Loss	Testing Loss	Average Fitted Mean
Data				0.0027
NN (25,20,15)	no dropout	2.6581	2.5416	0.0027
NN (25,20,15)	p=1%	2.6563	2.5437	0.0025
NN (25,20,15)	p=2%	2.6571	2.5418	0.0025
NN (25,20,15)	p=5%	2.6622	2.5484	0.0023
NN (25,20,15)	p=10%	2.6708	2.5441	0.0022
NN (20,15,10)	no dropout	2.6625	2.5458	0.0026
NN (20,15,10)	p=1%	2.6594	2.5478	0.0025
NN (20,15,10)	p=2%	2.6606	**2.5411**	0.0024
NN (20,15,10)	p=5%	2.6726	2.5554	0.0022
NN (20,15,10)	p=10%	2.6756	2.5566	0.0021

Table 5.5: Testing loss, learning loss and average fitted mean of the Poisson neural network models ((20,15,10) and (25,20,15) architecture) with dropout rates of $0\%, 1\%, 2\%, 5\%$ and, 10%.

5.4.1.4 Comparison of model improvement approaches for avoiding over-Fitting

The model improvement approaches discussed above are subject to the inherent randomness in neural network training. Hence, for comparing the different approaches, 50 different calibrations were considered under each approach for both (25,20,15) and (20,15,10) architectures. The results from these different runs are illustrated in Figure 5.8.

Comparisons among these approaches indicated that the early stopping and dropout approaches significantly improved predictive performance. Since both approaches can be applied on the same run, it was decided to adopt both simultaneously as well. For the (25,20,15) architecture, the combination of dropout and early-stopping approaches had a similar performance as that of the two approaches when applied individually. In contrast, the (20,15,10) architecture showed improvement in the predictive performance when both approaches were applied simultaneously (see Figure 5.8). Hence, it was decided to proceed by adopting the combined improvement approach (early stopping and dropout).

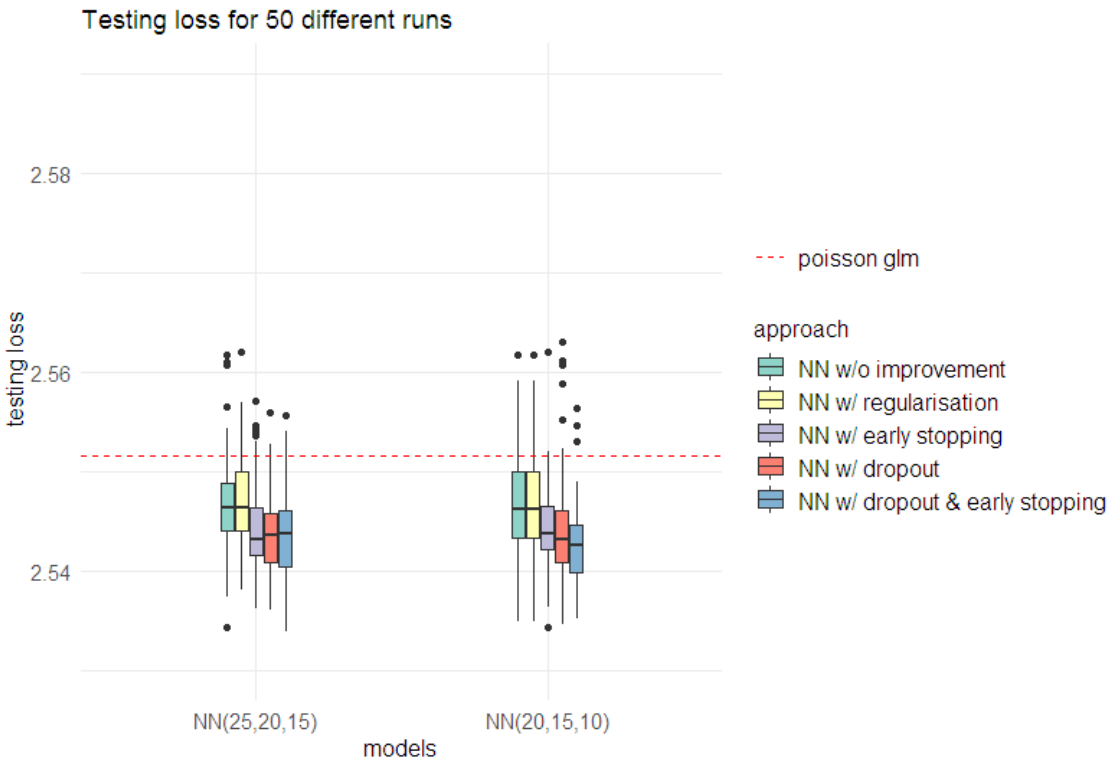


Figure 5.8: Performance from 50 different calibrations of the Poisson NN model under (25,20,15) and (20,15,10) architectures on testing and learning data sets for different model improvement approaches.

5.5 Negative Binomial Neural Network Models

A comparison was carried out between the different models under Poisson and negative binomial distribution assumptions. The predictive performance of the models was evaluated based on their performance on the testing set. The Poisson distribution is nested in the negative binomial distribution (Tzougas et al. (2014), Cameron and Trivedi (2013)), but the deviance values are on different scales. Hence, throughout this thesis, we utilise negative log-likelihood (NLL) to compare models under different distributional assumptions. For the network-based models, a single calibration with the combined improvement approach (early stopping and dropout) was considered to directly compare with the regression models (see Table 5.6). For the network-based models, (25,20,15) and (20,15,10) architectures were used with a batch size of 30,000 and 500 epochs for training the model.

Model	Learning Loss	Testing Loss	Average Fitted Mean
Data			0.0027
Pois. reg	28450.5	2995.4	0.0027
NN_{Pois} (20,15,10)	28349.5	2972.1	0.0028
CANN_{Pois} (20,15,10)	28333.0	2967.7	0.0027
NN_{Pois} (25,20,15)	28312.5	2973.2	0.0026
CANN_{Pois} (25,20,15)	28309.0	2974.2	0.0028
NB.reg	26441.9	2872.1	0.0028
NN_{NB} (20,15,10)	26326.4	2845.7	0.0028
CANN_{NB} (20,15,10)	26394.1	2839.0	0.0028
NN_{NB} (25,20,15)	26362.1	2850.1	0.0029
CANN_{NB} (25,20,15)	26426.1	**2838.8**	0.0029

Table 5.6: Testing loss and learning loss in terms of NLL and average fitted mean of regression and network-based models under the Poisson and NB distributional assumptions.

The models under the negative binomial distribution assumption show better predictive performance than those under the Poisson assumption. The main difference between the models under Poisson and negative binomial assumption is the additional dispersion parameter ϕ in the negative binomial distribution. The dispersion parameter is not trained as part of the neural network model fitting and is considered separately. For training the network-based models, the dispersion parameter determined from the negative binomial regression model is used as the initial value, and the parameter is adjusted once separately after the first round of model training. The adjustment factor used to update the dispersion parameter is the ratio between the testing loss of the best neural network model and the regression model. Once the dispersion parameter is modified by multiplying with the adjustment factor, the model is freshly trained using the new value of the dispersion parameter.

From the results shown in Table 5.6, it is evident that the models under the negative binomial distribution assumption have lower testing loss indicating a much better predictive performance compared to models under the Poisson assumption. It is true for NN and CANN models of 25,20,15 and 20,15,10 architectures. The results also show that the CANN model performed better than the NN model under the negative binomial assumption for both 25,20,15, and 20,15,10 architectures. The

same is true for 20,15,10 architecture under the Poisson assumption. For the 25,20,15 architecture, the Poisson NN model had a slightly lower testing loss than the Poisson CANN model. The average fitted mean from the different network-based models was slightly different compared to the actual data. This points toward the network-based model's failure to balance property at the population level and is discussed in the section below.

5.5.1 Bias regularisation

One main criticism faced by neural network models is that the balance property fails to hold on a population level: although the model gives accurate results on granular individual-level data, unbiasedness (or the balance property) fails on a population level. In actuarial applications, this presents an important concern as the model can potentially lead to substantial mispricing at a population level. The root cause is the limited number of steps in gradient descent algorithms, which may restrict the parameter estimates from reaching the critical points of the Poisson deviance loss function (Wüthrich (2020)). For models under the negative binomial distributional assumption, the adopted log link function (which is not the canonical link function for the negative binomial distribution), can also contribute to bias in the results (Hilbe (2011)).

A common bias regularisation approach is to adjust the intercept $\beta_0^{(d+1)}$ in the linear function from the last layer of the neural network, which can be implemented by multiplying the results with a constant c given by

$$c = \frac{\bar{\mu}}{\hat{\mu}}, \tag{5.21}$$

where $\hat{\mu}$ is the mean of the predicted values $\hat{\mu}_i$ and $\bar{\mu}$ is the mean of the observed admission numbers y_i (Tzougas and Li (2021)). This will ensure that the means of the modified predicted values are the same as the means of the observed values y_i,

such as

$$\frac{1}{n}\sum c\hat{\mu}_i = c\frac{1}{n}\sum \hat{\mu}_i = \frac{\bar{\mu}}{\hat{\mu}}\hat{\mu} = \bar{\mu}. \tag{5.22}$$

As an alternative approach, the GLM regularisation method was also considered. Under this approach, a GLM is added after the last hidden layer in the neural network. In essence, the neural network acts as a pre-process and the output from the last hidden layer is used to fit a GLM to predict the response. The last hidden layer $\boldsymbol{z}^{(d)}$ is considered a learned representation of the feature space $\mathcal{X}$ created by the network

$$\mathcal{X} \mapsto \boldsymbol{z}^{(d)} \tag{5.23}$$

which could be viewed as a new augmented data set $(\hat{\mathcal{D}} = y_i, \boldsymbol{z}_i^{(d)} : i = 1, ..n)$ for modelling admission counts with n being the number of records in the original data set (Ferrario et al. (2020)). In other words, the GLM produces IWLS estimates for the weights in the last layer, thereby ensuring unbiasedness and balance. The sample code for implementing this approach is given in Listing A.7 and the pseudo-code is given in Algorithm A.2. For further discussion regarding the portfolio or population level unbalance of network results and details regarding the different bias regularisation approaches, we refer to Wüthrich (2021) and Wüthrich (2020). The GLM bias regularisation approach was considered using NN (20,15,10) and CANN (20,15,10) models under the negative binomial distribution assumption. This presents room for further potential improvement due to the choice of the log link function in the regression implemented for bias regularisation. Hence, the simple bias regularisation approach was also adopted, together with the regression bias regularisation method. In particular, for both the NN and CANN models, the bias regularisation was implemented by extracting their last hidden layers and feeding them into the corresponding negative binomial regression models for which their intercepts were adjusted in order to control the portfolio or population bias. Moreover, note that the bias regularisation approaches were applied in addition to the model improvement approaches of early stopping and dropout, with a batch size of 30,000 and 500 epochs. Table 5.7 shows the performance of the different models with and without

bias-regularisation. Bias-regularisation has a clear impact, with the average fitted mean of the relevant methods being the same as that of the observed data.

Model	Learning Loss	Testing Loss	Average Fitted Mean
Data			0.0027
NB.reg	1.0599	1.0599	0.0028
NB.reg w/bias regu	1.0600	1.0595	0.0027
NN_{NB} (20,15,10)	1.0381	1.0251	0.0028
NN_{NB} (20,15,10) w/bias regu	1.0379	1.0244	0.0027
$CANN_{NB}$ (20,15,10)	1.0435	1.0168	0.0028
$CANN_{NB}$ (20,15,10) w/bias regu	1.0431	1.0176	0.0027
NN_{NB} (25,20,15)	1.0424	1.0298	0.0029
NN_{NB} (25,20,15) w/bias regu	1.0418	1.0271	0.0027
$CANN_{NB}$ (25,20,15)	1.0454	1.0150	0.0029
$CANN_{NB}$ (25,20,15) w/bias regu	1.0442	**1.0115**	0.0027

Table 5.7: Testing loss, learning loss and average fitted mean of regression and network-based models under negative binomial distributional assumptions with and without bias regularisation.

5.5.2 Nagging predictor

As mentioned in Section 5.3.2.1, one of the main issues associated with neural network models is that results can vary among repeated runs. Aspects that can bring out this randomness are discussed in detail by Richman and Wüthrich (2020), and can include:

- The split of the learning data into training and validation sets;
- The split of the training data into mini-batches;
- Model initialisation.

These aspects are influenced by the choice of a random seed value for running the associated algorithms, and different seed values can potentially give slightly different

results. Although the differences might be small, they should not be ignored in the context of applications of incidence rate models. The nagging predictor proposed by Richman and Wüthrich (2020) acts as a sensible approach to tackle this. The approach acts similar to the traditional bagging and aggregating approach (bagging), but without re-sampling. In other words, the aggregation occurs over the network, i.e., over different calibrations (seed values) rather than by re-sampling. The data composition in terms of the split between learning and testing data remains the same for all the calibrations, whereas the aspects of randomness, such as the ones mentioned above, vary. For the i^{th} observation in the out-of-sample data, the nagging predictor is given by

$$\bar{\widehat{\mu}}_i^{(M)} = \frac{1}{M}\sum_{t=1}^{M} \widehat{\mu}_i^{(t)} = \frac{1}{M}\sum_{t=1}^{M} \mu(\boldsymbol{x}_i, \hat{\boldsymbol{\beta}}^{(t)})$$

where $\widehat{\mu}_i^{(t)}$ is the predictor obtained for the i^{th} observation with the t^{th} network calibration (e.g., as given in Equation (5.12)) and M is the number of calibrations (seed) values considered. The testing (out-of-sample) loss for the nagging predictors is given by:

$$\mathcal{L}\big(\mathcal{T}; \bar{\widehat{\mu}}_{i=1,\ldots,n}^{(M)}\big) = \frac{1}{n}\sum_{i=1}^{n} \delta\big(Y_i, \bar{\widehat{\mu}}_i^{(M)}\big)$$

where n is the number of observations in the testing data set and $\delta(Y_i, \bar{\widehat{\mu}}_i^{(M)})$ represents the unit deviance. The nagging predictor approach was applied on top of the already identified improvement approaches (early stopping and dropout) and bias regularisation. The nagging predictors for $M = 1, 2, 3 \ldots, 50$ were calculated for both the NN and CANN models under the negative binomial distributional assumption. Due to computational limitations, we only considered a (20,15,10) architecture for the NN and CANN models, and the results are shown in Table 5.8 and illustrated in Figure 5.9.

Model	Index M	Learning Loss	Testing Loss	Average Fitted Mean
NN_{NB} (20,15,10)	M = 50	1.0570	1.0647	0.0027
$CANN_{NB}$ (20,15,10)	M = 50	1.0605	1.0503	0.0027

Table 5.8: Testing loss, learning loss and average fitted mean of the nagging predictor (M = 50) for the NN (20,15,10) and CANN (20,15,10) models under the negative binomial distributional assumption.

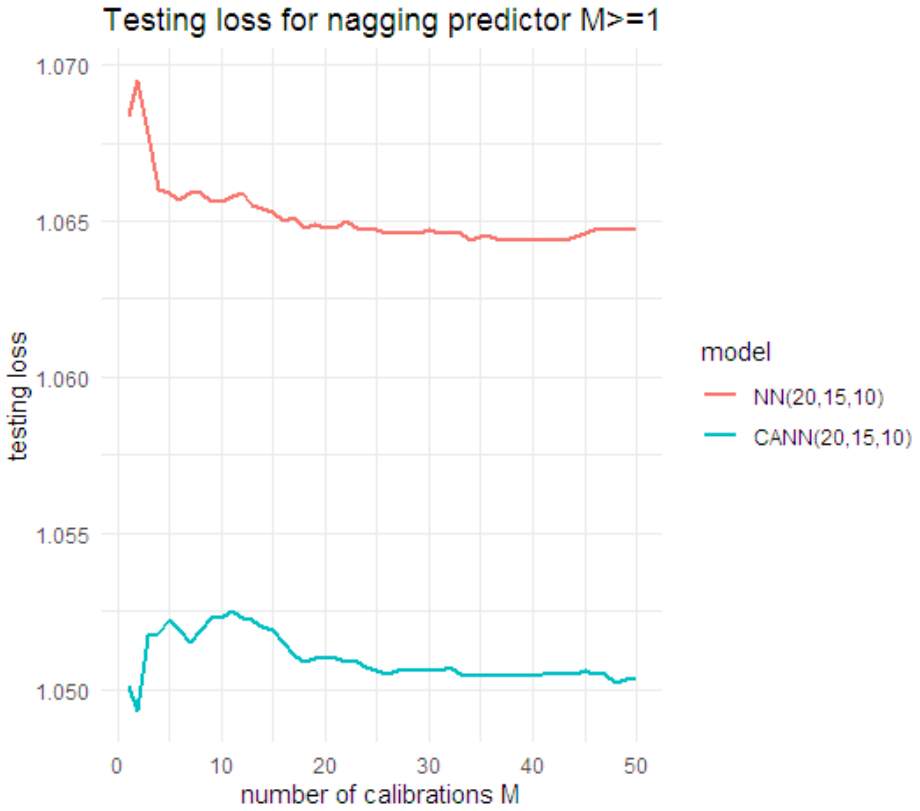


Figure 5.9: Testing loss for the nagging predictors under the negative binomial NN and CANN models, using (20,15,10) architecture.

The results suggest that the nagging predictor under the CANN approach performs better than the NN approach in terms of testing loss. Moreover, under both the NN and CANN approaches, as M increases, the testing loss starts to converge to a stabilised value, demonstrating a reduction in the variability of the prediction outcomes (see Algorithm A.3 for the pseudo code for nagging predictor) .

5.6 k-Fold Validation

In this section, we address the issue of potential variations in the NN model results, arising from the choice to split the data into learning and testing sets. The nagging predictor, presented in Section 5.5.2, does not account for this variability as it considers only those sources of randomness that arise once this split is done. Here, we consider a k-fold cross-validation approach to analyse the impact of the learning/testing split on the results and compare the performance of different models. Under the k-fold cross-validation approach introduced by Geisser (1975), the full data set is initially split into k roughly equal sets. The models under consideration are then trained using the $k-1$ set and validated/evaluated using the remaining set. The process is then repeated k times, altering the choice of validation set (Jung (2018)). In the current context, The value of k was set as $k = 10$ to maintain the 90:10 split of learning and testing data. Here, we used cross-validation as a model selection procedure as discussed by Arlot and Celisse (2010) and not for training the model. The different models were compared using the average deviance loss value from the 10 folds (e.g., as in Equation (5.17)). As the models under the negative binomial distribution assumption demonstrated better predictive performances in our earlier analysis, compared to those under the Poisson assumption, only the former were considered in the k-fold validation. A nagging predictor ($M = 25$), together with the model improvement approaches discussed earlier (early stopping and dropout), was considered here for the network-based models. Bias regularisation was also applied, implementing both approaches as described in Section 5.5.1. The average testing and learning loss over the ten different folds are given in Table 5.9.

Model	Index M	Learning Loss	Testing Loss	Average Fitted Mean
NB.reg		1.0664	1.0807	0.0027
NN_{NB} (20,15,10)	M = 25	1.0460	**1.0645**	0.0027
$CANN_{NB}$ (20,15,10)	M = 25	1.0563	1.0722	0.0027

Table 5.9: Average of testing loss, learning loss and average fitted mean of regression and network-based models under negative binomial distributional assumptions from the 10 different folds.

The results from the cross-validation indicate that the NN performed better than the regression and CANN models.

5.7 Impact of Data Structure and Composition on NN Models

Additionally, to analyse the impact of data structure and composition on NN models, the following analysis was undertaken using the admission counts data for respiratory diseases. Firstly, the same models were considered for data from different years, and we examined whether the sequential order of the models in terms of predictive performance is consistent over the years. Secondly, for all the years, we slightly altered the data composition by replacing the granular EGEOLOC variable with the REGION variable, and we examined the variation in performance. Each of these scenarios is analysed over the models with an underlying Poisson distributional assumption. For the network models, a (20,15,10) architecture with the 30,000 batch size and 500 epochs is used with the additional early stopping and dropout model improvements. The performance of the models over data from different years is given in Tables 5.10 and 5.11.

The results indicate that in all years except in 2017, under the case when the EGEOLOC variable was used, the NN model showed the best predictive performance. For the 2017 data, when the EGEOLOC variable was used, the CANN model outperformed the NN model. Although, in some instances, the regression model performed

better than the CANN model, this superior predictive performance was not observed when the alternative variable was used.

Model	Year	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg	2016	2.6811	2.5516	0.0027
NN(20,15,10)	2016	2.6634	**2.5419**	0.0026
CANN(20,15,10)	2016	2.6752	2.5463	0.0027
Pois.reg	2017	2.4546	2.4442	0.0025
NN(20,15,10)	2017	2.4227	2.4441	0.0026
CANN(20,15,10)	2017	2.4549	**2.4435**	0.0025
Pois.reg	2018	2.3993	2.4894	0.0024
NN(20,15,10)	2018	2.3826	**2.4845**	0.0022
CANN(20,15,10)	2018	2.3995	2.4906	0.0024
Pois.reg	2019	2.3380	2.3412	0.0023
NN(20,15,10)	2019	2.3198	**2.3346**	0.0024
CANN(20,15,10)	2019	2.3403	2.3449	0.0025

Table 5.10: Testing loss, learning loss and average fitted mean of regression and network-based models under the Poisson distributional assumption for admission counts data for respiratory diseases from different years, using the 'EGEOLOC' variable.

Model	Year	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg	2016	2.6912	2.5628	0.0027
NN(20,15,10)	2016	2.6557	**2.5576**	0.0026
CANN(20,15,10)	2016	2.6529	2.5668	0.0027
Pois.reg	2017	2.4631	2.4436	0.0025
NN(20,15,10)	2017	2.4327	**2.4348**	0.0027
CANN(20,15,10)	2017	2.4363	2.4355	0.0025
Pois.reg	2018	2.4049	2.4893	0.0024
NN(20,15,10)	2018	2.3865	**2.4831**	0.0023
CANN(20,15,10)	2018	2.3871	2.4842	0.0024
Pois.reg	2019	2.3447	2.3439	0.0023
NN(20,15,10)	2019	2.3251	**2.3258**	0.0024
CANN(20,15,10)	2019	2.3243	2.3378	0.0023

Table 5.11: Testing loss, learning loss and average fitted mean of regression and network-based models under the Poisson distributional assumption for admission counts data for respiratory diseases from different years, using the 'REGION' variable.

The above analysis showcased the data-driven nature of the neural network methodologies and how the variation in the structure and composition of the underlying

data set affect the model performance. Moreover, it highlights the need for additional tuning of hyper-parameters prior to adapting the model for a new data set. Furthermore, an examination of the impact of data granularity on the performance of the network-based models is carried out later on in Section 5.10 while comparing the NN models with the classical and Bayesian regression models.

5.8 NN Models for Admissions Due to Respiratory Diseases

The neural network models discussed so far in this chapter were developed using the admission counts data for respiratory diseases in 2016. In this section, we adapt the various modelling techniques for the multi-year data (data from years 2016-2019). As before, we consider individual-level data, which comprises 8,325,661 records. Instead of the EGEOLOC geographical variable, we use the less granular REGION variable. Hyper-parameter tuning was once again undertaken to identify the model architecture and the choice of batch size and epochs. Distinct model architectures of varying complexities with early stopping and dropout improvements were considered over different combinations of batch size and epochs (see Table A.16). From testing loss for the varied combination of model architecture, batch size and epochs, the (20,15,10) architecture with 75,000 batch size and 250 epochs was identified as a sensible choice. Thus, the above-mentioned model specification was adapted for all the different network-based models. All the models were considered both under Poisson and negative binomial distributional assumptions with additional bias regularisation applied to the models under NB distributional assumption. The predictive performance of the different models was then compared to the corresponding regression models (see Table 5.12)

Model	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg	106137.5	12067.7	0.0025
NN_{Pois}(20,15,10)	105380.6	11996.5	0.0026
CANN_{Pois}(20,15,10)	105217.0	12012.4	0.0025
NBI.reg	99346.1	11229.4	0.0025
NN_{NB}(20,15,10)	98690.4	**11165.7**	0.0025
CANN_{NB}(20,15,10)	98749.0	11182.3	0.0025

Table 5.12: Testing loss and learning loss in terms of NLL and average fitted mean of regression and network-based models under Poisson and NB distributional assumptions in the case of individual-level data for admissions due to respiratory diseases in the years 2016 to 2019.

The results in 5.12 indicate that under both distributional assumptions, the NN model gave the best performance, followed by the CANN and regression models. Moreover, both the NN and CANN models outperform the corresponding regression model under the same distributional assumption.

5.9 NN Models for Admissions Due to Neoplasms Among Males and Females

The development of NN models for admissions related to neoplasms was also carried out separately for males and females, as in the case of the regression and Bayesian models. As pointed out earlier, this is due to the significant variation observed between the rates of admission in males and females, arising from the prevalence of breast cancer in females. For males and females, the individual-level data from all the years was used, comprising 3,948,592 and 4,377,069 records, respectively. As for the admissions related to respiratory diseases, additional hyper-parameter tuning was undertaken separately for male and female data sets. Based on the results (see Tables A.17 and A.18), for the male data set, a (25,20,15) architecture with batch size 75,000 and 500 epochs was adopted, and for the female data set, a (15,10,5) architecture with 50,000 batch size and 500 epochs was considered. Using the above model specifications, separate NN and CANN models with underlying Poisson and

NB distributional assumptions were developed for the male and female populations, the results of which are shown in Tables 5.13 and 5.14. For the regression models, the model structure identified during the model fitting process of the classical regression model (Section 4.2) was used, and it thus contained an age polynomial of degree three.

Model	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg	63109.1	6837.8	0.0032
NN_{Pois}(25,20,15)	62148.7	6773.8	0.0032
CANN_{Pois}(25,20,15)	62157.8	6770.1	0.0032
NBI.reg	59878.5	6455.9	0.0032
NN_{NB}(25,20,15)	59154.2	**6408.2**	0.0032
CANN_{NB}(25,20,15)	59549.3	6425.9	0.0032

Table 5.13: Testing the loss and learning loss in terms of NLL and average fitted mean of regression and network-based models under Poisson and NB distributional assumptions in the case of individual-level data for neoplasm-related admissions in males from 2016 to 2019.

Model	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg	91242.5	10241.5	0.0043
NN_{Pois}(15,10,5)	89599.7	10075.9	0.0043
CANN_{Pois}(15,10,5)	89582.4	10080.5	0.0042
NBI.reg	87971.5	9940.0	0.0043
NN_{NB}(15,10,5)	86857.3	**9846.5**	0.0043
CANN_{NB}(15,10,5)	87853.3	9911.6	0.0043

Table 5.14: Testing the loss and learning loss in terms of NLL and average fitted mean of regression and network-based models under Poisson and NB distributional assumptions in the case of individual-level data for neoplasm-related admissions in females from 2016 to 2019.

In all instances, the models with the NB distributional assumption outperformed the corresponding models with a Poisson assumption. Furthermore, for the female population, under both distributional assumptions, the NN model had the best performance, followed by the CANN and regression models. Similarly, for the male population, the NN model performed better than the CANN model under the NB distributional assumption, whereas, under the Poisson distribution assumption, the

CANN model slightly outperformed the NN model.

5.10 Comparison of Bayesian and NN Methodologies

A comparison between the predictive performance of classical, Bayesian and NN models was carried out using a k-fold approach. Different models with an underlying Poisson distributional assumption were considered using the admissions data for respiratory diseases in 2016. The Bayesian models discussed in Chapter 4 were fitted using the risk profile data, whereas individual-level data were used for neural network models detailed in this Chapter. In order to make the models directly comparable, all the network-based models were refitted using risk profile level data. Among the Bayesian models, the BP-G model was considered, and for the network models, NN (20,15,10) and CANN (20,15,10) models with additional early stopping and dropout improvements were considered. So far, for the CANN Models, our primary focus has been on the approach where we keep the regression component or the skip connection non-trainable, and hence, now we also consider an additional variant where the regression component is also trained (see Equation 5.13). This resulted in the following two variants, which were then evaluated using the k-fold approach.

- Variant 1: the regression component in the CANN model is also trained alongside the network component. More specifically, the model parameter in the skip connection is also updated while training the model.
- Variant 2: the regression component in the CANN model is not trained, and the weights of the regression component are kept fixed as the iterated weighted least squares (IWLS) estimate from the corresponding regression model.

For all the models mentioned above, a 90:10 split of the data set, comprising 104,930 records, was used to create training and testing sets. Additionally, for the k-fold

approach, ten folds ($k = 10$) were considered, thus resulting in mutually exclusive training sets under each fold. The previously identified combination of architecture (20,15,10), batch size (30,000) and epochs (250) was used for all the network models, with the additional early stopping and dropout approaches. The performance of each model was compared using the average deviance loss for testing and training sets. The performance of the different models under each fold in terms of deviance loss is shown in Table A.19, and the average deviance loss is shown in Table 5.15.

Model	**Learning Loss**	**Testing Loss**	**Average Fitted Mean**
Data			0.0027
Pois.reg	16.747	16.849	0.0027
BP-G	16.771	16.785	0.0027
NN_{Pois} (20,15,10)	22.762	22.765	0.0061
CANN_{Pois}(20,15,10): variant 1	20.089	20.095	0.0027
CANN_{Pois}(20,15,10): variant 2	16.749	16.853	0.0027

Table 5.15: Average testing loss, learning loss and average fitted mean of classical, Bayesian regression and network-based models under the Poisson distributional assumptions from the ten different folds.

The results indicate that the BP-G model has better predictive performance than the other models in terms of testing loss, followed by the regression model. Compared to the neural network (NN) and variant 1 of the CANN model, the average testing loss for variant 2 of the CANN model was significantly lower and closer to that of the regression model. This could be attributed to the fact that, in variant 2, since the regression component is not trained, the regression model is encompassed within the CANN model as it is. Moreover, unlike in the case of the NN model, even without additional hyper-parameter tuning, the CANN: variant 2 model performs well for the aggregate level data.

In this analysis so far, the choice of hyper-parameters, such as batch size, epochs, and architecture, was identified using individual-level data. This has potentially put the network-based models, particularly the NN model, at a disadvantage, and may have caused the inferior predictive performance. In order to overcome this issue,

separate hyper-parameter tuning was carried out for each of the different network-based models using the risk profile-level data. As before, different architectures of varying complexity were considered for each model over different combinations of batch size and epochs. For all the models, in order to avoid over-fitting, an early stopping using callback was employed. The results from the hyper-parameter tuning for NN, CANN: variant one and CANN: variant two models are provided in Tables A.20, A.21 and A.22, respectively.

It appears from the results that the combination of 5,000 batch size and 1,000 epochs is a good choice across all three network models; hence, the same was adopted. Furthermore, for the NN model with the above combination of 5,000 batch size and 1000 epochs, the (25,20,15) architecture gave the best performance, followed by the (15,10,5) and (20,15,10) architectures. Likewise, for the CANN models, (50,35,25) and (75,50,25) architectures were adopted for variants 1 and 2, respectively. In addition to the best-performing architecture for each of the models, the (20,15,10) architecture was also considered for comparative reasons as in the k-fold comparison carried out prior to the additional hyper-parameter tuning, the (20,15,10) architecture was used. The purpose of also using the same architecture was to identify and compare the improvement in predictive performance obtained from the new combination of batch size and epochs identified from the hyper-parameter tuning exercise. Furthermore, a variant of the Poisson GLM and the BP-G models, incorporating the SEX: EMPREL interaction term identified during model formulation (see Table 4.7), was also considered to assess the variation in predictive performance. The performance of the NN and CANN models with the new combinations of architectures, batch size and epochs along with the Poisson GLMs and the Bayesian Poisson-Gamma models under each fold is shown in Table A.23, and the average performance is shown in Table 5.16.

Model	Learning Loss	Testing Loss	Average Fitted Mean
Data			0.0027
Real data			0.0027
Pois.reg	16.747	16.849	0.0027
BP-G	16.771	16.785	0.0027
Pois.reg w/ interaction	16.711	16.816	0.0027
BP-G w/ interaction	16.749	16.865	0.0028
NN_{Pois} (20,15,10)	16.378	16.652	0.0027
CANN_{Pois} (20,15,10): variant 1	16.836	16.956	0.0027
CANN_{Pois} (20,15,10): variant 2	16.475	16.830	0.0027
NN_{Pois} (25,20,15)	16.362	16.686	0.0025
CANN_{Pois}(50,35,25): variant 1	16.403	16.695	0.00246
CANN_{Pois}(75,50,25): variant 2	16.490	16.814	0.0027

Table 5.16: Average testing loss, learning loss and average fitted mean for classical, Bayesian regression, and network-based models under the Poisson distributional assumption from the ten different folds after hyper-parameter tuning of neural network models for risk profile level data.

Over the ten folds, the NN (20,15,10) model exhibited the best average performance, followed by the NN (25,20,15) and the CANN (50,35,25): variant 1 models. Without the interaction term, the BP-G model outperformed the corresponding GLM model. The average predictive performance of the GLM model improved with the addition of the interaction term, whereas that of the BP-G model decreased when the interaction term was introduced.

It is also worth noting that a limitation of the above comparison of the predictive accuracy of various models is that it is based on single-point estimates and does not take into account the uncertainty around predictions. This is not ideal, particularly for Bayesian models, where the most significant advantage lies in the use of the entire posterior distribution, which can also capture the uncertainty around predictions. Moreover, only a finite small number of MCMC iterations were considered under each fold, affecting accuracy, and this highlighted the impact of aspects such as the burn-in period and number of iterations on the performance of Bayesian models.

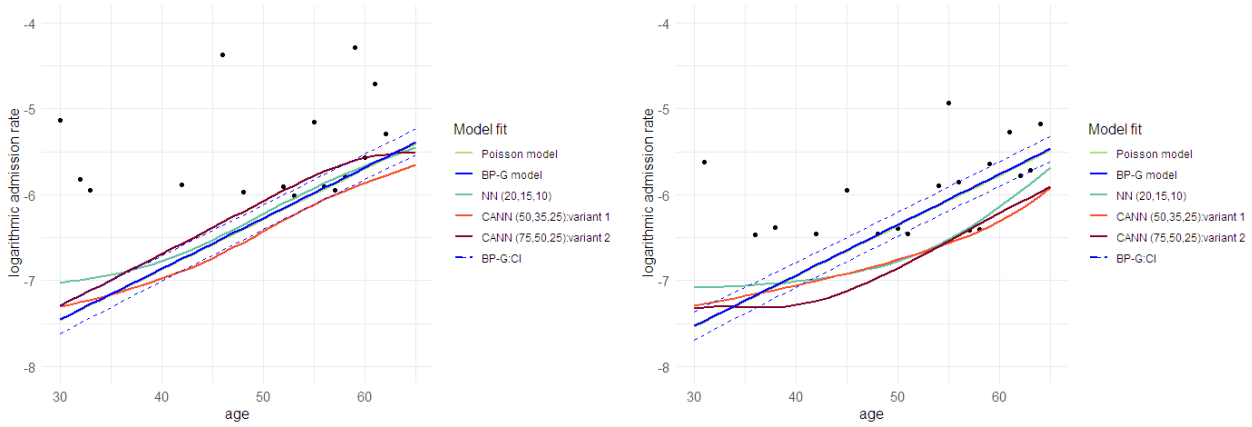
In general, network-based models appear to outperform the GLM and Bayesian models in terms of predictive performance. Therefore, for the remainder of the thesis,

we primarily concentrate on neural network methodology and develop network-based models that can more effectively account for the excess zero nature of the count data and compare them with the corresponding regression model (see Chapter 6).

Furthermore, the deviance loss for the testing set indicates that for the network models, particularly for the NN and CANN variant 1 models, even with the same architecture, the performance has improved significantly for the new batch size and epochs compared to the results reported in Table 5.15. Additionally, in the case of the CANN models, the new choice of architecture is performing better on average than the previously used architecture (20,15,10). Unlike this, for the NN model, the (20,15,10) architecture performs better than the (25,20,15) architecture identified via hyper-parameter tuning. Furthermore, the results in Table A.23 clearly indicate that the sequential order of the models in terms of predictive performance, mainly that of the network-based models, varies over the different folds. This showcases the significance of the choice of hyper-parameters as well as the data dependency of the deep learning methods.

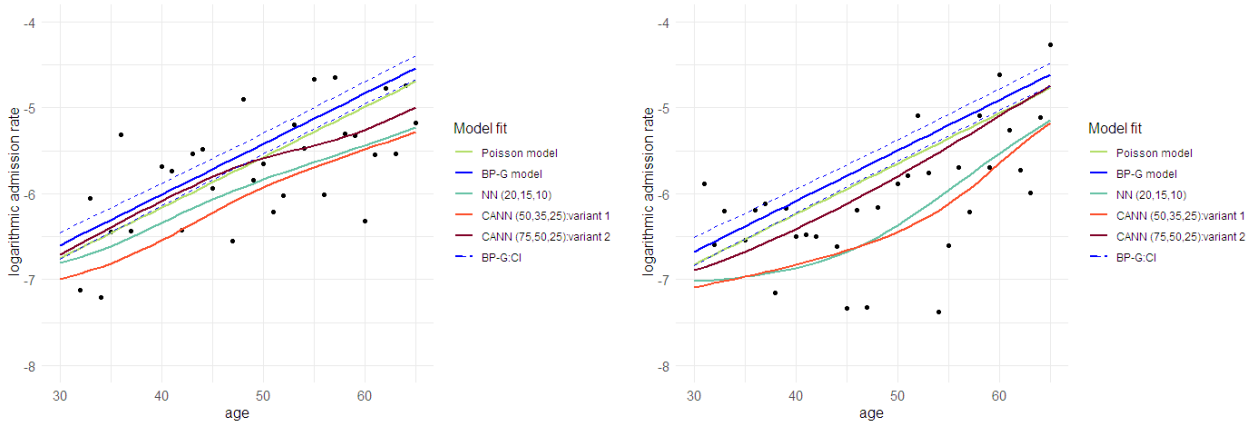
5.10.1 Plots of predictive curves

In order to better understand the predictive behaviour of the models mentioned above, we also considered the predicted curves representing the logarithmic admission rates generated by the models. To achieve this, specific risk profiles were systematically selected as the testing set over which the predicted curves are analysed. All the models were then fitted/trained using the entire rest of the data set. Following this, the predicted curves from the different models are plotted for each of the risk profiles that have been set aside as part of the testing set (see Figures 5.10 and 5.11). For the NN and CANN models, the best-performing architectures were considered with 5,000 batch size and 1000 epochs and for the BP-G model, the model was trained over 30,000 iterations with 10,000 being burin-period iterations.



(a) (b)

Figure 5.10: Observed logarithmic admission rates and fitted rates under different models along with CI intervals for Bayesian models. Risk profile: UR-urban; REGION-south; EECLASS-salary non-union; EESTATU-active fill-time; EMPREL-employee; PLANTYP- PPO. (**a**) females; (**b**) males.



(a) (b)

Figure 5.11: Observed logarithmic admission rates and fitted rates under different models along with CI intervals for Bayesian models. Risk profile: UR-urban; REGION-south; EECLASS-unknown; EESTATU-unknown; EMPREL-employee; PLANTYP- PPO. (**a**) females; (**b**) males.

As anticipated, for the different risk profiles, Bayesian and classical regression models produce a straight line, while the network-based models have a more flexible shape. Moreover, in all instances, the network models seem to have better accounted for and predicted the variations in the admission rates across different ages.

5.11 Chapter Summary

To sum up, in this chapter, different network-based models were developed for predicting the rate of admissions related to neoplasms and chronic respiratory diseases. The results indicate that the neural network-based models have better predictive performance compared to traditional GLM-type models. More specifically, our k-fold cross-validations (see Sections5.6 and 5.10) indicated that for both the Poisson and negative binomial distributional assumptions, the NN models gave better average predictive performance, as determined by the testing data loss, than the corresponding GLM-type and CANN models. The performance of the CANN model seems to be improving when the regression component is also trained while fitting the model. The alternative variant of the CANN model (regression part not trained), on the other hand, although it has a lower predictive performance, appears to be less sensitive to the choice of hyper-parameters.

Furthermore, the performance of the network-based models seems to be influenced by the structure, composition and granularity of the underlying data set. Additional model improvement approaches, such as early stopping and dropout, further improved the predictive performance of the models. The nagging predictor addresses the inherent randomness in neural network results, and the adapted bias regularisation approach effectively resolves the population level bias in the model results.

In the next chapter, we develop zero-inflated versions of the NN-based models considered in this chapter, motivated by the high concentration of zeros in data of this nature. Additionally, we develop interpretable versions of the network models that will aid in overcoming the general lack of interpretability associated with NN models without compromising the predictive performance.

Chapter 6

Zero-Inflated Models and Neural Network Based Extensions

One of the main issues associated with modelling hospital admission frequency data and similar type of rare event count data has been the zero-inflated nature of the data leading to over-dispersion. While the negative binomial regression model can handle over-dispersion, the issue of excessive number of zeros may not be effectively addressed (Gurmu and Trivedi (1996)).

In this chapter, we construct zero-inflated regression and interpretable zero-inflated neural network models for modelling the previously considered data on hospital admission counts related to respiratory diseases among the US insured population. Certain parts of the work presented in this chapter have been discussed in Jose et al. (2024). Specifically, we extend the NN and CANN approaches discussed in the previous chapter and construct a Zero-inflated Poisson Neural network (ZIPNN) and Zero-inflated Poisson Combined Actuarial Neural network (ZIPCANN) models for modelling hospital admission frequency data. By embedding a zero-inflated Poisson regression into the neural network framework, we can explore the gains in predictive power compared to conventional regression models, and at the same time, we can accommodate the high presence of zeros in the count data. Furthermore, we adopt the LocalGLMnet approach from Richman and Wüthrich (2022) for interpreting

the ZIPNN model results. This facilitates the analysis of the impact of a number of socio-demographic factors on the admission rates related to respiratory disease while benefiting from improved predictive performance.

The real-life utility of the methodologies developed as part of this work lies in the fact that they facilitate accurate rate setting, in addition to offering the potential to inform health interventions. More specifically, insightful forecasting of hospital admission rates would assist hospitals in anticipating fluctuations in demand and thereby enhancing the overall quality of patient care. Furthermore, retaining interpretability is crucial for investigating the impact of covariates or input variables (see Table 5.1) on the admission counts.

One drawback of NN models, as discussed in the previous chapter, is the potential failure to maintain the balance property, thereby leading to bias at the population level (see Wüthrich (2020), Wüthrich (2021)). Several bias regularization approaches, such as those examined in the previous chapter, have been proposed to address this problem. Additionally, autocalibration, considered by Denuit et al. (2021), is a technique that can be employed to mitigate population-level bias, and for this reason, it has gained wide popularity recently. Further criticism for NN models relates to the lack of explainability and interpretability of the results due to the inherent black box nature arising from their complex structure. Several techniques, such as SHapley Additive exPlanations (SHAP) by Lundberg and Lee (2017), LocalGLMet from Richman and Wüthrich (2022), locally interpretable model-agnostic explanation (LIME) proposed by Ribeiro et al. (2016) etc., have been developed to address the interpretability issue. Out of these different approaches, here we adopt and extend the LocalGLMnet approach, owing to its ease of implementation and the likeness that it provides to interpretations derived from traditional regression models.

It should be noted that interpreting results from a ZIP distributional assumption is more complex compared to the Poisson distributional assumption counterpart. This complexity arises due to the model's mixture nature, wherein input features

are incorporated through the rate and probability parameters. Furthermore, the interpretations obtained from the NN-based model are compared to those derived with the coefficient estimates of the ZIP regression model. This comparison showcases the impact of potential non-linear interactions captured by the NN on the final results.

Concerning the above-mentioned work, we have taken into account a wide range of alternative approaches that have been developed in the recent actuarial literature concerning network-based models. For instance, Noll et al. (2020) and Gao et al. (2019) consider NN models in the context of motor insurance. Hejazi and Jackson (2016) propose a network-based approach for the valuation of large portfolios of variable annuities, while Kuo (2019) consider a deep learning approach to loss reserving. Additionally, Hainaut (2018) adapt NNs for mortality forecasting. Richman (2021) provides a detailed review of recent advances of Artificial Intelligence (AI) in actuarial science. For more details regarding the application of NNs in an actuarial context, please refer to textbooks Denuit et al. (2019) and Wüthrich and Merz (2023). Moreover, it is worth noting that the above approaches and their combinations are used for the first time in the literature regarding the zero-inflated Poisson model for the case of hospital admission data.

6.1 Data Used and Supplemental Feature Processing.

The various models mentioned above were developed using the individual-level admission counts data due to respiratory diseases in 2016. The details regarding the different features within the data set are given in Table 5.1. Note that for all the models discussed in the chapter, the REGION geographical variable was chosen over the more granular EGEOLOC variable for ease of interpretation. Prior to carrying out any modelling, in addition to the data considerations detailed in Chapter 3, feature pre-processing for network models (see Section 5.1) was also undertaken. More

specifically, a min-max scaler was applied to the continuous and binary variables. On the other hand, one hot encoding was used instead of data embedding for the categorical variables. Thus a categorical variable with l categories $c_1, c_2, .., c_l$was represented using a l dimensional feature vector which is of the form

$$x_j \mapsto \left(\mathbb{1}_{\{x_j=c_1\}}, ..., \mathbb{1}_{\{x_j=c_l\}}\right)^T \in \mathbb{R}^l. \tag{6.1}$$

Following the data pre-processing steps, as always, a 90:10 random split of the entire data set was created to be used as the learning $\mathcal{D}$ and testing $\mathcal{T}$ data sets. All the models were fitted using the learning data set, and the performance of the models on the testing data set was used to compare their performances.

6.2 Zero-inflated Models

The main models discussed as part of this chapter are the ZIPNN and ZIPCANN models and their extensions developed by incorporating the LocalGLMnet approach, which facilitates the interpretation of the results. A zero-inflated Poisson (ZIP) regression model is also considered for comparing the predictive performance between traditional regression modelling and network-based approaches. Before extending the LocalGLMnet approach to the ZIPNN model the same was implemented for the NN model.

6.2.1 Zero-inflated Poisson regression

As mentioned earlier, zero-inflated models are based on a mixture distribution comprising two components (Cameron and Trivedi (2013) and Lambert (1992)). In practice, a count distribution such as the Poisson or negative binomial distribution is used to represent the count component, and a Bernoulli distribution is assumed for the zero component. Hence a zero-inflated Poisson (ZIP) regression model is of

the form

$$Y_i \sim \begin{cases} 0 & \text{with probability } \pi_i \\ Poisson(\lambda_i e_i) & \text{with probability } (1-\pi_i) \qquad \text{... for } \; i = 1, ..., n \end{cases} \tag{6.2}$$

with $\mu_i = \lambda_i e_i$ being the mean of the Poisson part of the ith record with exposure e_i and rate parameter λ_i, while π_i represents the probability of only having zero admissions. Zeros arise from both the zero component and the count component, with the two components having probability π_i and $(1-\pi_i)$ respectively. Hence, the probability mass function (PMF) of the ZIP mixture distribution is

$$Pr(Y_i = y_i) = \begin{cases} \pi_i + (1-\pi_i)e^{-\mu_i}, & y_i = 0 \\ \\ (1-\pi_i)\frac{e^{-\mu_i}\mu_i^{y_i}}{y_i!} & y_i > 0 \end{cases} \tag{6.3}$$

and the corresponding mean and variance are given by

$$E(Y_i) = (1-\pi_i)\mu_i, \qquad V(Y_i) = (1-\pi_i)\mu_i(1+\pi_i\mu_i). \tag{6.4}$$

The ZIP model allows both μ_i and π_i to be modelled using a set of covariates. This would be of the form :

$$\log(\mu_i) = o_i + \beta_0 + \boldsymbol{\beta}_{reg}^\top \boldsymbol{x_i} = o_i + \beta_0 + \langle \boldsymbol{\beta}_{reg}, \boldsymbol{x_i} \rangle \tag{6.5}$$

and

$$logit(\pi_i) = \gamma_0 + \boldsymbol{\gamma}^\top \boldsymbol{w_i} \tag{6.6}$$

where $o_i = \log(e_i)$ is the offset term, β_0, γ_0 the intercept terms and $\{\boldsymbol{\beta}_{reg}^\top, \boldsymbol{\gamma}^\top\} = (\beta_1, .., \beta_q, \gamma_1, .., \gamma_s)$ being the unknown vector of coefficients to be estimated corresponding to the sets of covariates $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,q})^\top$ and $\boldsymbol{w}_i = (\boldsymbol{w}_{i,1}, \ldots, \boldsymbol{w}_{i,s})^\top$ considered in the two regression functions with dimensions $q \times 1$ and $s \times 1$ respec-

tively. Regarding the treatment of exposure within the ZIP model, it is possible that the exposure or period at risk could potentially influence the probability of admission, as well as the rates of admission. Consequently, the exposure could be factored into both components, as detailed in Feng (2022). Here, we follow the general practice of treating it as an offset term in the regression function for μ, as discussed in Lee et al. (2001).

A model selection procedure was carried out for both count and zero components of the ZIP regression model. All variables entered the model for the count component, and thus, a full model with all the covariates was utilised for the Poisson regression function. Similarly, for the regression function associated with the zero component, a forward step-wise variable selection process based on the Bayesian information criterion (BIC) was carried out(Vrieze (2012)). As we are using logistic regression to model the probability parameter, we refer to it as the logistic component from here on. The summary of the variable selection process of the logistic component of the model is given in Table 6.1. Under the forward step-wise variable selection process, we start from a model without any terms in the logistic component and consider adding one variable at a time. The outcome of any particular step is the model that yields the lowest BIC value, and the whole process is repeated until there is no further reduction in the BIC value. The model thus identified from the variable selection process, had the vector of coefficients $\{\beta_0, \boldsymbol{\beta}_{reg}, \gamma_0, \boldsymbol{\gamma}\} = (\beta_0, \beta_1, .., \beta_q, \gamma_0, \gamma_1)$ corresponding to the complete set of covariates as presented in Table 6.2. The ZIP regression model was fitted using the `zeroinfl()` function in the `pscl` package which employs the `optim()` function to estimate the model parameters in both components simultaneously, by maximum likelihood, using optimisation algorithms such as Nelder-Mead or a quasi-Newton method (Zeileis et al. (2008a)).

Model	Degrees of freedom	BIC
Step 1: ∼1		
Null model	34	53847.78
+ AGE	35	**53829.5**
+ REGION	38	53890.38
+ SEX	35	53860.14
+ UR	35	53862.2
+ EECLASS	42	53945.26
+ EESTATU	42	53925.05
+ EMPREL	36	53867.1
+ PLANTYP	41	53910.87
Step 2: ∼AGE		
+ REGION	39	53873
+ SEXRT	36	53840.2
+ URRT	36	53843.82
+ EECLASS	43	53927.63
+ EESTATU	43	53914.32
+ EMPREL	37	53844.43
+ PLANTYP	42	53898.87

Table 6.1: Summary of the variable selection process of the logistic component of the ZIP regression model.

Covariate	Coefficient	Description
Count component		
Intercept	β_0	$\beta_{intercept}$
AGE	β_1	β_{age}
REGION	$\{\beta_2, ..., \beta_6\}$	β_{region_a} $a = 1, .., 5$
SEX	β_7	β_{sex}
UR	β_8	β_{ur}
EECLASS	$\{\beta_9, ..., \beta_{17}\}$	$\beta_{eeclass_b}$ $b = 1, ..., 9$
EESTATU	$\{\beta_{18}, ..., \beta_{26}\}$	$\beta_{eestatu_c}$ $c = 1, ..., 9$
EMPREL	$\{\beta_{27}, ..., \beta_{29}\}$	β_{empred_d} $d = 1, ..., 3$
PLANTYP	$\{\beta_{30}, ..., \beta_{37}\}$	$\beta_{plantyp_e}$ $e = 1, ..., 8$
Logistic component		
Intercept	γ_0	$\gamma_{intercept}$
AGE	γ_1	γ_{age}

Table 6.2: List of covariates and the corresponding coefficient parameters in both components of the ZIP regression model

6.2.1.1 Impact of varying exposure

At this point, we also investigate the impact of varying exposure. For instance, the ZIP model considered in the above section did not consider exposure in the zero component and it is possible that the exposure has an impact on the probability of excess zeros (Feng (2022)). In order to assess this, different variants of the ZIP model were considered. As in the previous model, the complete set of covariates was used for the count component, whereas in the zero component, only the AGE variable was used. In each of these variants, $\log(e_i)$ term was used as an offset or covariate in either component.

- Variant 1: $\log(e_i)$ as an offset in the count component.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^{\top} \boldsymbol{x_i} + \log(e_i) \end{aligned} \tag{6.7}$$

- Variant 2: $\log(e_i)$ as an offset in the zero component.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} + \log(e_i) \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^{\top} \boldsymbol{x_i} \end{aligned} \tag{6.8}$$

- Variant 3: $\log(e_i)$ as an offset in both components.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} + \log(e_i) \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^{\top} \boldsymbol{x_i} + \log(e_i) \end{aligned} \tag{6.9}$$

- Variant 4: $\log(e_i)$ as a covariate in the count component.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^{\top} \boldsymbol{x_i} + \xi_2 \log(e_i) \end{aligned} \tag{6.10}$$

- Variant 5: $\log(e_i)$ as a covariate in the zero component.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} + \xi_1 \log(e_i) \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^\top \boldsymbol{x_i} \end{aligned} \tag{6.11}$$

- Variant 6: $\log(e_i)$ as a covariate in the both components.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} + \xi_1 \log(e_i) \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^\top \boldsymbol{x_i} + \xi_2 \log(e_i) \end{aligned} \tag{6.12}$$

- Variant 7: $\log(e_i)$ as a covariate in zero component and as an offset in the count component.

$$\begin{aligned} \text{logit}(\pi_i) &= \gamma_0 + \gamma_{age} + \xi_1 \log(e_i) \\ \log(\lambda_i) &= \beta_0 + \boldsymbol{\beta}_{reg}^\top \boldsymbol{x_i} + \log(e_i) \end{aligned} \tag{6.13}$$

Model	# number of estimated parameters	BIC	AIC
Variant 1	35	53829.50	53394.52
Variant 2	35	56594.07	56159.09
Variant 3	35	53738.94	53303.96
Variant 4	36	**53590.29**	53142.88
Variant 5	36	53617.82	53170.41
Variant 6	37	53601.44	**53141.61**
Variant 7	36	53595.68	53148.27

Table 6.3: AIC and BIC values of the different variants of ZIP model

The results indicate that variant 4, in which $\log(e_i)$ is treated as a covariate in the count component, has the best fit in terms of BIC. Even when considering AIC, the difference in values between variant four and the model with the best fit according to AIC is minimal. This intensifies the doubt regarding exposure having a more direct impact on admission rate and not just as a measure of time at risk. Furthermore, variants 6 and 7 also showed improved fit compared to the conventionally used variant 1. While the results support the alternative treatment of

exposure, the exploratory analysis (see Section 3.4) did not reveal any clear pattern of dependence between exposure and admission rates. While we remain skeptical about the treatment of exposure due to a lack of more substantiating evidence, we proceed by following the general practice of treating it as an offset term in the count component, as described in Lee et al. (2001).

6.2.2 Zero-inflated Poisson Neural Network (ZIPNN) model

A generic feed forward NN (model 3 in Table 6.4) comprises an input layer, followed by multiple hidden layers and then the output layer. A feature space $\mathcal{X}$ is taken as the input layer with dimension q_0. Assuming a network architecture of $d \in \mathbb{N}$ hidden layers with $q_m \in \mathbb{N}, 1 \leq m \leq d$ neurons in each of the layers then a neuron $z_j^{(m)}, 1 \leq j \leq q_m$, in the m^{th} hidden layer $\mathbf{z}^{(m)}$ is given by

$$z_j^{(m)}(\mathbf{z}) = \psi\left(\langle \boldsymbol{\beta}_j^{(m)}, \mathbf{z} \rangle\right), \tag{6.14}$$

with $\mathbf{z}^{(m)}$ represented as

$$\mathbf{z}^{(m)} : \mathbb{R}^{q_{m-1}} \to \mathbb{R}^{q_m}, \quad \mathbf{z} \mapsto \mathbf{z}^{(m)}(\mathbf{z}) = (1, z_1^{(m)}(\mathbf{z}), \ldots, z_{q_m}^{(m)}(\mathbf{z}))^\top, \tag{6.15}$$

inclusive of the intercept component, where $\boldsymbol{\beta}_j^{(m)} = (\beta_{l,j}^{(m)})_{0 \leq l \leq q_{m-1}}^\top \in \mathbb{R}^{q_{m-1}+1}$ are the network parameters and $\psi : \mathbb{R} \to \mathbb{R}$, the activation function. The network parameters corresponding to the hidden layer $\mathbf{z}^{(m)}$ are the $\left(\boldsymbol{\beta}_1^{(m)}, \ldots, \boldsymbol{\beta}_{q_m}^{(m)}\right) \in \mathbb{R}^{q_m}$. The comprehensive set of network parameter is given by $\boldsymbol{\beta} = \left(\boldsymbol{\beta}_1^{(1)}, \ldots, \boldsymbol{\beta}_{q_d}^{(d)}, \boldsymbol{\beta}^{(d+1)}\right) \in \mathbb{R}^r$, where the dimension is $r = \sum_{m=1}^{d} q_m \left(q_{m-1} + 1\right) + (q_d + 1)$.The predictor of the NN obtained from the output layer is then of the form

$$(o_i, \boldsymbol{x}_i) \mapsto \log(\mu_i^{NN}) = o_i + \langle \boldsymbol{\beta}^{(d+1)}, (\mathbf{z}^{(d)} \circ \cdots \circ \mathbf{z}^{(1)})(\boldsymbol{x}_i) \rangle, \tag{6.16}$$

for $i = 1, \ldots, n$, where $\boldsymbol{\beta}^{(d+1)} \in \mathbb{R}^{q_d+1}$ are the weights associated with the output layer which connects the last hidden layer $\mathbf{z}^d$ to the output layer $\mathbb{R}_+$. For a

ZIPNN model, the feature space $\mathcal{X}$ with dimension q is taken as the input layer, i.e. $q_0 = q$. The construction of ZIPNN models involves incorporating zero-inflation considerations through the establishment of a distribution layer within the NN. This distribution layer, referred to as a lambda layer[1] in machine learning terminology, is a component in a NN that facilitates the integration of custom operations by defining a function applied to input data. In particular, this functionality allows for the inclusion of specific computations not covered by standard layers. In the specific context of zero-inflated NNs, the distribution lambda layer is employed to integrate a zero-inflated distribution assumption into the NN, enabling the model to capture complex relationships in the data (Dürr et al. (2020)). Also, we take $q_d = 2$ with $\psi : f(x) = x$. In other words, a dense layer with two neurons without activation is defined before the distribution layer.

The output of these two neurons is then used to calculate the probability (π) and rate (λ) parameters of the zero-inflated Poisson distribution constructed using the distribution layer that follows. The distribution layer thus created has an underlying zero-inflated Poisson mixture distribution. The model output is then derived as the mean of the distribution given by

$$E(Y_i) = (1 - \pi_i^{zipnn})\mu_i^{zipnn} \tag{6.17}$$

where π_i^{zipnn} and μ_i^{zipnn} are the rate and probability parameters of the underlying zero-inflated Poisson distribution. As in the case of a ZIP model, the exposure is taken as an offset term in the count component of the ZIPNN model. An exponential transformation is applied to the sum of the offset term and the output from the neuron associated with λ. A sigmoid function is applied to the output from the neuron associated with π to restrict the value of π to $[0, 1]$ interval. The μ_i^{zipnn} is

[1]For more details regarding the lambda layer, refer to https://keras.io/api/layers/core_layers/lambda/.

then given by

$$\log(\mu_i^{zipnn}) = z_1^{(d)}(\mathbf{z}) = \psi\left(\langle \boldsymbol{\beta}_1^{(d)}, \mathbf{z}\rangle\right) + o_i = \langle \boldsymbol{\beta}_1^{(d)}, \mathbf{z}\rangle + o_i \tag{6.18}$$

or equivalently

$$\log(\mu_i^{zipnn}) = o_i + \langle \boldsymbol{\beta}_1^{(d)}, (\mathbf{z}^{(d-1)} \circ \cdots \circ \mathbf{z}^{(1)})(\boldsymbol{x}_i)\rangle. \tag{6.19}$$

Similarly, π_i^{zipnn} is of the form

$$logit(\pi_i^{zipnn}) = z_2^{(d)}(\mathbf{z}) = \psi\left(\langle \boldsymbol{\beta}_2^{(d)}, \mathbf{z}\rangle\right) = \langle \boldsymbol{\beta}_2^{(d)}, \mathbf{z}\rangle. \tag{6.20}$$

$\boldsymbol{\beta}_1^{(d)}$ and $\boldsymbol{\beta}_2^{(d)}$ represent the network weights associated with the two neurons in the last hidden layer corresponding to probability and rate parameters. In essence the Equations 6.5 and 6.6 are replaced by Equations 6.19 and 6.20. Alternatively we could also define

$$logit(1 - \pi_i^{zipnn}) = z_2^{(d)}(\mathbf{z}) = \psi\left(\langle \boldsymbol{\beta}_2^{(d)}, \mathbf{z}\rangle\right) = \langle \boldsymbol{\beta}_2^{(d)}, \mathbf{z}\rangle. \tag{6.21}$$

Since the zero-inflated distribution is defined as a mixture distribution using the distribution layer, the models could also be constructed using the $p_i = (1 - \pi_i)$ parameter with ease. This means that instead of π, the probability of zero component, the distribution layer considers p, which is the probability of count component or, in other words, the probability of having a non-zero admission. The only difference is how the mixture distribution is defined. For ease of interpretation, while using the attention layers, we have adopted the latter approach. A schematic representation of a ZIPNN with 20,15,10,2 neurons (model 5 in Table 6.4) in each dense layer is shown in Figure 6.1. Code for implementing the same is given in Listing A.8, and the pseudo-code detailing the steps involved is given in Algorithm A.4. Additional details regarding the approach can be found in Dürr et al. (2020).

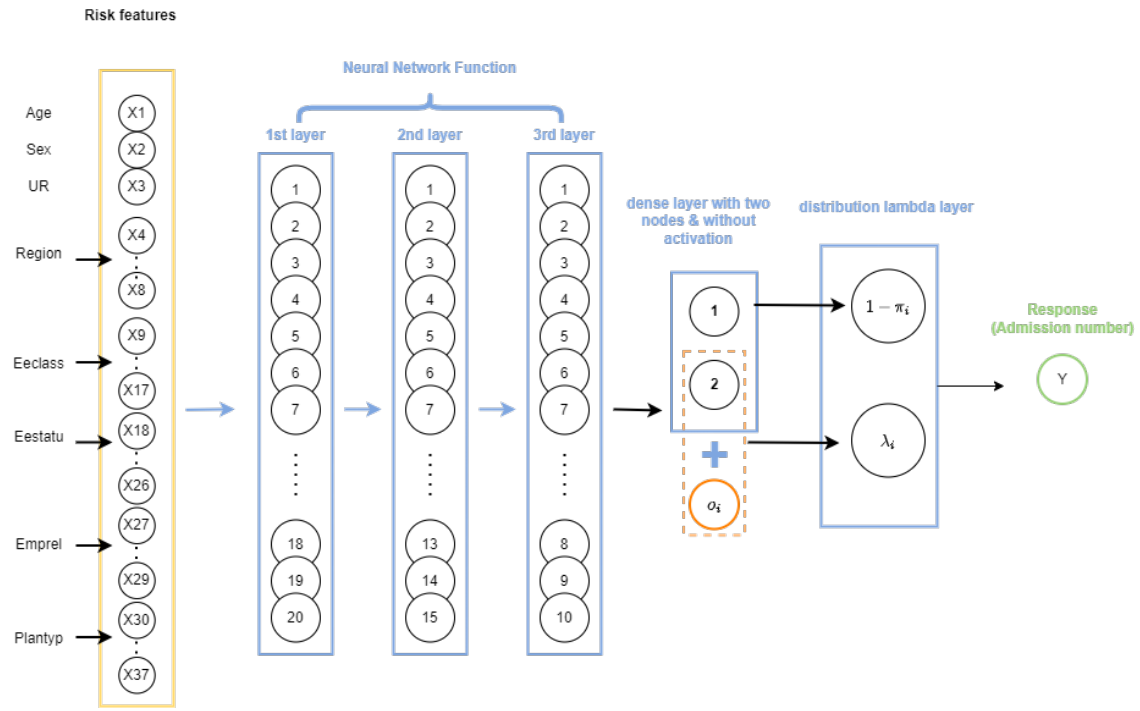


Figure 6.1: An illustration of a sample ZIPNN model with three hidden layers and 20,15,10,2 neurons in each layer.

The structure of the ZIPNN model in terms of the connection between layers, the shape of input and out of each layer and the number of parameters is shown in Listing 6.1. The final layer of the model is the distribution layer, which follows a zero-inflated Poisson distribution. The mean of the distribution is taken as the output or the response, which, in our case, is the admission counts.

```
Model: "ZIPNN model"
__________________________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
==================================================================================
 Design (InputLayer)         [(None, 37)]              0         []
 hidden1 (Dense)             (None, 20)                760       ['Design[0][0]']
 dropout_2 (Dropout)         (None, 20)                0         ['hidden1[0][0]']
 hidden2 (Dense)             (None, 15)                315       ['dropout_2[0][0]']
 dropout_1 (Dropout)         (None, 15)                0         ['hidden2[0][0]']
 hidden3 (Dense)             (None, 10)                160       ['dropout_1[0][0]']
 dropout (Dropout)           (None, 10)                0         ['hidden3[0][0]']
 Net (Dense)                 (None, 2)                 22        ['dropout[0][0]']
 LogVol (InputLayer)         [(None, 1)]               0         []
 concate1 (Concatenate)      (None, 3)                 0         ['Net[0][0]','LogVol[0][0]']
```

```
 distribution_lambda    (None, None)  0       ['concate1[0][0]']
 (DistributionLambda)
==========================================================================
Total params: 1,257
Trainable params: 1,257
Non-trainable params: 0
```

Listing 6.1: Structure of the $ZIPNN$ model.

6.2.3 Zero-inflated Poisson Combined Actuarial Neural Network (ZIPCANN) model

The ZIPCANN model contains additional skip connections, as in the case of a CANN model. In a CANN model (model 4 in Table 6.4), a skip connection connects the input features directly to the output layer. In effect, the predictor of a CANN model contains an additional regression function compared to the predictor of a NN and is of the form

$$(o_i, \boldsymbol{x}_i) \mapsto \log(\mu_i^{CANN}) = o_i + \langle \boldsymbol{\beta}^{reg}, \boldsymbol{x}_i \rangle + \langle \boldsymbol{\beta}^{(d+1)}, (\mathbf{z}^{(d)} \circ \cdots \circ \mathbf{z}^{(1)})(\boldsymbol{x}_i) \rangle, \quad (6.22)$$

with the parameters from the regression function or skip connection represented by $\boldsymbol{\beta}^{reg}$ (Schelldorfer and Wuthrich (2019)). As detailed in Schelldorfer and Wuthrich (2019), various versions of CANNs exist depending on whether the weights in the regression part are updated or not whilst training the model. More specifically, one can either estimate the weights in the skip connection during training, or alternatively, keep the weights of the regression component fixed as the iterated weighted least squares (IWLS) estimates from the corresponding regression model. In this chapter, we adopt the former approach for all models containing skip connections.

For the ZIPCANN model, two skip connections are used here; one for the rate parameter and one for the probability parameter. Instead of the output layer, the skip connection is made to the two neurons in the last hidden layer. In order to be consistent with the ZIP model, only the age variable was used in the skip connection

associated with π, whereas all the features were used in the skip connection for μ. Consequently, the predictor for the ZIPCANN model is of the form

$$E(Y_i) = (1 - \pi_i^{zipcann})\mu_i^{zipcann} \tag{6.23}$$

where $\mu_i^{zipcann}$ is given by

$$\log(\mu_i^{zipcann}) = \langle \boldsymbol{\beta}^{reg}, \boldsymbol{x}_i \rangle + \langle \boldsymbol{\beta}_1^{(d)}, \mathbf{z} \rangle + o_i \tag{6.24}$$

and $\pi_i^{zipcann}$ is given by

$$logit(\pi_i^{zipcann}) = \gamma_{age} x_i^{age} + \langle \boldsymbol{\beta}_2^{(d)}, \mathbf{z} \rangle \tag{6.25}$$

or alternatively,

$$logit(1 - \pi_i^{zipcann}) = \gamma_{age} x_i^{age} + \langle \boldsymbol{\beta}_2^{(d)}, \mathbf{z} \rangle. \tag{6.26}$$

Similarly to the ZIPNN model, the final output layer of the ZIPCANN model is also a distribution layer. The code and the pseudo-code for fitting the ZIPCANN model are given in Listing A.9 and Algorithm A.5, respectively. Also, a diagrammatic representation of a sample model with one-hot encoding, skip connections and 20,15,10,2 neurons in each of the layers (model 6 in Table 6.4) is shown in Figure 6.2. The structure of the ZIPCANN model is given in Listing A.10.

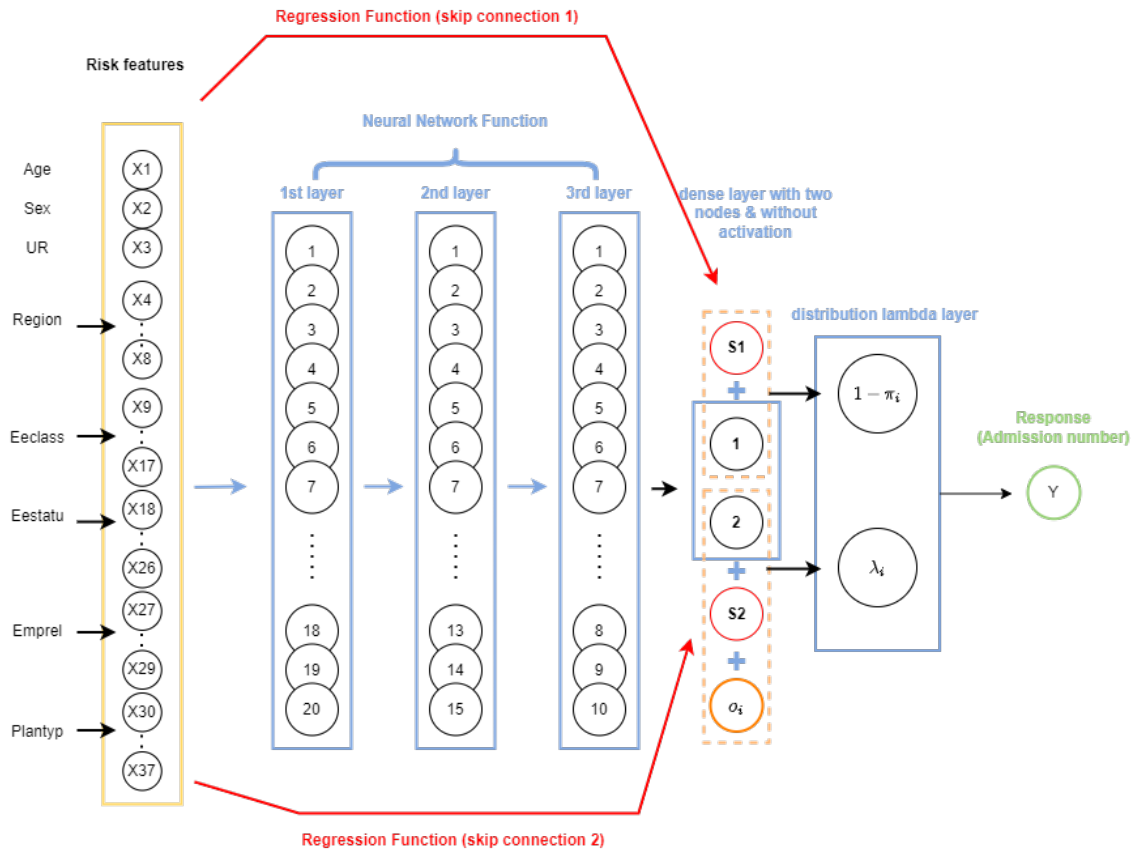


Figure 6.2: An illustration of a sample ZIPCANN model with skip connections, one-hot encoding and 20,15,10,2 neurons in each of the layers.

6.3 Details of Model Fitting and Hyper-parameter Choices

The ZIP regression model was fitted using `zeroinfl()` function in the `pscl` package in RStudio (R Core Team (2021), RStudio Team (2021), and Zeileis et al. (2008b)). The `zeroinfl()` utilises the Nelder and Mead optimisation method for estimating the model coefficients. For the network-based implementation of the ZIP model, we mainly employed the `tfprobability` package (Dillon et al. (2017) and Keydana (2022)) in addition to the `keras` (Allaire and Chollet (2021)) and `tensorflow` (Allaire and Tang (2021)) packages used for constructing NN models. The `tfprobability` package allows to define probability distributions within a

deep-learning model using a distribution layer. The main parts of the code used for building the network-based models are given in Appendix A.7.

The hyper-parameter options identified and adopted in Chapter 5 have also been used here, the details of which are briefed below.

- Gradient descent method (GDM): the Nesterov-accelerated adaptive moment estimation (Nadam) method was used as the choice of the Gradient descent optimisation algorithm for estimating the model weights.
- Network architecture: for NN and CANN models, a network architecture with three hidden layers with (20,15,10) neurons in each layer was considered. As mentioned earlier, the ZIPNN and ZIPCANN models have an additional layer with two neurons. In addition to the layers mentioned above, an attention layer is used to interpret the results from NN models (see section 6.4). Also, for the ZIPNN and ZIPCANN models, alternative architectures with different numbers of neurons in the first three hidden layers were considered. The results, as given in Table A.24, indicate that the choice of (20,15,10) architecture has the best predictive performance compared to other architectures in the case of the ZIPNN model. Moreover, for the ZIPCANN model, the (35,25,20) architecture only minimally outperforms the (20,15,10) architecture. Hence, the combination of (20,15,10) neurons in the initial three hidden layers is adopted for both models.
- Batch size and epochs: a combination of 30,000 batch size and 500 epochs was used for fitting the NN models. The 30,000 batch size was identified as a reasonable choice in the previous chapter, and the large epoch was used since the early stopping approach implemented using the callback method will restrict the model from over-fitting.
- Validation split: a further 80:20 split of the learning set was used as the training set, $\mathcal{D}^{(-)}$, and a validation data set $\mathcal{V}$.
- Loss function: the negative log-likelihood was used as the objective function,

which the GDM algorithm minimises for estimating the model weights. The log-likelihood of ZIP mixture distribution is given by

$$l(\pi, \mu; y) = \sum_{i=1}^{n} u_i \ln\left(\pi_i + (1-\pi_i)\exp(-\mu_i)\right) + (1-u_i)\big[\ln(1-\pi_i) - \mu_i + y_i \ln(\mu_i) - ln(y!)\big], \tag{6.27}$$

where $u_i = I(y_i = 0)$, μ_i is the mean of the Poisson component, π_i the probability parameter and y_i is the response variable. It is worth noting that in the case of the zero-inflated models, the saturated model reduces to the saturated version of the count component (Martin and Hall (2016)). This follows from the fact that a saturated model has the mean responses equivalent to the data, i.e., $E(y_i) = y_i$. For the zero-inflated models, this is true when we set $\mu_i = y_i$ and replace π_i with $u_i = I(y_i = 0)$ since $E(y_i) = (1-\pi_i)\mu_i$. Hence, the saturated model for ZIP mixture distribution will be a saturated Poisson distribution.

As previously mentioned, for the ZIPNN and ZIPCANN models, the output layer is a distribution layer (see Listings 6.1 and A.10) with underlying Zero-inflated Poisson distribution and the mean of the distribution is taken as the model outcome or response which is the admission number. More precisely the the distribution layer is considered as the model output. The NLL is the negative sum of the logarithm of the likelihood assigned to the observed count by the conditional probability distribution ($-\sum \log(P(y|x, w))$). As the model's output is the distribution layer, we can obtain this by defining the loss function as shown in Listing 6.2.

```
#negative log-likelihood loss function
nll<-function(y_true,y_pred)
{
  -y_pred$log_prob(k_reshape(y_true,shape = c(-1)))# k_shape for flattening the
      tensor
  #y_pred is the ZIP distribution layer
  #y_true is the observed admission counts
}
```

Listing 6.2: Negative log-likelihood loss function used for training zero-inflated neural network models.

Additionally, the early stopping and dropout model improvement approaches, as detailed in Chapter 5, were used to avoid over-fitting for all the network-based models.

6.4 Interpreting Network-based Models Using the LocalGLMnet Approach

The underlying principle of the LocalGLMnet approach, as proposed by Richman and Wüthrich (2022), is to use a NN for estimating the model coefficients of a GLM, as indicated by the name local generalized linear model network or LocalGLMnet. In this case, the coefficients $\boldsymbol{\beta}$ are replaced by feature-dependent non-linear functions $\boldsymbol{\beta}(\boldsymbol{x})$. The implementation of this approach involves creating an additional network layer, called the attention layer, with the same dimension as that of the feature space $q = q_0$, containing the so-called regression attention $\boldsymbol{\beta}(\boldsymbol{x})$. The regression attention terminology follows from its similarity to the attention weights proposed by Bahdanau et al. (2014) and Vaswani et al. (2017). The motivation behind the attention weights is to give sufficient weight (attention) to the different features reflecting their significance. Richman and Wüthrich (2022) further extended this to develop LocalGLMnet to derive explanations from the results similar to a GLM. The key assumption behind this is the additive decomposition of the predicted value in terms of the features, as shown in Equation 6.28. In other words, the predicted value could be represented in terms of the weighted sum of features. Thus, in the

case of a LocalGLMnet, the predictor is of the form

$$\begin{aligned} \boldsymbol{x_i} \mapsto g(\mu_i) = g(\mu(\boldsymbol{x_i})) &= \beta_0 + \langle \boldsymbol{\beta(x_i), x_i} \rangle + o_i \\ &= \beta_0 + \big(\beta_1(\boldsymbol{x_i})x_{i1} + ... + \beta_q(\boldsymbol{x_i})x_{iq}\big) + o_i \end{aligned} \tag{6.28}$$

where

$$\boldsymbol{x_i} \mapsto \boldsymbol{\beta(x_i)} = \boldsymbol{z^{(d:1)}(x_i)} = \big(\boldsymbol{z^{(1)} \circ .... \circ z^{(d)}}\big)\boldsymbol{(x_i)} \tag{6.29}$$

with $g()$ being the link function and μ_i being the mean of the distribution underlying the NN regression function. Interpretations are then obtained by considering the covariate contribution $\beta_j(\boldsymbol{x_i})x_{ij}$ associated with each of the features $x_{ij}, j = 1, ..., q_0$, obtained by extracting the component-wise product. The layer thus containing the component-wise product was termed the LocalGLM layer by Richman and Wüthrich (2022). As $\beta_j(\boldsymbol{x_i})$ depends on the features $\boldsymbol{x}_i = (x_{i1}, ..., x_{i,j})$, $\beta_j(\boldsymbol{x_i})x_{ij}$ vary for each of the record and interpretations are derived for each feature x_j by looking at the $\beta_j(\boldsymbol{x_i})x_{ij}$ for $i = 1, ..., n$. Prior to extending the approach to the ZIPNN model, we implemented it for a NN model, as proposed by Richman and Wüthrich (2022). This highlights the merits of the LocalGLMnet approach, as well as its potential to be extended for more complex models. A diagrammatic representation of a sample LocalGLMnet with three hidden layers and (20,15,10) neurons in each of the layers (model 7 in Table 6.4) is shown in Figure 6.3. The pseudo-code and the sample code for implementing it is given in Algorithm A.6 and Listing A.11 respectively..

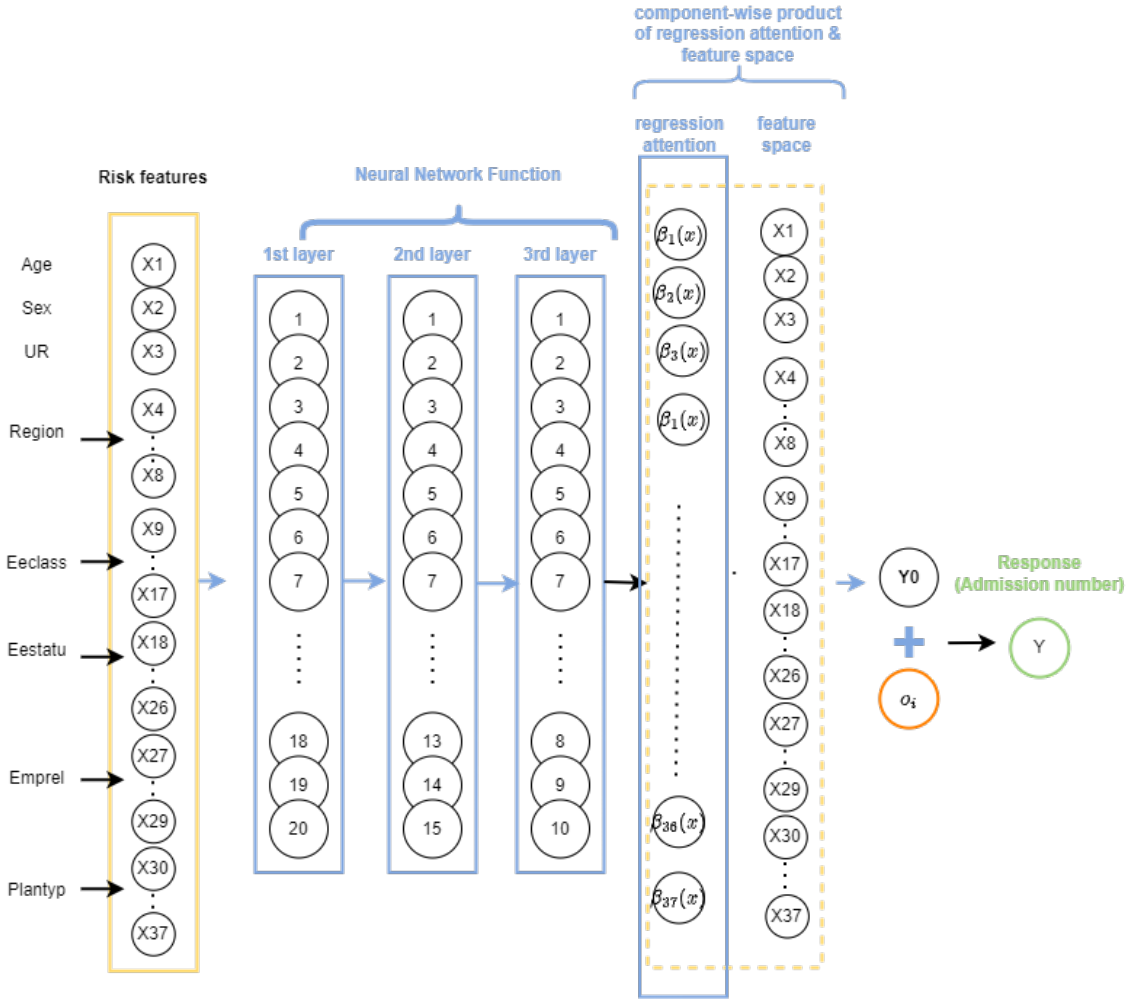
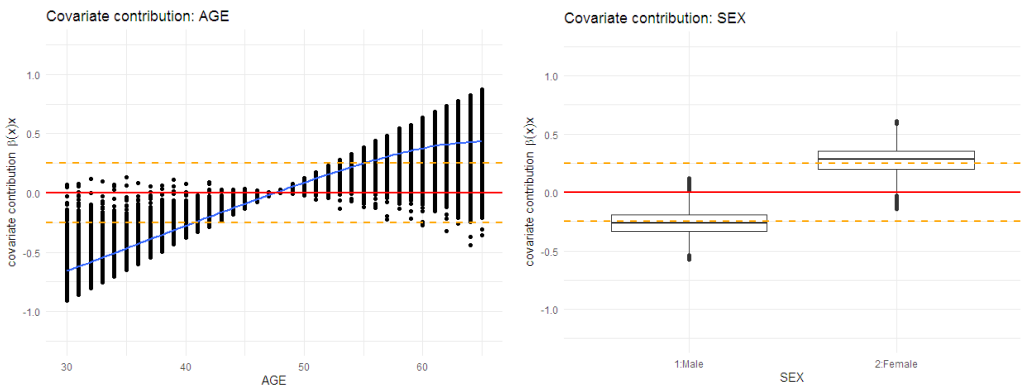


Figure 6.3: An illustration of a sample LocalGLMnet model with LocalGLM layer, one-hot encoding and 20,15,10 neurons in each of the layers.

To demonstrate the process of deriving interpretations we consider, as an example, the covariate contributions for the variables AGE and SEX. These are shown in Figure 6.4, while the corresponding crude admission rates are shown in Figure 6.5. Detailed analysis and interpretations for other covariate contributions is given later using the ZIPNN model.



(a) (b)

Figure 6.4: Graphical representation of covariate contribution from LocalGLMnet model for the testing set (**a**) age variable; (**b**) male, female; the blue line indicates a spline fit approximate curve and the yellow line has been added as a reference line at levels -0.25 and 0.25.

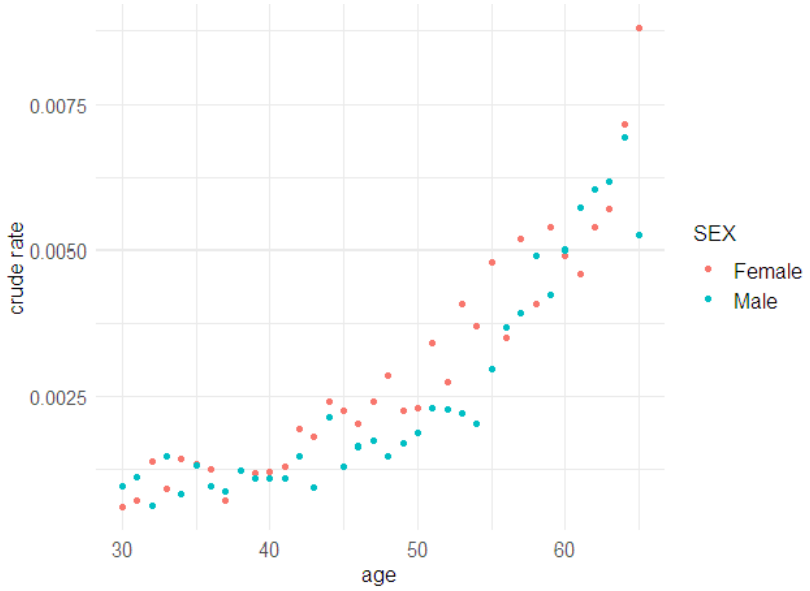


Figure 6.5: Age-wise crude rates of admission due to respiratory diseases for male and female patients for the entire data set.

The crude admission rates clearly show that the rates increase with age for both males and females. Also, the female population generally has higher rates compared to the male population. The same conclusion can be inferred from the covariate contributions. The median value of covariate contributions for females (SEX=2) is higher than for males (SEX=1), indicating higher admission rates. Similarly, the covariate contribution for the AGE variable implies an increasing trend of admission

rates with age, with more variability at younger and older ages.

6.4.1 Interpreting the ZIPNN model

In order to interpret results obtained from ZIPNN model, we extend the additive decomposition assumption to the μ and $p = (1-\pi)$ parameters of the underlying ZIP mixture distribution as shown in Equation 6.3. The regression weights corresponding to each feature are calculated separately for μ and p. Hence the dimension of the layer analogous to the LocalGLM layer with the component-wise product of regression weights and features, is $2 \times q_0$. Among these $2 \times q_0$ weights, q_0 weights are associated with μ and the rest with p.

6.4.1.1 Regression attention for ZIPNN

The ZIPNN model, as defined in Section 6.2.2, consists of an input layer, multiple hidden layers, with the last hidden layer containing two neurons, followed by the distribution layer and the output layer. To create the interpretable ZIPNN model, the attention layer is introduced before the distribution layer, effectively replacing the last hidden layer of the ZIPNN model with two neurons (see Figure 6.6).

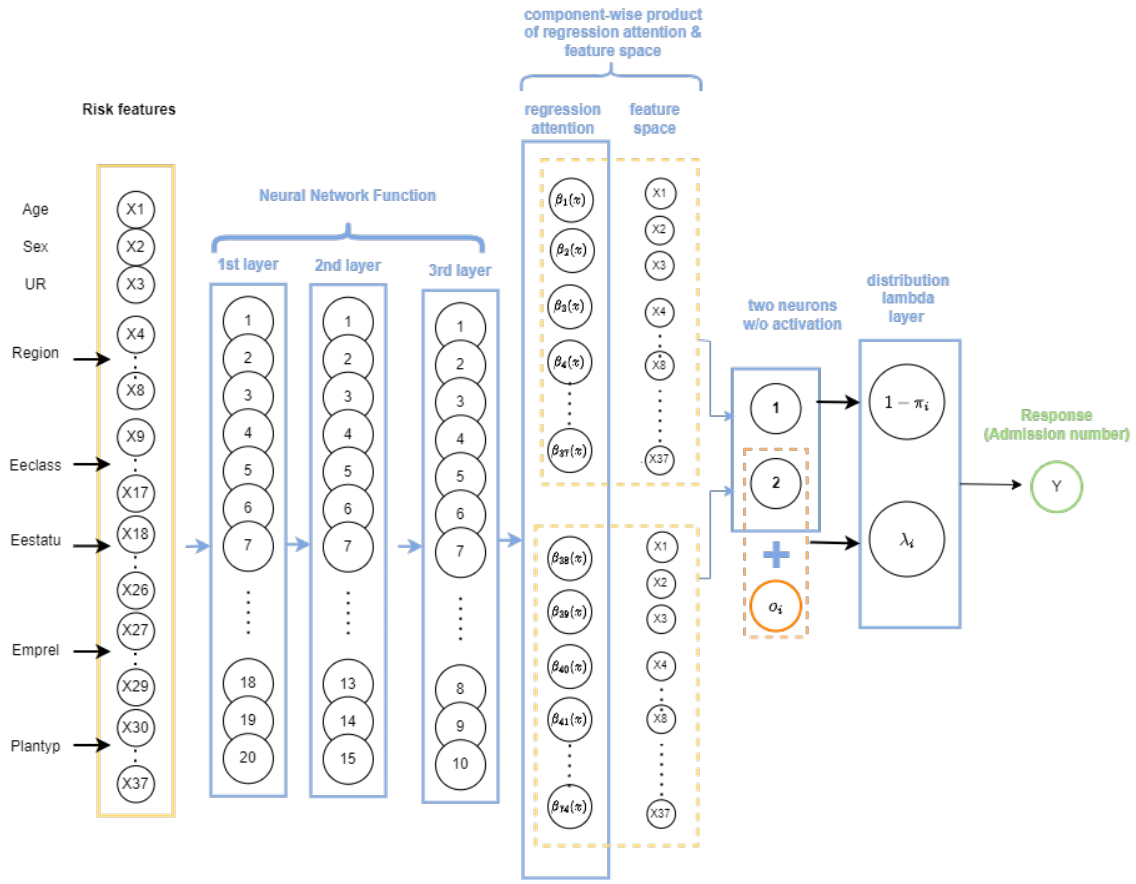


Figure 6.6: An illustration of an interpretable ZIPNN model with regression attention, one-hot encoding and 20,15,10 neurons in each of the layers.

Hence the rate and probability parameters defined in Equations 6.19 and 6.20 are replaced by

$$\log(\mu_i^{zipnn}) = \beta_{0\mu} + \langle \boldsymbol{\beta}_{(\boldsymbol{q_0+1:2q_0})}(\boldsymbol{x}_i), \boldsymbol{x_i} \rangle + o_i \tag{6.30}$$

and

$$logit(p_i^{zipnn}) = \beta_{0\pi} + \langle \boldsymbol{\beta}_{(\boldsymbol{1:q_0})}(\boldsymbol{x}_i), \boldsymbol{x_i} \rangle \tag{6.31}$$

with $\boldsymbol{x_i} \mapsto \boldsymbol{\beta}(\boldsymbol{x_i}) = \boldsymbol{z}^{(\boldsymbol{d:1})}(\boldsymbol{x_i}) = \left(\boldsymbol{z}^{(\boldsymbol{1})} \circ .... \circ \boldsymbol{z}^{(\boldsymbol{d})}\right)(\boldsymbol{x_i})$. The attention weights $\boldsymbol{\beta}_{(\boldsymbol{1:q_0})}(\boldsymbol{x})$ and $\boldsymbol{\beta}_{(\boldsymbol{q_0+1:2q_0})}(\boldsymbol{x})$ are associated with parameters p and μ respectively. The interpretations or the impact of the different features on parameters μ and p are derived by analysing the covariate contributions: $\beta_j(\boldsymbol{x_i})x_{ij}$ for p_i and $\beta_{q_0+j}(\boldsymbol{x_i})x_{ij}$ for μ_i, where $j = 1, ..., q_0$. Sample code for creating a ZIPNN model (model 8 in

Table 6.4) is given in Listing A.12. The structure of the interpretable ZIPNN model is given in Listing A.13.

Although it is possible to create interpretable versions of both CANN and ZIPCANN models similar to those for the NN and ZIPNN models, the introduction of the attention layer nullifies one of the primary purposes of skip connections, which is to distinguish the main effects from the complex effects. This is because the attention layer will integrate the main effect represented by the skip connection and with the complex effect from the network. Hence, we do not discuss interpretable versions of the CANN and ZIPCANN models. A brief description is provided in Section A.7.1.

6.5 Comparison of Models

In this section, we compare the different models discussed so far. The models are compared in terms of predictive performance and the insights obtained from the model results.

6.5.1 Predictive performance

The NLL was used to compare the predictive performance of the implemented models. While other comparison measures are available, such as the mean absolute error and the mean square error, these focus on the deviation of the predicted value from the actual observed value, and do not fully account for the considered probability distribution when using a probabilistic model. Also, other alternative measures, such as the predicted probability of a zero count, fall short of providing a comprehensive comparison as they only consider the probability of admission as a dichotomous event, and fail to consider the admission rate. The NLL of a model has the form:

$$\mathcal{L}_{\mathcal{A}}(\boldsymbol{\beta}) = -\log\left(L_f\right) \tag{6.32}$$

where $\mathcal{A}$ represents the data set and L_f gives the likelihood of the fitted model (e.g. equation 5.15 for the log-likelihood of a ZIP mixture distribution). Performance was

compared based on the NLL $\mathcal{L}_{\mathcal{T}}$ of different models for the testing set $\mathcal{T}$. Values of the NLL for both the testing and training data sets, for all considered models, are shown in Table 6.4. Table 6.4 also contains the average fitted mean $\hat{\mu}$ for each model for the full data set. The difference between the average fitted mean of a model and that of the empirical mean of observed data indicates population-level bias.

Model no.	Model	Learning loss	Testing loss	Average fitted mean
Model 1	Pois.reg	28524.2	3016.8	0.0027
Model 2	ZIP.reg	26662.3	2822.2	0.0028
w/o attention layer				
Model 3	NN(20,15,10)	28202.3	3010.5	0.0026
Model 4	CANN(20,15,10)	28220.6	3005.1	0.0025
Model 5	ZIPNN(20,15,10)	26364.5	2808.3	0.0027
Model 6	ZIPCANN(20,15,10)	26399.7	2809.4	0.0027
w/ attention layer				
Model 7	NN(20,15,10)	28217.9	3000.2	0.0024
Model 8	ZIPNN(20,15,10)	26424.2	**2807.8**	0.0026

Table 6.4: Predictive performance of Poisson and ZIP regression models and network-based models with and without an additional layer for interpretation based on NLL. The empirical mean of the observed data is 0.0027.

The results indicate that the ZIPNN model has the lowest testing loss compared to other models. In general, the models with an underlying zero-inflated mixture distribution assumption performed better than those with a basic Poisson distribution assumption. The ZIPNN and ZIPCANN models performed better than the traditional ZIP regression model. Similarly, the NN and CANN models performed better than the conventional Poisson regression model. As the difference between the testing loss of ZIPNN and ZIPCANN models is small, it is not possible to conclusively state the supremacy of one over the other due to the inherent randomness in NN results. The different attributes leading to the randomness in NN model results

have been extensively discussed by Richman and Wüthrich (2020). Approaches such as nagging predictor and k-fold validation could be adopted to address this randomness, as done in previous chapter. More than the inherent randomness, the underlying architecture sways the model performance on a larger scale. Hence we analysed this impact by considering different architectures of varying complexity. As shown in the Table A.24, the combination of 20,15,10 neurons in the first three hidden layers is a suitable choice for both ZIPNN and ZIPCANN models.

6.5.2 Model interpretability

In order to compare the interpretations derived from the model results, we consider the coefficient estimates from the ZIP regression model and the covariate contributions from the ZIPNN model. The coefficient estimates ($\boldsymbol{\beta_{reg}}, \boldsymbol{\gamma}$) from the ZIP regression model are given in Table 6.5. For ease of interpretion of the results, a sum-to-zero constraint was applied to the coefficient estimates of different levels of all the non-binary categorical variables.

The interpretation of coefficient estimates from a ZIP distributional assumption is more complex compared to a conventional regression model, because it refers to a mixture distribution. For the ZIP regression model considered here, the count component considers all features, whereas the zero component only considers the age variable. Additionally, in the case of the ZIPNN model, an attention layer is used for both λ and p parameters, which means that the covariate contributions of different variables for λ need to be considered in conjunction with those for the p parameter. This makes interpreting the results and comparison with ZIP regression model less straightforward. The comparison was carried out between the most significant coefficient estimates in the ZIP regression model and the corresponding covariate contribution using the ZIPNN model.

Coefficient	Estimate	Std. Error	z value	Pr(> |z|)	Signi.
Count component					
$\beta_{intercept}$	-1.7956	0.3522	-5.0990	0.0000	***
β_{age}	0.0268	0.0054	4.9380	0.0000	***
$\beta_{region_1-northeast}$	-0.2113	0.1084	-1.9490	0.0513	.
$\beta_{region_2-northcentral}$	-0.0911	0.1070	-0.8520	0.3945	
$\beta_{region_3-south}$	-0.0959	0.1047	-0.9160	0.3598	
β_{region_4-west}	-0.5460	0.1109	-4.9230	0.0000	***
$\beta_{region_5-unknown}$	0.9444	0.4050	2.3320	0.0197	*
$\beta_{sex-female}$	0.1022	0.0363	2.8170	0.0048	**
$\beta_{ur-urban}$	-0.1491	0.0502	-2.9720	0.0030	**
$\beta_{eeclass_1-salary\ non-union}$	-0.2894	0.0530	-5.4610	0.0000	***
$\beta_{eeclas_2-salary\ union}$	-0.0385	0.1510	-0.2550	0.7985	
$\beta_{eeclass_3-salary\ other}$	-0.2526	0.0910	-2.7760	0.0055	**
$\beta_{eeclass_4-hourly\ non-union}$	0.1980	0.0537	3.6870	0.0002	***
$\beta_{eeclass_5-hourly\ union}$	0.2905	0.0579	5.0200	0.0000	***
$\beta_{eeclass_6-hourly\ other}$	0.2276	0.0875	2.6010	0.0093	**
$\beta_{eeclass_7-non-union}$	0.1227	0.0535	2.2920	0.0219	*
$\beta_{eeclass_8-union}$	-0.0624	0.0951	-0.6560	0.5117	
$\beta_{eeclass_9-unknown}$	-0.1958	0.0513	-3.8200	0.0001	***
$\beta_{eestatu_1-active\ full\ time}$	-0.3383	0.0650	-5.2010	0.0000	***
$\beta_{eestatu_2-active\ part\ time\ or\ seasonal}$	-1.2290	0.2237	-5.4950	0.0000	***
$\beta_{eestatu_3-early\ retiree}$	-0.1872	0.0777	-2.4100	0.0160	*
$\beta_{eestatu_4-medicare\ eligible\ retiree}$	-0.1774	0.1340	-1.3230	0.1857	
$\beta_{eestatu_5-retiree\ (status\ unknown)}$	-0.2135	0.2855	-0.7480	0.4545	
$\beta_{eestatu_6-COBRA\ continuee}$	0.3664	0.1607	2.2810	0.0226	*
$\beta_{eestatu_7-long\ term\ disability}$	1.0403	0.1598	6.5120	0.0000	***
$\beta_{eestatu_8-surviving\ spouse/depend.}$	0.5113	0.1943	2.6310	0.0085	**
$\beta_{eestatu_9-unknown}$	0.2274	0.0771	2.9510	0.0032	**
$\beta_{emprel_1-employee}$	-0.5073	0.0831	-6.1010	0.0000	***
$\beta_{emprel_2-spouse}$	-0.2611	0.0836	-3.1240	0.0018	**
$\beta_{emprel_3-child/other}$	0.7684	0.1622	4.7360	0.0000	***
$\beta_{plantyp_2-comprehensive\ plan}$	0.0854	0.0898	0.9500	0.3420	
$\beta_{plantyp_3-EPO\ plan}$	0.0834	0.1741	0.4790	0.6318	
$\beta_{plantyp_4-HMO\ plan}$	0.0846	0.0595	1.4220	0.1550	
$\beta_{plantyp_5-Non-Cap\ PoS\ plan}$	-0.1646	0.0725	-2.2680	0.0233	*
$\beta_{plantyp_6-PPO\ plan}$	0.0898	0.0423	2.1240	0.0336	*
$\beta_{plantyp_7-Cap\ or\ Part\ Cap\ PoS\ plan}$	0.1335	0.1284	1.0390	0.2986	
$\beta_{plantyp_8-CDHP}$	0.0243	0.0636	0.3830	0.7021	
$\beta_{plantyp_9-HDHP}$	-0.3364	0.0786	-4.2820	0.0000	***
Zero component					
$\gamma_{intercept}$	6.1238	0.2899	21.1240	< 2e-16	***
γ_{age}	-0.0314	0.0052	-6.0520	0.0000	***

Table 6.5: Coefficient estimates based on the ZIP regression model with the significance codes (‘***’,‘**’,‘*’, ‘.’,‘ ’) indicating the level of significance of the estimates within the intervals (0, 0.001], (0.001, 0.01], (0.01, 0.05], (0.05, 0.1], and (0.1, 1].

The plots showing the covariate contributions were produced using the test data. Figures 6.7 and 6.7 (continued) show the covariate contributions for different variables associated with the rate (λ) parameter, while Figures 6.8 and 6.8 (continued) show the covariate contributions for the p parameter. The horizontal yellow line in the plots was added as a reference line at levels -0.25 and 0.25. This can help to compare the magnitude of the covariate contributions and how they are associated with the parameters.

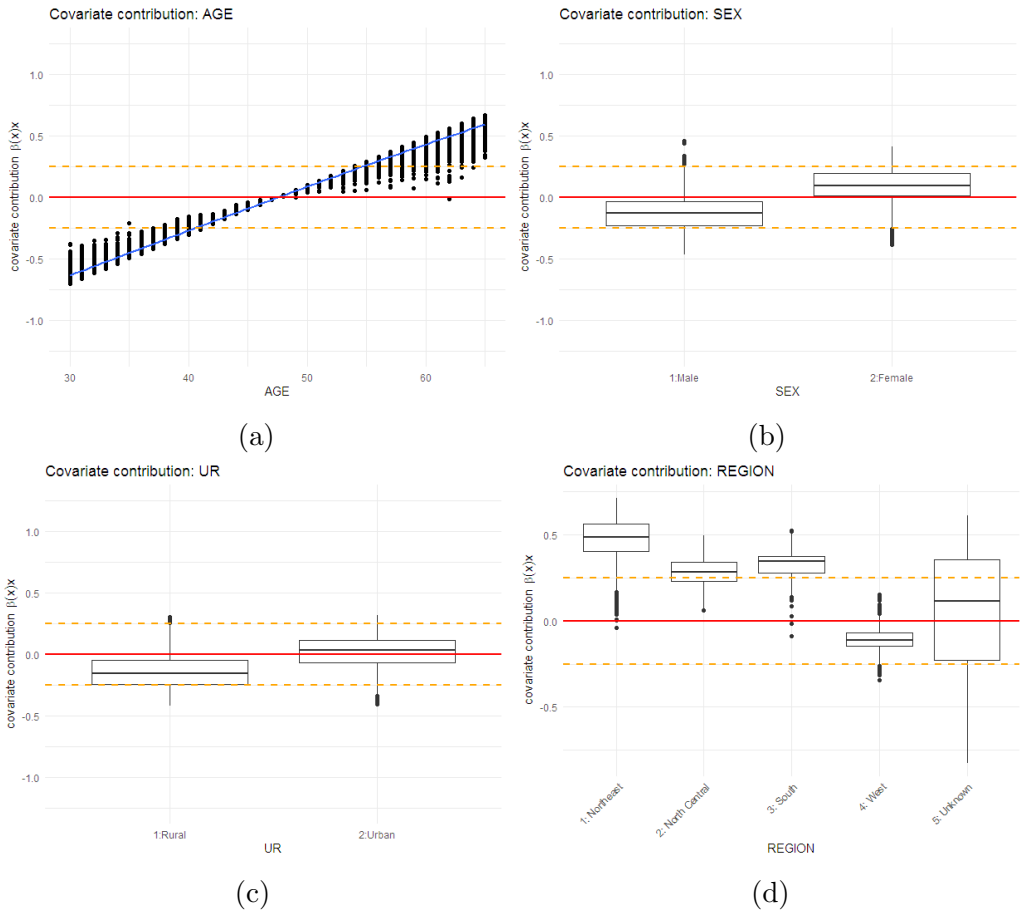


Figure 6.7: Graphical representation of covariate contributions for parameter λ in the ZIPNN model: **(a)** AGE; **(b)** SEX; **(c)** UR ; **(d)** REGION; **(e)** EECLASS; **(f)** EESTATU.

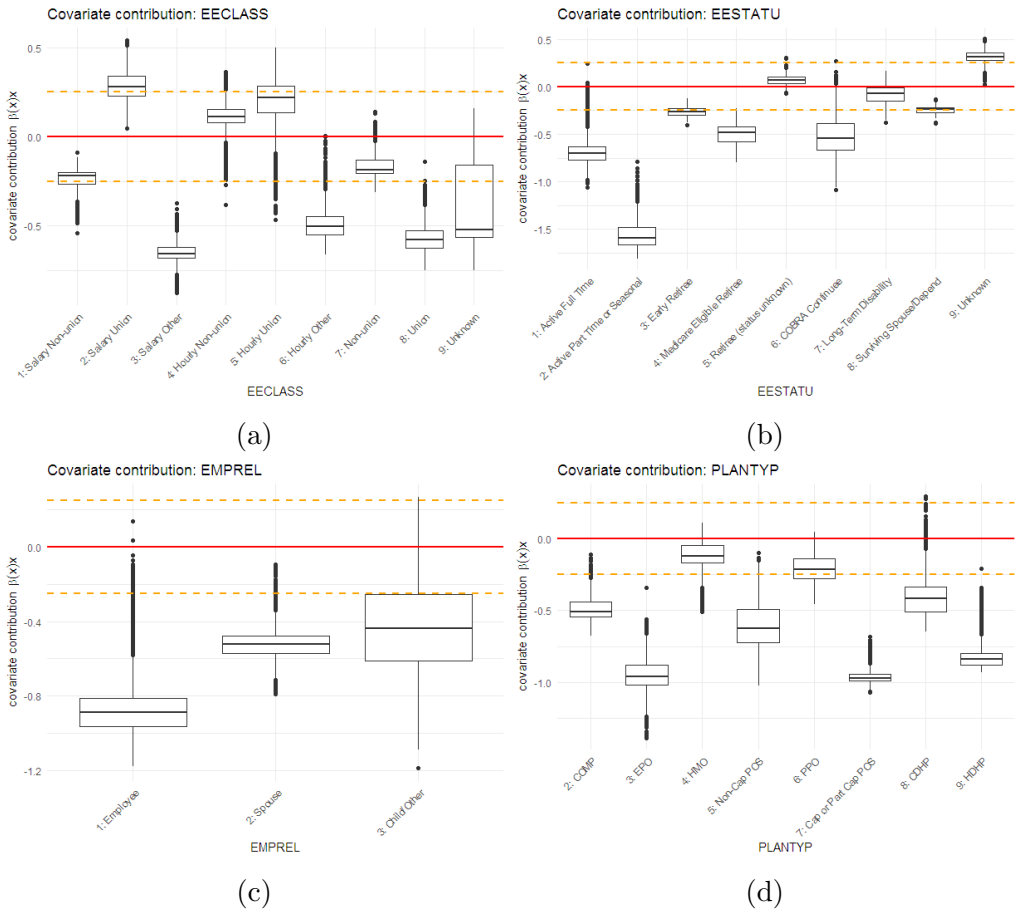


Figure 6.7 (continued): Graphical representation of covariate contributions for parameter λ in the ZIPNN model: **(a)** EECLASS; **(b)** EESTATU; **(c)** EMPREL; **(d)** PLANTYP.

The analysis of the covariate contributions using the ZIPNN model for the AGE variable for both the probability and rate parameters (Figures 6.7 and 6.8) indicates an increasing trend over age. This suggests a positive relationship showing that as age increases, both the probability of having an admission and the rate of admission increase. The spline fit (blue line) indicates an almost linear trend, with the greatest variability at younger and older ages. In both cases, this suggests that the impact of the age variable on the parameters varies at different ages. The variation appears to be lowest around the age of 47 before increasing for older ages as well. The highly significant positive coefficient estimate for the AGE variable in the count component of the ZIP regression model (see Table 6.5) indicates that the rate of admission increases with age. The negative estimate in the zero component associated with the π parameter also suggests that the probability of having excess zero decreases

with age. In other words, the probability of having non-zero admissions increases with age.

The covariate contributions under the ZIPNN model for the SEX variable indicate a higher probability and rate of admission for females (SEX=2) compared to males (Figures 6.7 and 6.8). The significant positive coefficient estimate in the count component of the ZIP regression model also implies the same. This is in line with the pattern displayed by the crude rates (see Figure 6.5). The low variability in both cases suggests that gender has a consistent and explicit impact on admissions related to respiratory diseases.

Regarding the UR variable, the ZIP regression model suggests a lower admission rate for individuals in the urban area (UR=2), based on its significant negative coefficient estimate (Table 6.5). This contradicts the pattern observed under the ZIPNN model. Although not considerably different, the covariate contributions suggest that people in urban areas are more prone to respiratory disease-related admissions (Figures 6.7 and 6.8). This difference could be arising due to the fact that the network architecture captures complex interactions between different features, which is accounted for while estimating the nonlinear regression weights, $\boldsymbol{\beta(x)}$. For instance, there may exist an interaction between the age and UR variables, due to the possibility that retired or older individuals are more likely to reside in rural areas.

The coefficient estimates based on the ZIP regression model for the REGION variable ($\beta_{region_z}, z = 1, .., 5$), given in Table 6.5, indicate a higher rate of admission when the REGION variable is unknown (REGION=5). However, the estimate is not significant. The high variability of covariate contribution under the ZIPNN model for level 5 of the region variable is also pointing towards this uncertainty (see Figure 6.7). On the other hand, the 'west' region (REGION=4) has a significant negative coefficient estimate under the ZIP regression model, implying a lower admission rate for individuals from the western region. The covariate contribution in the ZIPNN model corresponding to the λ parameter for the REGION variable also

suggests the same (Figure 6.7).

In the case of the EECLASS variable, a significantly higher rate of admission is inferred from the coefficient estimates of the ZIP regression model for 'hourly non-union' and 'hourly union' (levels 4 and 5) and significantly low rates for individuals with unknown (level 9) employee classification (Table 6.5). A comparable trend is observed when the covariate contributions for both rate and probability parameters under the ZIPNN models are considered together, which is essential as interpreting the results based on a single parameter could be unclear. For example, the covariate contributions for the 'salary union' level (level 2) of the EECLASS variable related to the rate parameter (see Figure 6.7 (continued)) suggest that the admission rate is high for individuals belonging to that employee classification cohort. However, the covariate contribution associated with the probability parameter (Figure 6.8) indicates that the probability of having an admission is very low for individuals in that group. In contrast, for individuals with unknown employee classification, the covariate contributions not only indicate a lower probability of admission, but also imply a lower admission rate.

Similarly, for the EESTATU variable, the combined analysis of covariate contributions of both parameters (Figures 6.7 (continued) and 6.8) points towards a higher propensity for admission related to respiratory diseases for individuals with employment status categorised as 'long term disability' (level 7), whereas it is low for individuals with employment status 'Active Part Time or Seasonal' (level 2). The same pattern is inferred from the associated coefficient estimates in the ZIP regression model. It is reasonable to conclude that the higher admission rates for people with long-term disability are as anticipated. Drawing further inferences regarding the relationship between employment details of individuals and hospital admission for respiratory diseases is difficult in the current context of admission data. The data provider has compiled the data set by assigning the employment-related information of the primary beneficiary/employee to their dependents as well. In other words, the employment information of those individuals classified as dependents (EMPREL as

'spouse' (level =2) or 'child/other' (level=3)) in the data set does not accurately represent their actual employment status. For instance, an unemployed dependent of an employee/primary beneficiary will also have the same employment-related information as the employee. Due to these inaccuracies within the data set, it becomes challenging to draw conclusive associations between employment information and the rate of admission due to respiratory diseases.

For the EMPREL variable, the pattern is slightly more evident. The coefficient estimates based on the ZIP regression model indicate a significant difference in the admission rates for different levels of EMPREL covariate (Table 6.5). The individuals with EMPREL as 'child/other' (level=3) have the highest rate of admission, followed by 'spouse'(level =2) and 'employee'(level=1). Although the same pattern is observed in the covariate contribution related to the λ parameter (Figure 6.7 (continued)), the covariate contribution associated with p (Figure 6.8 (continued)) indicates a lower probability of admission for individuals with EMPREL categorised as 'spouse' compared to 'employee'.

The coefficient estimates for the different levels of the PLANTYP variable in the ZIP regression model indicate that the admission rates are significantly lower for the High-Deductible Health Plan (HDHP) (level=9) than for other types of plans. This trend is also supported by the ZIPNN model. The covariate contributions not only indicate a lower probability of admission (Figure 6.8 (continued)), but also suggest lower admission rates (Figure 6.7 (continued)) for individuals under the 'HDHP' plan type. This suggests that individuals who are covered under HDHP plans may have a lower likelihood of getting hospitalised for respiratory diseases, and if they do, it occurs at a lower rate. This might be due to the fact that the HDHP plan incentivises individuals to be more proactive in managing their health and seek care only when essential or have more restrictive coverage.

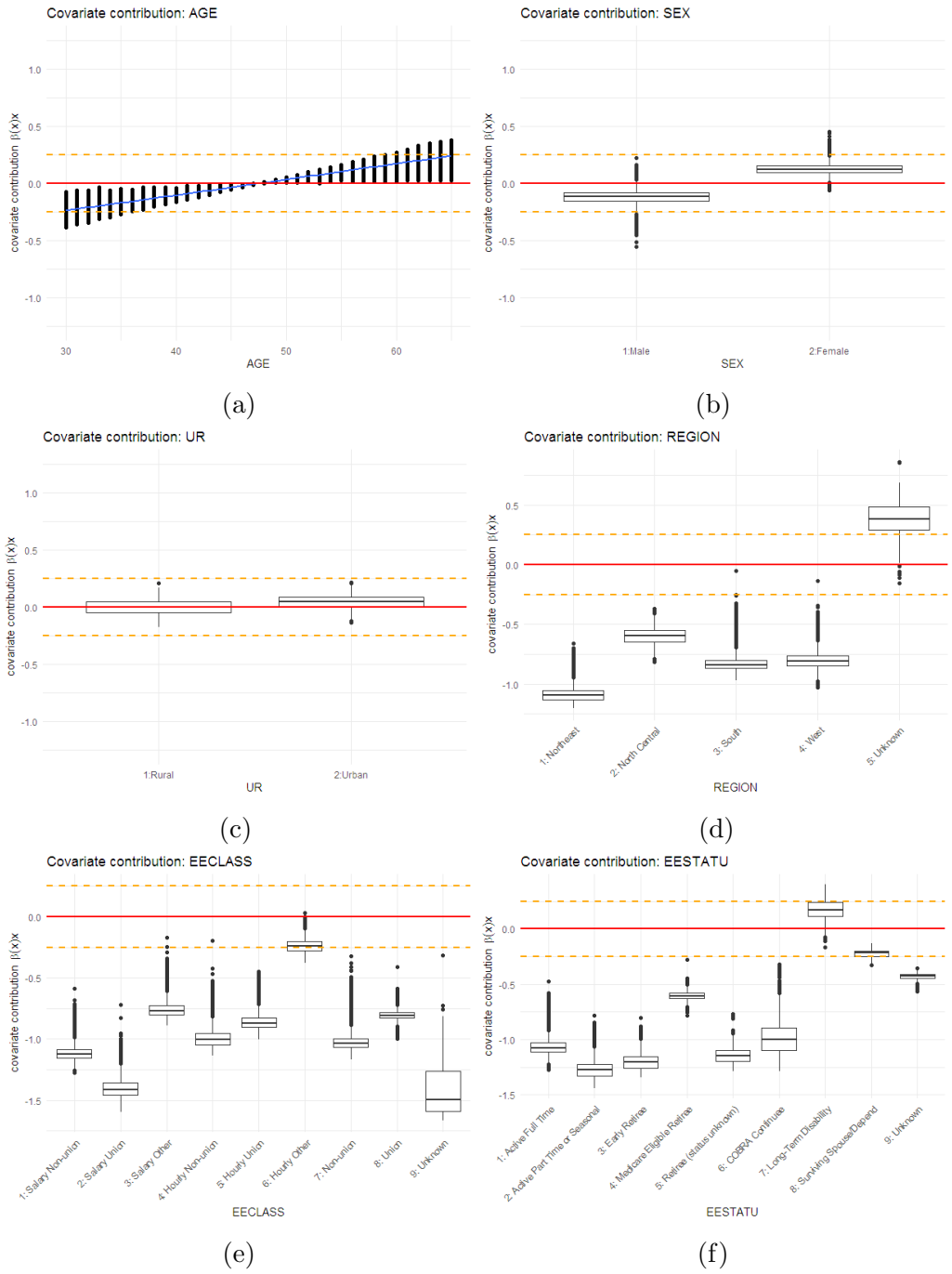
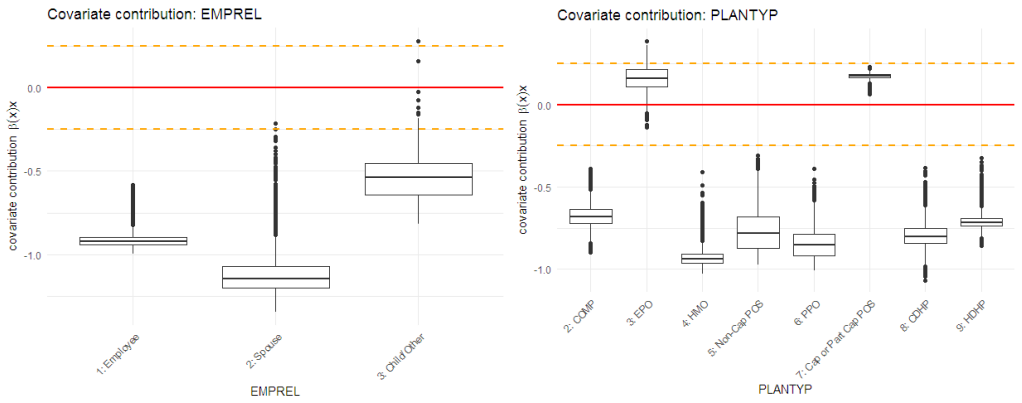


Figure 6.8: Graphical representation of covariate contributions for parameter p in the ZIPNN model: **(a)** AGE; **(b)** SEX; **(c)** UR ; **(d)** REGION; **(e)** EECLASS; **(f)** EESTATU.



(a) (b)

Figure 6.8 (continued): Graphical representation of covariate contributions for parameter p in the ZIPNN model: **(a)** EMPREL; **(b)** PLANTYP.

6.6 Chapter Summary

In summary, in this chapter, we developed the ZIPNN and ZIPCANN models for modelling admission rates related to health insurance. We also employed the LocalGLMnet approach to interpret the findings under the ZIPNN model and compared them with those obtained from the ZIP regression model. The comparisons demonstrated the capability of NN-based models to capture potential non-linear interactions among risk factors for both the probability and rate parameters of the mixture model. Furthermore, from a practical standpoint, the separate attention layers that we included in the NN structure for the two model components, enabled us to analyse the impact of important risk factors on both the probability and rate of admissions.

Furthermore, the underlying zero-inflated mixture distribution allows the ZIPNN and ZIPCANN models to accommodate the excess zero nature of the data, while also benefiting from improved predictive performance offered by NN methodologies. Additionally, the attention layer approach facilitates deriving interpretations from the model, allowing for an in-depth understanding of the various risk attributes on admissions related to respiratory diseases.

Chapter 7

Conclusions and Future work

7.1 Summary

As part of the work discussed in the previous chapters of the thesis, we have developed a suite of statistical and network-based predictive models for medical morbidity risks, with a particular focus on the accurate prediction of rates of admission to hospitals and other healthcare facilities due to critical illnesses such as neoplasms and chronic respiratory diseases. This was achieved by incorporating various statistical methodologies, including conventional regression models (Chapter 4) and their Bayesian adaptations, as well as deep learning techniques (Chapters 5 and 6). Furthermore, the approaches were extended to develop novel models with enhanced and more appealing characteristics and capabilities.

Given the varying levels of complexity within the statistical and NN-based models, the efficiency of the proposed models was assessed and compared in terms of their predictive and fitting performance while duly considering ease of implementation and complexity. Additionally, for the NN-based models, we employed interpretability tools and compared the findings to those resulting from the conventional models. This revealed the strengths and weaknesses of different methodologies, making it challenging to ascertain an all-encompassing superior model. In particular, improved performance in one attribute came at the expense of a decline in another desirable

quality. For example, generally, an increase in the computational complexity of the model resulted in better predictive performance while compromising on the ease of implementation.

The aforementioned pattern was observed while comparing GLM-type models with neural network-based models. The results, in general, indicate that the suite of neural network-based models benefits from superior predictive performance compared to the generally simpler conventional regression models and their Bayesian adaptations (see Section 5.10) but is considerably complex. At the same time, it is also worth noting the extent of the variation in the ease of implementation between each approach. While fitting a GLM model is straightforward, fitting network models, especially in terms of identifying the optimal hyper-parameters and architecture, is demanding.

A potential reason for the improved predictive performance of the network-based models may be the ability to capture possible non-linear interactions between the different features. Although, in principle, these interactions could also be captured in GLM-type models, they must be identified and specified explicitly within the model. Adapting traditional approaches based on methods such as step-wise Akaike information criterion (AIC) or Bayesian information criterion (BIC) for identifying relevant interactions can be tedious and time-consuming for complex and large data sets, as in this case. More specifically, k-fold cross-validation indicated that under both Poisson and negative binomial distributional assumptions, the NN models gave better predictive performances, as determined by the testing data loss, than the GLM-type and CANN models (see Sections 5.6 and 5.10). The better performances of the NN models compared to the CANN models could be partly due to the latter not involving training processes for the regression parts of the models. When the regression component is also trained while fitting the model, the performance of the CANN model seems to be improving. On the other hand, the alternative variant of the CANN model (regression part not trained), although having a lower predictive performance, appears to be less sensitive to the choice of hyper-parameters(see

Section 5.10). Nevertheless, owing to the inherent randomness within the neural network results, it is difficult to conclusively establish the predictive superiority of one over the other. Additionally, the analysis carried out in Sections 5.7 and 5.10 showcased the sensitivity of network models to the data structure, composition and granularity. Due to data-driven nature of the neural network methodologies, any variation in the makeup of the underlying data set affected the model performance and the hyper-parameter tuning had to be carried out again.

The predictive performance of the NN models improves when additional model improvement approaches, such as early stopping and dropout, are incorporated into the model. Additionally, despite being computationally intensive, the nagging predictor approach could be employed to tackle the inherent randomness in neural network results. Moreover, adopting suitable bias regularisation approaches aids in mitigating population-level bias in the NN model results. Furthermore, the lack of interpretability associated with the NN models was addressed with the help of the LocalGLMnet approach, which facilitated obtaining interpretations in the same manner as from traditional regression models.

A drawback of the NN-based models lies in the lack of clarity regarding the significance of parameters and the absence of a statistically sound approach for model selection. Specifically, when comparing NN models, we primarily rely on testing loss, which is often very close among NN-based models. The absence of a significance testing approach concerning the difference in testing loss makes it challenging to choose between such models based on their predictive performance.

Among the Bayesian models, the BP-G model exhibited the best fit compared to the corresponding BP and BP-LN models in terms of the DIC values. The pattern was consistently observed in all contexts, except among the models pertaining to neoplasms-related admissions in males. In that particular case, the BP-LN model demonstrated a slightly superior fit compared to the BP-G model. As in the case of the NN models, where the choices made during the model-fitting process impact the performance, for the Bayesian models, care needs to be exercised to ensure

convergence and that the choice of burn-in period and post-burn-in sample size is adequate.

In Chapter 6, models with underlying zero-inflated mixture distribution assumptions, such as the ZIP regression, ZIPNN and ZIPCANN models, were developed. They showcased better predictive performance than those with a basic Poisson distribution assumption. This improvement can be directly attributed to the excess zero nature of the underlying dataset. The network-based models, such as the ZIPNN and ZIPCANN models, outperformed the traditional ZIP regression model, once again highlighting the capability of NN structure to capture potential non-linear interactions among risk factors. Furthermore, the interpretable ZIPNN model, which contains attention layers, enabled us to analyse the impact of important risk factors on both the probability and rate of admissions.

In summary, while aspects such as ease of implementation and interpretability make GLM-type models appealing, the improved predictive performance and the capability of incorporating prior beliefs advocate for the Bayesian models. Although hyperparameter sensitivity, its data-centric design, and a lack of established statistical-based variable selection processes, such as the likelihood-based measures in the case of regression models, are some of the disadvantages of NN models, their capability to handle complex datasets along with superior predictive performance is what makes them highly appealing. Furthermore, advancements in computing capabilities are paving the way for tackling some of the aforementioned issues.

Regardless of the variations in the model performances, all models indicated the impact of various socio-demographic and employment-related factors on the admissions related to critical illnesses such as neoplasms and chronic respiratory diseases. As discussed in the earlier chapters, variations in morbidity trends concerning specific risk attributes were clearly observable. While some patterns, for instance, the higher prevalence of critical illness in older ages, were anticipated, a more in-depth understanding obtained through precise quantification of the effect would assist in devising more targeted intervention programs. However, it is crucial to keep in mind

that potentially more crucial risk drivers of these illnesses may exist that we did not have access to beyond the factors we considered, which require further investigation and consideration while devising such intervention programs. Even though this was not the primary aim of our work, it's important to be mindful of these factors while undertaking studies aimed at the development of such programs.

Additionally, it is worth noting some of the limitations associated with the work discussed in the thesis. Firstly, we highlight the assumption regarding the data that it is a good representative sample of the insured population in the US. More precisely, we presume that the range of employers and health plans from which the data provider sourced the information is a sufficiently diverse mix from all over the country, and the data is, indeed, representative of the above-mentioned cohort. Secondly, it is crucial to keep in mind that the morbidity trends identified are that of the insured population and do not pertain to the general population, as morbidity trends generally tend to vary among insured and non-insured populations.

Thirdly, we acknowledge that the aspect of comorbidity needs to be taken into consideration. We examined neoplasms and CRDs in isolation and did not sufficiently ponder the relationship between them. In general, it is highly likely that there is a connection between the prevalence of terminal illnesses. More specifically, it is plausible that a person suffering from CRD is at a higher risk of developing respiratory cancer than a healthy individual.

Finally, there is a need to accurately quantify the financial advantages associated with enhanced predictive performance. While the ability of network architecture to capture interactions among different risk attributes is highly valuable for devising targeted interventions, the explicit benefit obtained from the improved prediction in the context of insurance rate-setting remains somewhat unclear. An analysis that examines and quantifies the aforementioned monetary benefit would be one of the intriguing lines of research alongside those discussed in the next section.

7.2 Future work

The comprehensive nature of the developed models allows them to be extended with ease for modelling admission rates for other diseases or other data in similar contexts and applications. A potentially interesting line of further research would be to consider bivariate or multivariate versions of the developed models, aiming to address rates related to comorbidities. More specifically, we could adapt techniques such as trivariate reduction (Kocherlakota and Kocherlakota (2001)) and copulas (Cameron et al. (2004)) and extend them using NN-based approaches similar to those developed in this thesis for the bivariate or multivariate case. This extension would allow us to simultaneously capture the correlation between two or more response variables while taking into account potential nonlinear interactions between the explanatory variables in the model equations. Alternatively, we could employ the approach proposed by Jeong et al. (2023), which involves pairing the random effects of a mixed Poisson model for maintaining identifiability in higher dimensions while boosting forecasting performance using deep learning models. These models could extensively aid in devising healthcare intervention programs and investigating admissions related to other diseases. They could also be appropriately extended and adapted for other rate-setting problems within the insurance sector.

Moreover, it is worth noting that other zero-inflated models similar to those discussed in Chapter 6, such as a zero-inflated negative binomial model, can be implemented under the proposed framework, while extensions to bivariate and/or multivariate versions of such zero-inflated models for dealing with different types of claims are possible. Also, Bayesian neural network approaches as discussed in Wang et al. (2023) and Denuit et al. (2019), which leverage both Bayesian and deep learning modelling frameworks, could aid in developing models with more appealing characteristics and better performance despite being computationally intensive.

Furthermore, the development of hyper-parameter tuning algorithms for NN models that can determine a range of different hyper-parameters under a systematic search

approach, would be beneficial, especially in the context of large volumes of data involving high computational cost. Additionally, it might also be interesting to consider recently developed alternative approaches such as the Shapley Additive exPlanation (SHAP), local interpretable model-agnostic explanations (LIME) and explainable artificial intelligence (XAI) for interpreting the results from network-based models and compare them with the LocalGLMnet approach utilised in this thesis.

Finally, a potentially fruitful research avenue worth exploring might be the possibility of conducting a time series analysis as data emerges for additional years. In particular, we can develop NN-based time series models by combining traditional time series models such as ARIMA with NN models such as LSTM, following an approach similar to Peters et al. (2022). In the context of the health data considered in this thesis, such an analysis can contribute towards deriving actionable data insights that can potentially guide healthcare policies and intervention efforts, as well as the development and management of relevant insurance products.

Appendix A

Appendix

A.1 Data variables and description

Name	Description	Included
ADMDATE	Date of Admission	Link
ENROLID	Unique ID Used to identify individuals. Used to link different tables	Link
AGE	Age of patient - age last birthday	From enrollment
AGEGRP	Age group	From enrollment
DATATYP	Data Type is used to identify whether the claim record is a fee for service or an encounter record	From enrollment
DOBYR	Year of birth of the patient	From enrollment
EECLASS	Used to identify the employee classification. The same code is used for dependants.	From enrollment
EESTATU	Gives the status of employment of the primary beneficiary. The same code is used for dependents.	From enrollment
EFAMID	Identifies all the members of the same family	From enrollment
EGEOLOC	Geographic Location based on the postal ZIP Code of residence of the main beneficiary. Could change in enrolment table (T)-	From enrollment
EMPREL	Defines the relation of a dependent to the primary beneficiary	From enrollment
HLTHPLAN	Health Plan Indicator shows whether the data was provided by an employer or a health plan	From enrollment
INDSTRY	The type of industry that the primary beneficiary is employed in	From enrollment
MSA	Metropolitan Statistical Area-An area with high population density at the centre and the surrounding communities having significant social and economic integration with the central region	From enrollment
PLANTYP	Gives the type of plan the individual is part of	From enrollment

Table A.1: Description of variables in the admission table in the CCAE database.

Name	Description	Included
REGION	Geographical region of the individuals residence	From enrollment
SEX	Gender of Patient	From enrollment
YEAR	Date Year Incurred	From enrollment
ADMTYP	Type of admission, e.g.:Surgical, medical, Maternity etc...	No
CASEID	Unique ID Used to identify each admission. Used to link data from different tables	Gives admission count
DAYS	Length of stay of the admission	No
DISDATE	Date of Discharge	No
DSTATUS	Status of the patient when discharged from the hospital, e.g., Alive, Dead,..	No
MDC	Major Diagnostic Category is a disease or body system-related grouping assigned by Watson Health to clinical conditions using the diagnosis codes	No
PDX	Principal diagnosis is created by the market scan for inpatient admission records.	No
POAPDX	Indicator of whether Principal diagnosis was present on admission	No
DRG	Diagnosis Related Group is assigned by MarketScan to identify the various categories of inpatient care. 1000 codes with a description containing medical terms	No
DX1-15	Diagnosis Codes associated with that admission. Upto 15 codes. DX1 is the same as the Principal diagnosis code. The remaining codes are recorded in chronological order. Not considered since using the principal diagnosis	No
DXVER	Identifies the version of diagnosis codes(ICD-9/10)	No
EIDFLAG	Indicator of the methodology used to assign ENROLID	No
ENRFLAG	Indicator of whether the data contributor provided Enrollment data. 1= Enrollment data provided	No
HOSPNET	Net payments made to the Hospital for the admission. Gross minus payment made by any other party (employer or out-of-pocket payments)	No
HOSPPAY	Gross Payments made to the Hospital for the admission	No
MHSACOVG	Flag to identify whether the mental health /substance abuse claims data is available for the individual	No
PHYFLAG	Indicates whether the data comes from a data contributor who provides detailed physician speciality coding	No
PHYSID	Unique ID to identify the principal Physician	No
PHYSNET	Net payments made to the principal physician. Gross minus payment made by any other party (employer or out-of-pocket payments)	No
PHYSPAY	Gross Payments received by the principal physician	No
POADX1-15	Indicator of whether the corresponding diagnosis was present on admission	No
PPROC	Principal procedure field provides details regarding the main procedure performed during admission. This is taken into account while assigning DRG groupings	No
PROC1-15	Procedure codes associated with that admission. Upto 15 codes. PROC1 is the same as the Principal procedure code	No
RX	Indicator whether the drug claim data is available for the individual plan group. Useful only if we are interested in Drug data.	No

Table A.1: *Cont.*

Name	**Description**	**Included**
SEQNUM	Sequence Number	No
STATE	The state in which the admission happened.Same as EGEOLOC from 1999	No
TOTCOB	Payment from coordination of benefits, i.e. by a third party other than the patient or the main submitting plan	No
TOTCOINS	Coinsurance amount paid for the admission	No
TOTCOPAY	Copayment amount paid for the admission	No
TOTDED	The total deductible amount for the admission	No
TOTNET	Sum of the net payments made to all the service providers as part of the admission	No
TOTPAY	The total gross payment for the admission	No

Table A.1: *Cont.*

Name	Description	Included
AGE	Age of patient - age last birthday	Yes
AGEGRP	Age group. 1: 0-17, 2: 18-34, 3: 35-44, 4: 45-54, 5: 55-64, 6: 65 and older	Yes
DATATYP	Data Type is used to identify whether the claim record is a fee for service or an encounter record	Yes
EECLASS	Used to identify the employee classification. The same code is used for dependants. E.g., 1: Salary Non-union, 2: Salary Union, ..., 6: Hourly Other, 7: Non-union, ...	Yes
EESTATU	Gives the status of employment of the primary beneficiary. The same code is used for dependents. 1: Active Full Time, 2: Active Part Time or Seasonal,.., 4: Medicare Eligible Retiree,...,	Yes
EFAMID	Used to identify all the members of the same family	Yes
EGEOLOC	Geographic Location based on the postal ZIP Code of residence of the main beneficiary. Could change in enrolment table (T)- 52 values	Yes
EMPREL	Defines the relation of a dependent to the primary beneficiary	Yes
ENROLID	Unique ID Used to identify individuals. Used to link data from different tables	Yes
HLTHPLAN	Health Plan Indicator shows whether the data was provided by an employer or a health plan	Yes
INDSTRY	The type of industry that the primary beneficiary is employed in	Yes
MEMDAYS	The number of days the individual was enrolled. MEMDAYS=DTEND-DTSTART+1	Yes
MSA	Metropolitan Statistical Area-An area with high population density at the centre and the surrounding communities having significant social and economic integration with the central region. 412 values. An urban-rural indicator variable is derived based on the MSA variable	Yes
PLANTYP	Gives the type of plan the individual is part of	Yes
REGION	Geographical region of the individual's residence. 1: Northeast, 2: North Central, 3: South, 4: West, 5: Unknown	Yes
SEX	Gender of Patient	Yes
YEAR	Calendar year	Yes
DTEND	Date at which the continuous enrollment period ended. Useful if interested in a time-based subset or smaller study period	Yes
DTSTART	Date at which the continuous enrollment period started	Yes
DOBYR	Patient Birth Year. In the enrollment table, it is calculated from the date of birth(not available to us). 69 values	No
MHSACOVG	Flag to identify whether the mental health /substance abuse claims data is available for the individual	No
PHYFLAG	Indicates whether the data comes from a data contributor who provides detailed physician speciality coding	No
RX	Indicator whether the drug claim data is available for the individual plan group. Useful only if we are interested in Drug data.	No
SEQNUM	Sequence Number	No
VERSION	Version number for Watson Health internal database	No

Table A.2: Description of variables in the enrollment detail table in the CCAE database.

Value	Description	Value	Description
1	Nation, unknown region	36	North Carolina
4	Connecticut	37	South Carolina
5	Maine	38	Virginia
6	Massachusetts	39	West Virginia
7	New Hampshire	41	Alabama
8	Rhode Island	42	Kentucky
9	Vermont	43	Mississippi
11	New Jersey	44	Tennessee
12	New York	46	Arkansas
13	Pennsylvania	47	Louisiana
16	Illinois	48	Oklahoma
17	Indiana	49	Texas
18	Michigan	52	Arizona
19	Ohio	53	Colorado
20	Wisconsin	54	Idaho
22	Iowa	55	Montana
23	Kansas	56	Nevada
24	Minnesota	57	New Mexico
25	Missouri	58	Utah
26	Nebraska	59	Wyoming
27	North Dakota	61	Alaska
28	South Dakota	62	California
31	Washington, DC	63	Hawaii
32	Delaware	64	Oregon
33	Florida	65	Washington
34	Georgia	97	Puerto Rico
35	Maryland		

Table A.3: Lookup table for EGEOLOC variables.

PLANTYP	**Incentive to use certain provider**	**Primary care physician (PCP) assigned?**	**Referrals from PCP to specialists required?**	**Out-of-Network Services Covered?**	**Partially or fully capitated?**
1. Basic/Major Medical Plan	No	No	n/a	n/a	No
2. Comprehensive plan	No	No	n/a	n/a	No
3. Exclusive provider organization plan	Yes	Yes	Yes	No	No
4. Health maintenance organization plan	Yes	Yes	Yes	No	Yes
5. Non-capitated (non-cap) point-of-service plan	Yes	Yes	Yes	Yes	No
6. Preferred provider organization plan	Yes	No	n/a	Yes	No
7. Capitated (Cap) or partially capitated (part cap) point-of-service plan	Yes	Yes	Yes	Yes	Yes
8. Consumer-driven health plan	Varies	No	n/a	Varies	No
9. High-Deductible health plan	Varies	No	n/a	Varies	No

Table A.4: Details regarding the characteristics of different plan types.

A.2 Additional details regarding models for admissions due to respiratory diseases

Model	# of parameters	BIC
Step 1: ∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL + PLANTYP		
+ SEX:EMPREL	35	24851
Full model	33	24866
+ AGE:SEX	34	24872
+ AGE:EMPREL	35	24875
+ SEX:UR	34	24876
+ AGE:UR	34	24877
+ REGION:UR	37	24887
+ UR:EMPREL	35	24888
+ AGE:REGION	37	24903
+ REGION:SEX	37	24909
+ AGE:PLANTYP	40	24932
+ SEX:EESTATU	41	24934
+ UR:EECLASS	41	24937
+ REGION:EMPREL	41	24938
+ UR:PLANTYP	40	24939
+ AGE:EESTATU	41	24939
+ SEX:PLANTYP	40	24941
+ UR:EESTATU	41	24944
+ EESTATU:EMPREL	48	24949
+ AGE:EECLASS	41	24952
+ SEX:EECLASS	41	24953
+ EMPREL:PLANTYP	47	24995
+ EECLASS:EMPREL	49	25036
+ REGION:EESTATU	65	25137
+ REGION:EECLASS	65	25141
+ REGION:PLANTYP	61	25147
+ EECLASS:EESTATU	97	25300
+ EESTATU:PLANTYP	87	25361
+ EECLASS:PLANTYP	85	25375

Table A.5: Details of the step-wise selection process for identifying relevant interaction terms of Poisson regression model for admissions due to respiratory diseases using BIC.

Model	# of parameters	BIC
Step 2: ∼ AGE + REGION + SEX + UR + EECLASS + EESTATU + EMPREL + PLANTYP + SEX:EMPREL		
+ AGE:SEX	36	24857
+ SEX:UR	36	24860
+ AGE:EMPREL	37	24862
+ AGE:UR	36	24862
+ REGION:UR	39	24872
+ UR:EMPREL	37	24873
+ AGE:REGION	39	24888
+ REGION:SEX	39	24894
+ SEX:EESTATU	43	24917
+ AGE:PLANTYP	42	24917
+ REGION:EMPREL	43	24922
+ UR:EECLASS	43	24923
+ AGE:EESTATU	43	24924
+ UR:PLANTYP	42	24924
+ SEX:PLANTYP	42	24926
+ UR:EESTATU	43	24929
+ AGE:EECLASS	43	24936
+ SEX:EECLASS	43	24937
+ EESTATU:EMPREL	50	24937
+ EMPREL:PLANTYP	49	24978
+ EECLASS:EMPREL	51	25021
+ REGION:EESTATU	67	25123
+ REGION:EECLASS	67	25125
+ REGION:PLANTYP	63	25130
+ EECLASS:EESTATU	99	25289
+ EESTATU:PLANTYP	89	25346
+ EECLASS:PLANTYP	87	25357

Table A.5: *Cont.*

Coefficient	Estimate	Std. Error	z value	Pr(> |z|)	Signi.
$\beta_{intercept}$	-4.9782	0.1362	-36.5470	< 2e-16	***
β_{age}	0.5973	0.0189	31.5190	< 2e-16	***
$\beta_{region1:northeast}$	-0.1742	0.0882	-1.9750	0.0483	*
$\beta_{region2:northcentral}$	-0.0508	0.0868	-0.5860	0.5580	
$\beta_{region3:south}$	-0.0644	0.0849	-0.7590	0.4480	
$\beta_{region4:west}$	-0.4991	0.0909	-5.4930	0.0000	***
$\beta_{region5:unknown}$	0.7886	0.3275	2.4080	0.0160	*
$\beta_{sex2-female}$	-0.0316	0.1378	-0.2300	0.8184	
$\beta_{ur2-urban}$	-0.1634	0.0414	-3.9450	0.0001	***
$\beta_{eeclass1:salarynon-union}$	-0.2697	0.0460	-5.8610	0.0000	***
$\beta_{eeclass2:salaryunion}$	-0.0429	0.1273	-0.3370	0.7361	
$\beta_{eeclass3:salaryother}$	-0.2561	0.0799	-3.2040	0.0014	**
$\beta_{eeclass4:hourlynon-union}$	0.2047	0.0451	4.5370	0.0000	***
$\beta_{eeclass5:hourlyunion}$	0.3475	0.0480	7.2330	0.0000	***
$\beta_{eeclass6:hourlyother}$	0.1827	0.0757	2.4130	0.0158	*
$\beta_{eeclass7:non-union}$	0.1029	0.0455	2.2640	0.0236	*
$\beta_{eeclass8:union}$	-0.0943	0.0829	-1.1370	0.2556	
$\beta_{eeclass9:unknown}$	-0.1749	0.0449	-3.8930	0.0001	***
$\beta_{eestatu1:activefulltime}$	-0.3899	0.0536	-7.2800	0.0000	***
$\beta_{eestatu2:activeparttimeorseasonal}$	-1.0823	0.1852	-5.8430	0.0000	***
$\beta_{eestatu3:earlyretiree}$	-0.2660	0.0637	-4.1750	0.0000	***
$\beta_{eestatu4:medicareeligibleretiree}$	-0.1444	0.1086	-1.3300	0.1836	
$\beta_{eestatu5:retiree(statusunknown)}$	-0.2624	0.2421	-1.0840	0.2786	
$\beta_{eestatu6:COBRAcontinuee}$	0.3885	0.1281	3.0330	0.0024	**
$\beta_{eestatu7:longtermdisability}$	1.2890	0.1291	9.9870	< 2e-16	***
$\beta_{eestatu8:survivingspouse/depend}$	0.3184	0.1546	2.0590	0.0395	*
$\beta_{eestatu9:unknown}$	0.1491	0.0649	2.2980	0.0216	*
$\beta_{emprel1:employee}$	-0.6865	0.0898	-7.6410	0.0000	***
$\beta_{emprel2:spouse}$	-0.1942	0.0914	-2.1250	0.0336	*
$\beta_{emprel3:child/other}$	0.8806	0.1744	5.0490	0.0000	***
$\beta_{plantyp2:comprehensiveplan}$	0.1292	0.0729	1.7710	0.0765	.
$\beta_{plantyp3:EPOPlan}$	0.0996	0.1467	0.6790	0.4974	
$\beta_{plantyp4:HMOPlan}$	0.0372	0.0502	0.7400	0.4592	
$\beta_{plantyp5:Non-CapPoSPlan}$	-0.1726	0.0608	-2.8400	0.0045	**
$\beta_{plantyp6:PPOPlan}$	0.0443	0.0355	1.2490	0.2118	
$\beta_{plantyp7:CaporPartCapPoSPlan}$	0.1835	0.1072	1.7120	0.0869	.
$\beta_{plantyp8:CDHP}$	-0.0004	0.0537	-0.0080	0.9936	
$\beta_{plantyp9:HDHP}$	-0.3207	0.0680	-4.7160	0.0000	***
$\beta_{sex2:emprel1}$	0.2656	0.1396	1.9030	0.0571	.
$\beta_{sex2:emprel2}$	-0.1375	0.1411	-0.9740	0.3299	
$\beta_{sex2:emprel3}$	-0.1281	0.2731	-0.4690	0.6390	

Table A.6: Coefficient estimates of the Poisson regression model for admissions related to respiratory diseases; with interaction terms for gender SEX and EMPREL variables

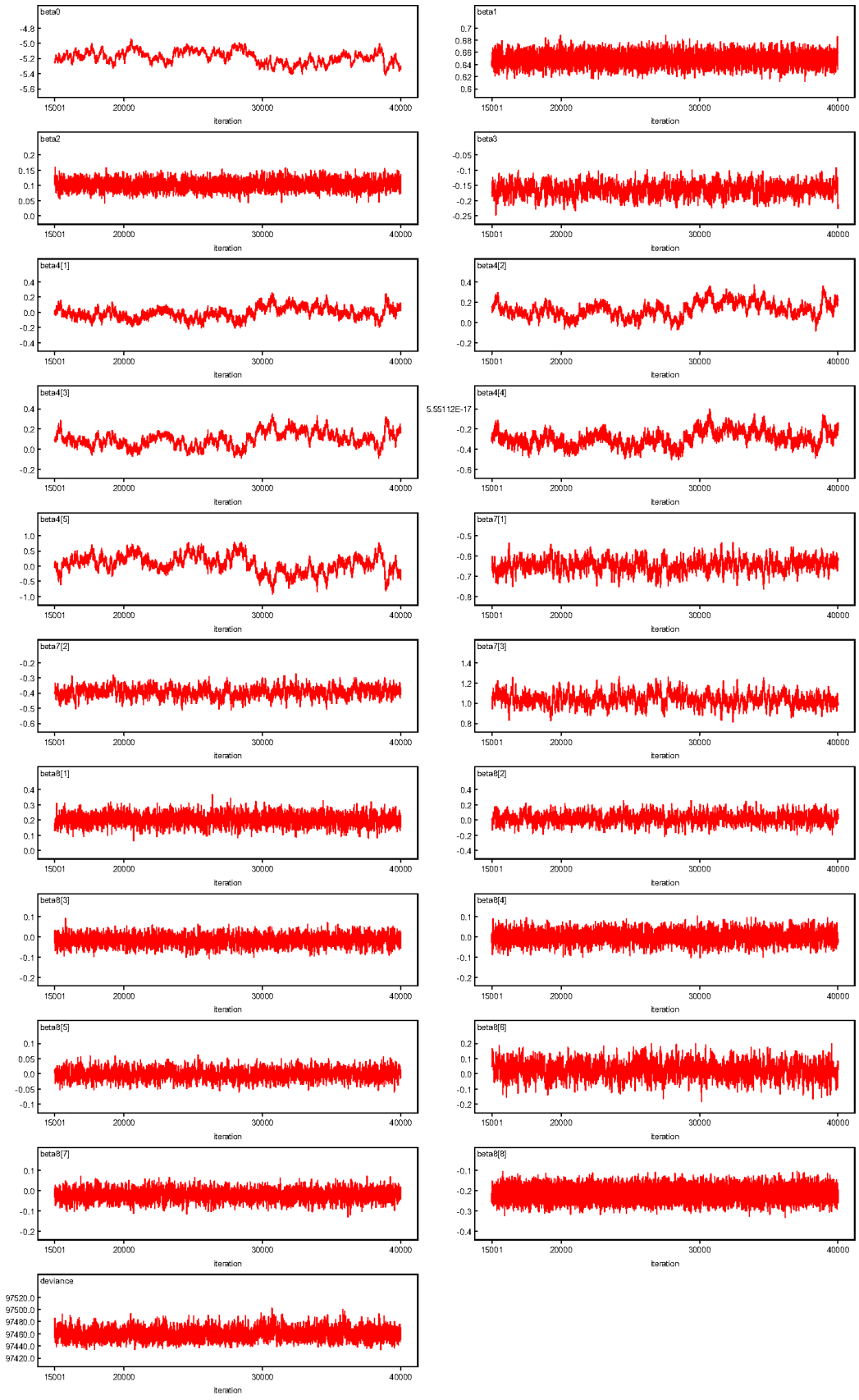


Figure A.1: Trace plots for the parameters of the BP model for admissions related to respiratory diseases in the years 2016-2019.

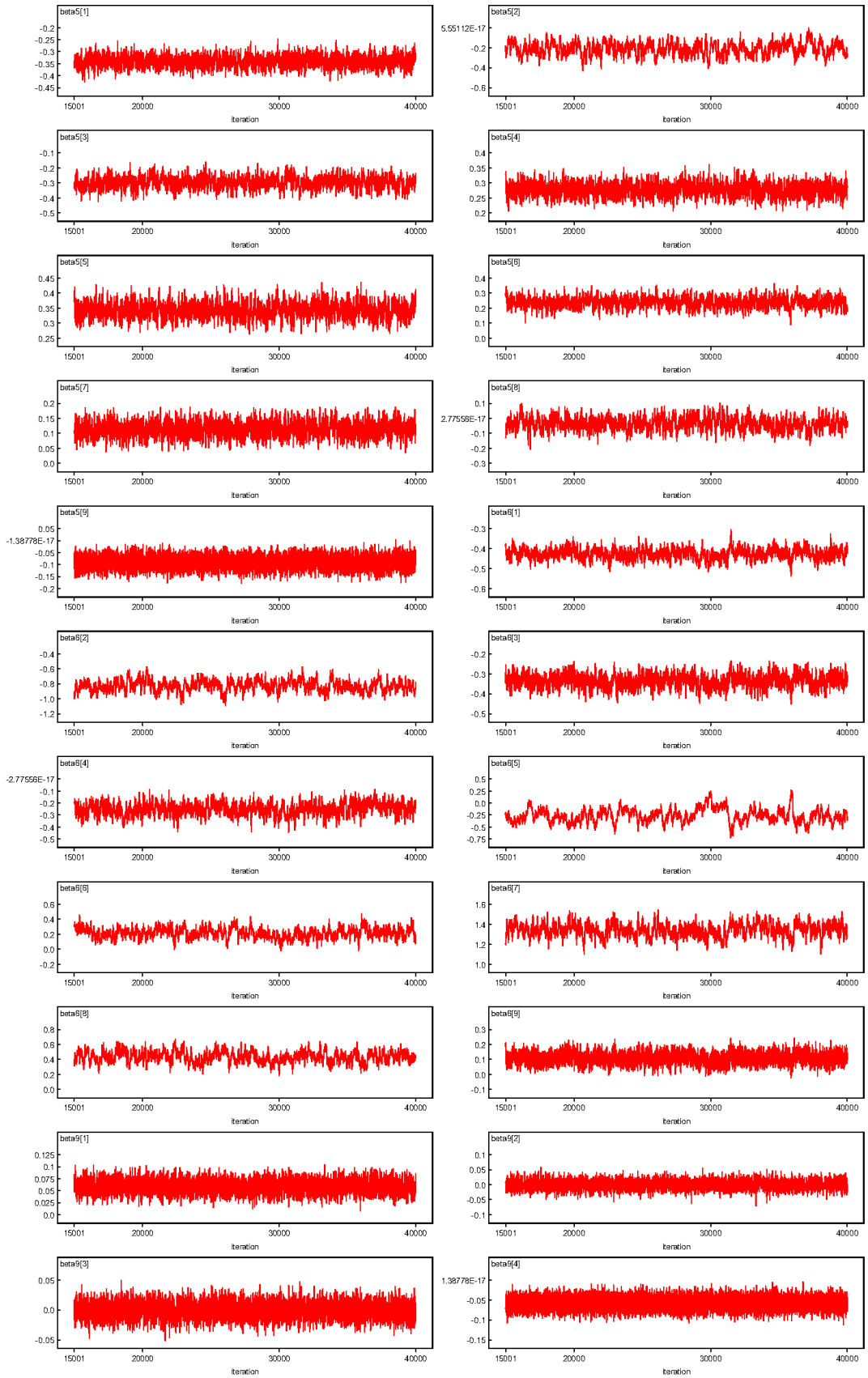


Figure A.1: *Cont.*

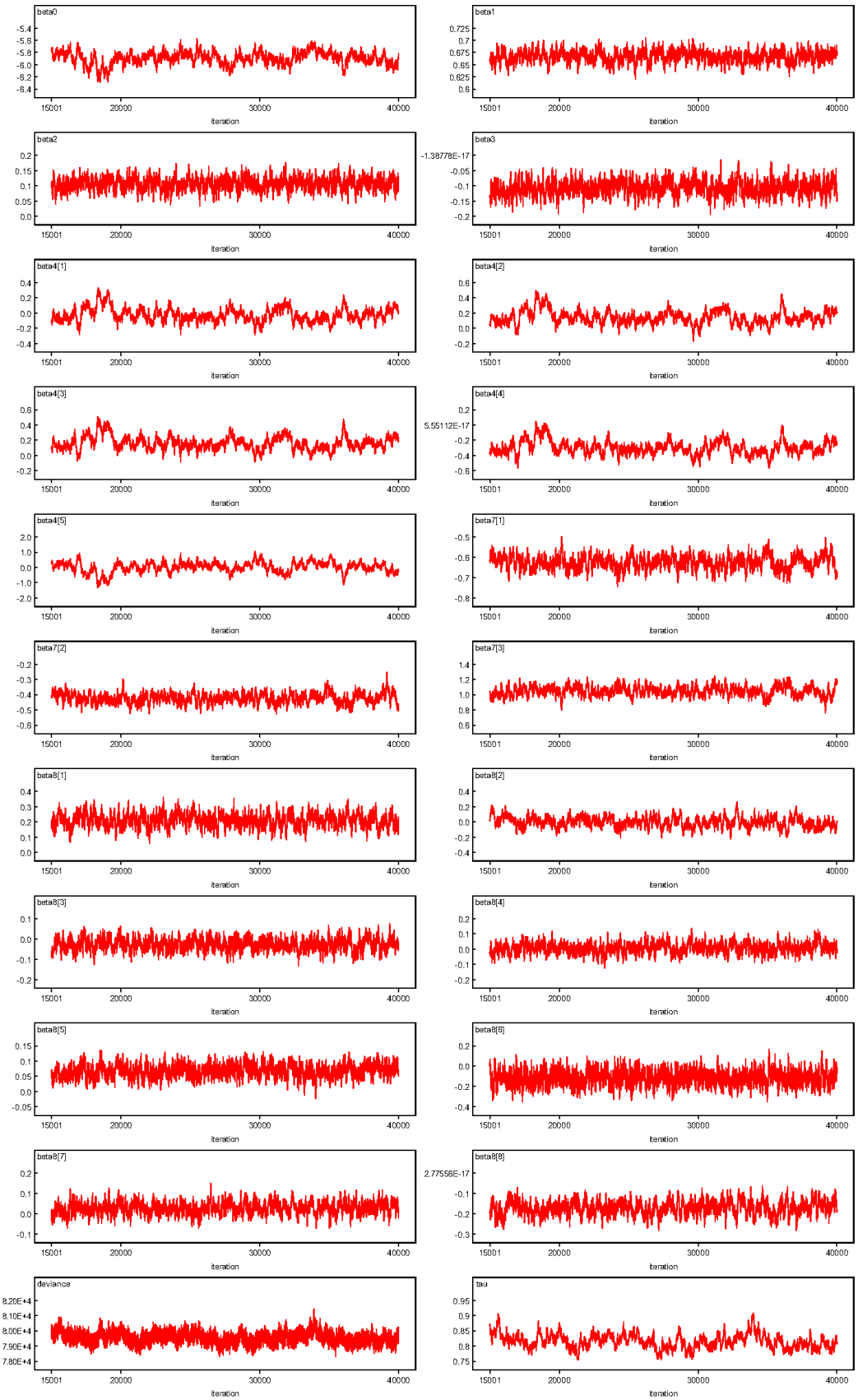


Figure A.2: Trace plots for the parameters of the BP-LN model for admissions related to respiratory diseases in the years 2016-2019.

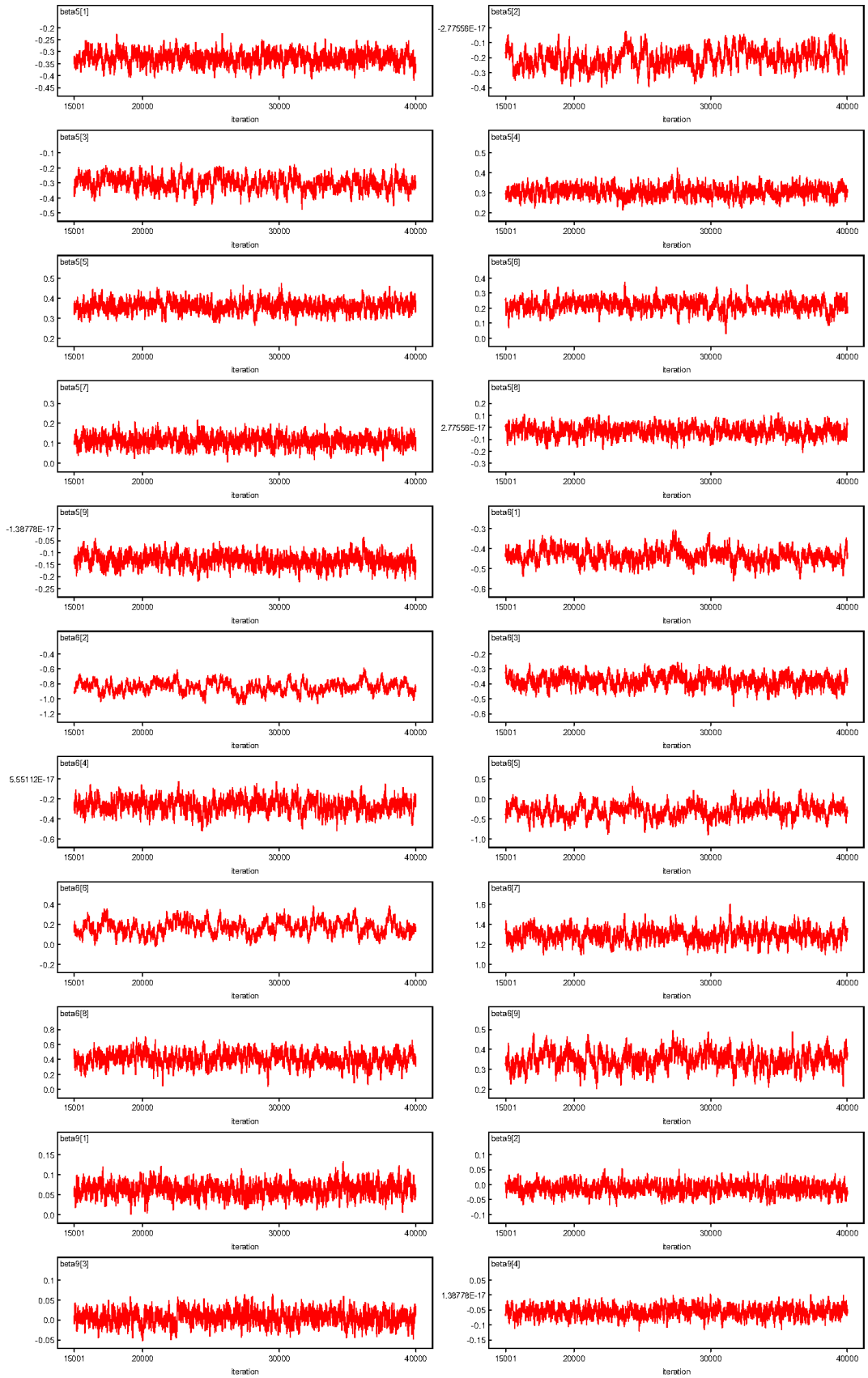


Figure A.2: *Cont.*

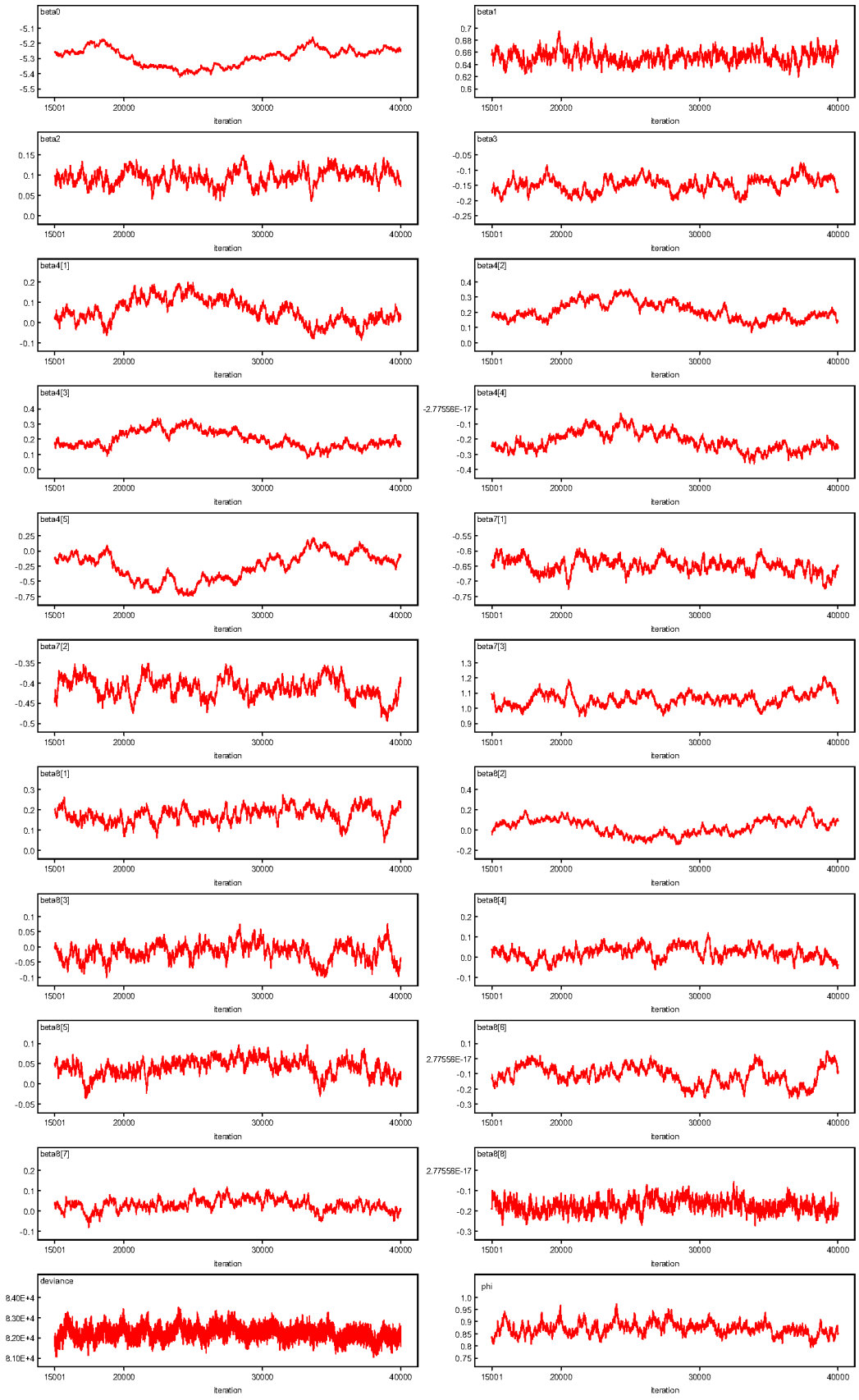


Figure A.3: Trace plots for the parameters of the BP-G model for admissions related to respiratory diseases in the years 2016-2019.

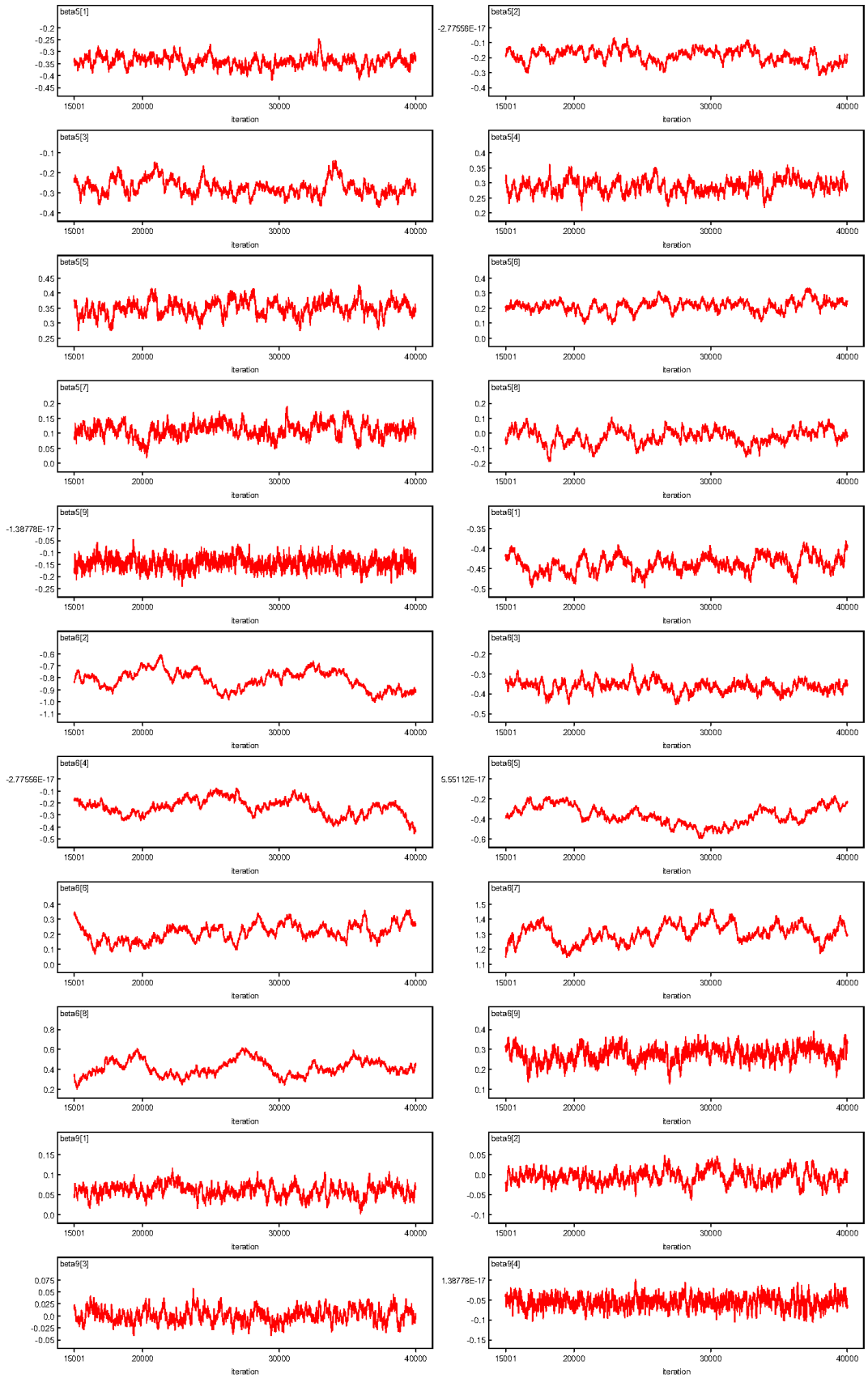


Figure A.3: *Cont.*

A.3 Details of the model formulation for admissions due to cancer among males

Model	# of parameters	BIC
Step 1: ∼ poly(AGE, degree = 3) + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
+ EECLASS:EESTATU	101	55425
Full model w/ age polynomial	37	56434
+ UR:EMPREL	39	56458
+ UR:YEAR	40	56466
+ poly(AGE, degree = 3):UR	40	56468
+ EESTATU:EMPREL	52	56470
+ REGION:UR	41	56481
+ EMPREL:YEAR	43	56500
+ poly(AGE, degree = 3):EMPREL	43	56502
+ UR:PLANTYP	44	56513
+ UR:EECLASS	45	56513
+ REGION:EMPREL	45	56518
+ UR:EESTATU	45	56522
+ poly(AGE, degree = 3):YEAR	46	56534
+ poly(AGE, degree = 3):REGION	49	56561
+ REGION:YEAR	49	56572
+ EMPREL:PLANTYP	51	56575
+ EECLASS:EMPREL	53	56597
+ poly(AGE, degree = 3):EESTATU	61	56627
+ PLANTYP:YEAR	58	56633
+ poly(AGE, degree = 3):PLANTYP	58	56651
+ EESTATU:PLANTYP	92	56668
+ EECLASS:YEAR	61	56682
+ EESTATU:YEAR	61	56698
+ poly(AGE, degree = 3):EECLASS	61	56699
+ REGION:PLANTYP	65	56725
+ REGION:EECLASS	69	56746
+ REGION:EESTATU	69	56759
+ EECLASS:PLANTYP	90	56935

Table A.7: Details of the step-wise selection process for identifying relevant interaction terms of Poisson regression model for admissions related to neoplasms in males using BIC.

Model	# of parameters	BIC
Step 2: ∼ poly(AGE, degree = 3) + REGION + UR + EECLASS + EESTATU + EMPREL +PLANTYP + YEAR + EECLASS : EESTATU		
+ UR:EMPREL	103	55450
+ EESTATU:EMPREL	116	55453
+ UR:YEAR	104	55459
+ poly(AGE, degree = 3):UR	104	55460
+ REGION:UR	105	55473
+ EMPREL:YEAR	107	55490
+ poly(AGE, degree = 3):EMPREL	107	55494
+ UR:EECLASS	109	55507
+ UR:PLANTYP	108	55508
+ UR:EESTATU	109	55508
+ REGION:EMPREL	109	55509
+ poly(AGE, degree = 3):YEAR	110	55526
+ poly(AGE, degree = 3):REGION	113	55549
+ REGION:YEAR	113	55560
+ EMPREL:PLANTYP	115	55570
+ EECLASS:EMPREL	117	55591
+ poly(AGE, degree = 3):EESTATU	125	55619
+ poly(AGE, degree = 3):PLANTYP	122	55646
+ PLANTYP:YEAR	122	55649
+ EECLASS:YEAR	125	55680
+ poly(AGE, degree = 3):EECLASS	125	55689
+ EESTATU:YEAR	125	55694
+ REGION:PLANTYP	129	55724
+ REGION:EECLASS	133	55753
+ REGION:EESTATU	133	55758
+ EECLASS:PLANTYP	154	55981
+ EESTATU:PLANTYP	155	56016

Table A.7: *Cont.*

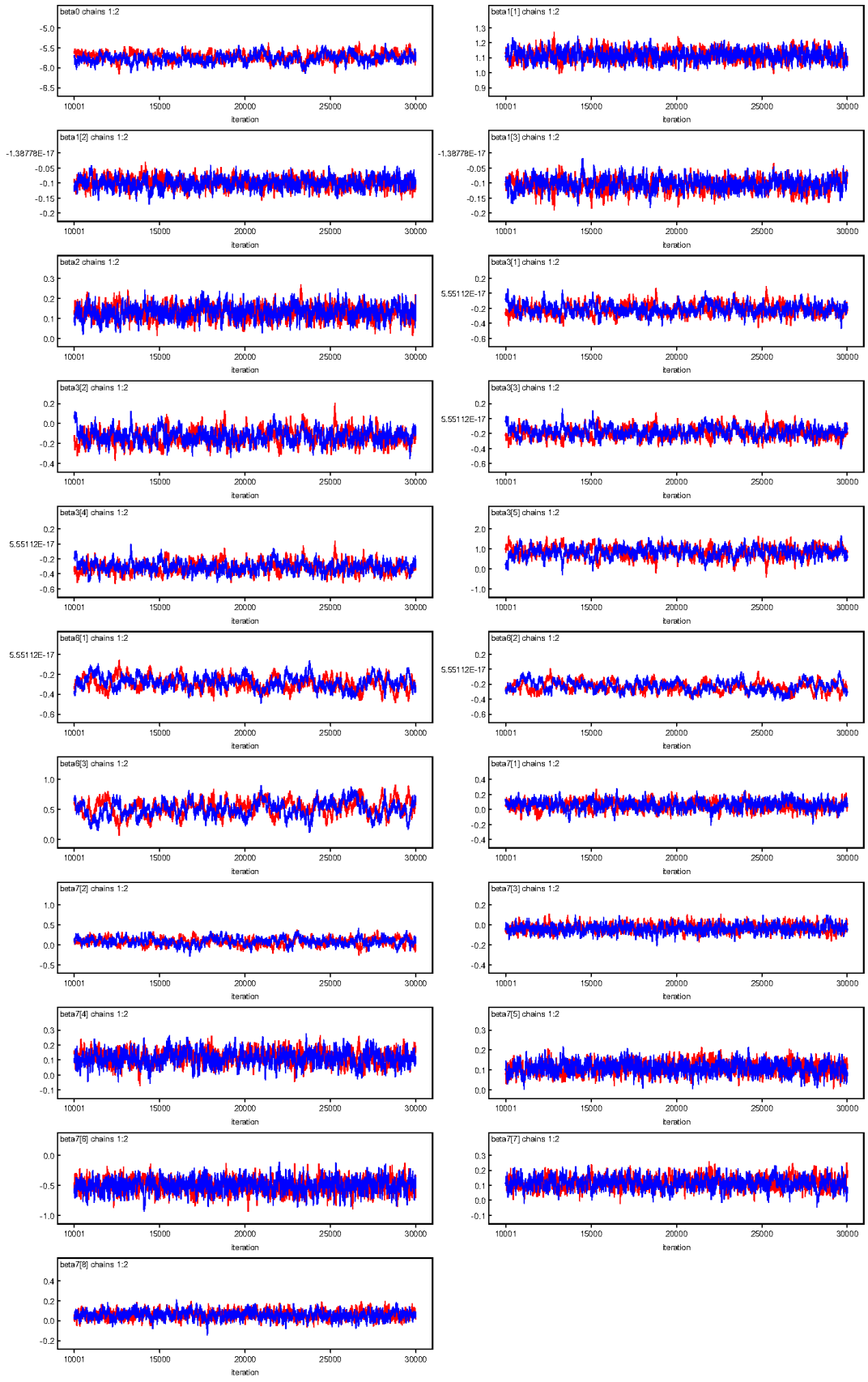


Figure A.4: Trace plots for the parameters of the BP-LN model for admissions related to neoplasms among males.

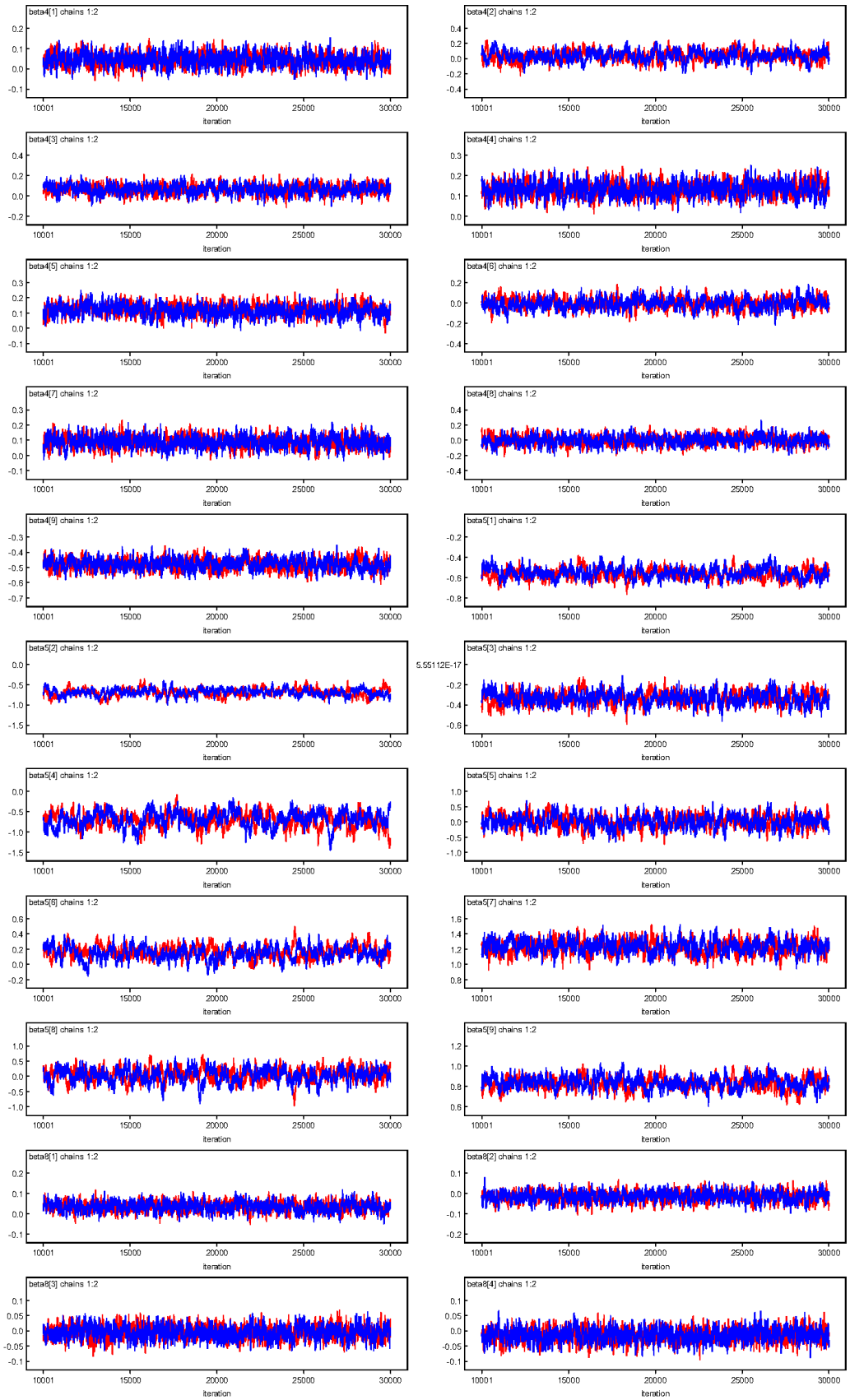


Figure A.4: *Cont.*

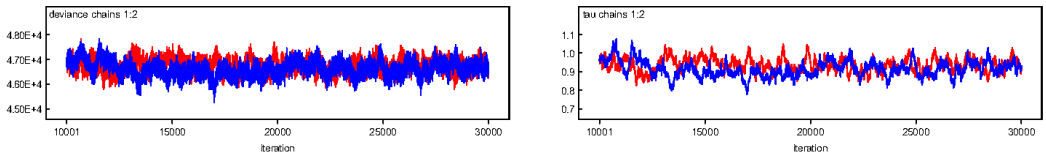


Figure A.4: *Cont.*

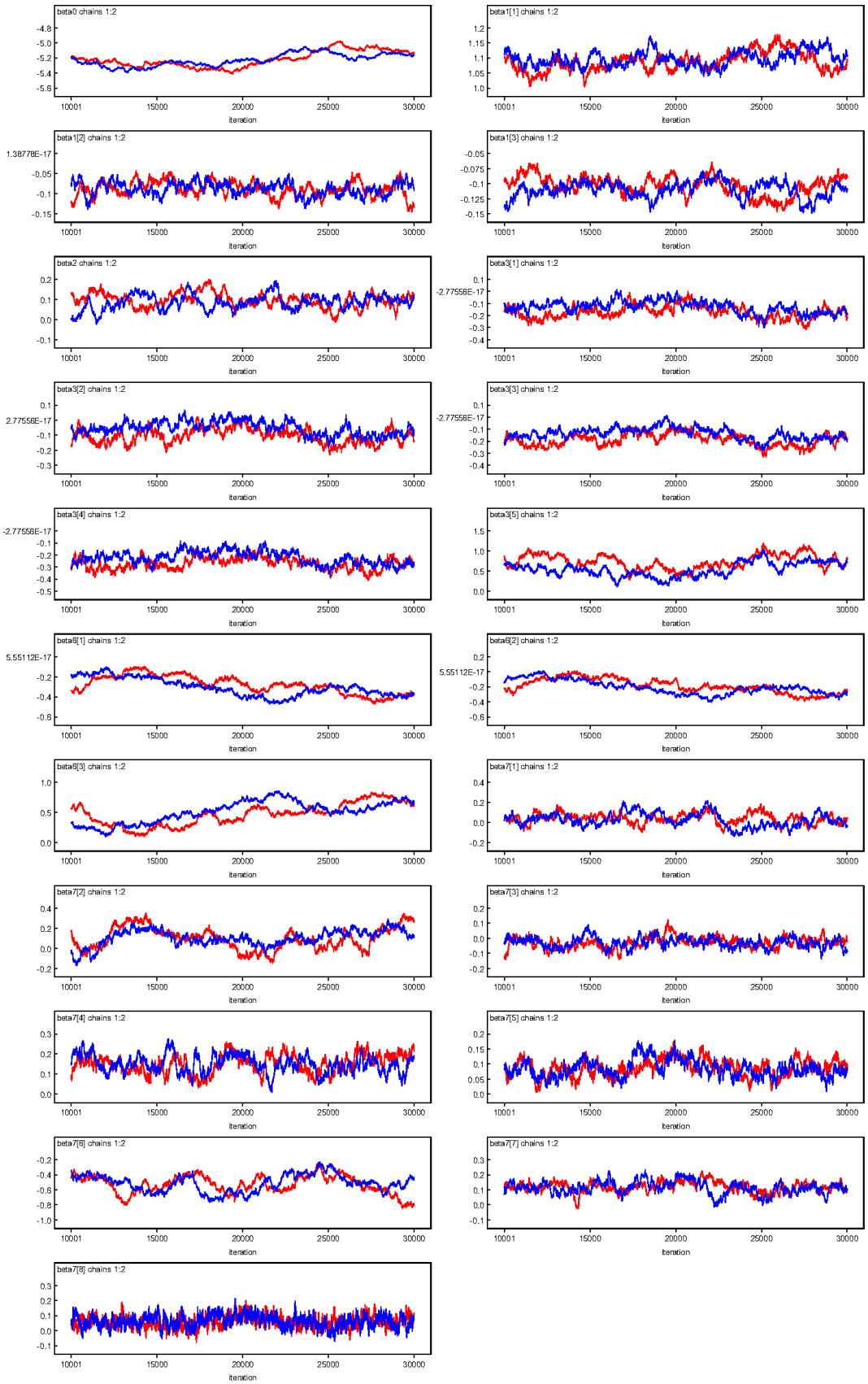


Figure A.5: Trace plots for the parameters of the BP-G model for admissions related to neoplasms among males.

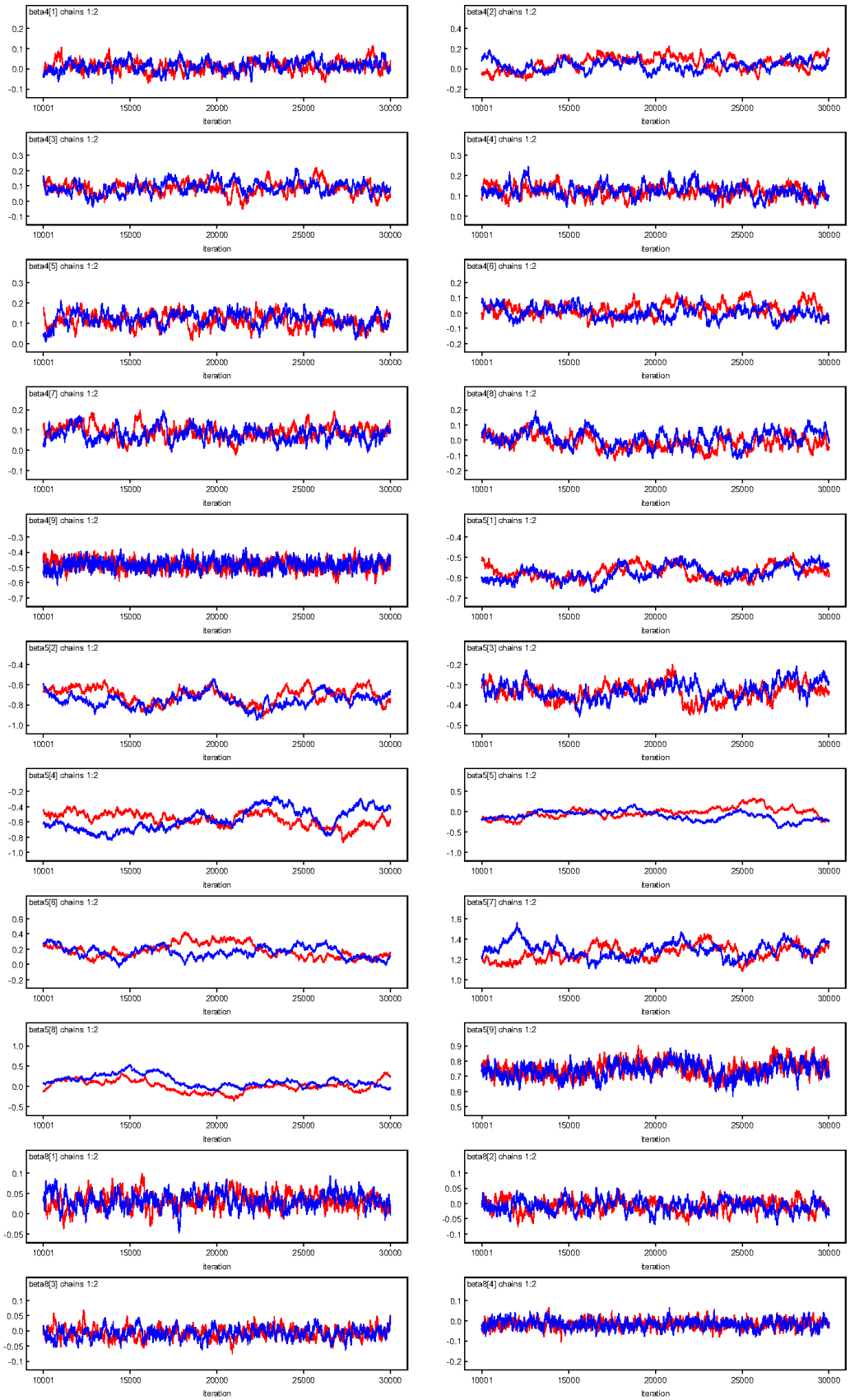


Figure A.5: *Cont.*

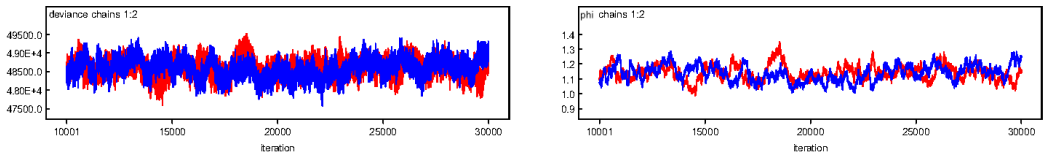


Figure A.5: *Cont.*

A.4 Details of the model formulation for admissions due to cancer among females

Model	# of estimated parameters	AIC
Step 1∼ AGE + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
Full model	35	77833
- YEAR	32	77838
- PLANTYP	28	77855
- UR	34	77860
- REGION	31	77904
- EMPREL	33	77965
- EECLASS	27	78287
- AGE	34	78899
- EESTATU	27	79314

Table A.8: Summary of the variable selection process for Poisson regression model for admissions related to cancer in females using AIC.

Model	# of estimated parameters	BIC
Step 1∼ AGE + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
- PLANTYP	28	78144
- YEAR	32	78169
Full model	35	78194
- UR	34	78212
- REGION	31	78224
- EMPREL	33	78306
- EECLASS	27	78566
- AGE	34	79250
- EESTATU	27	79592
Step 2∼ AGE + UR + REGION + EECLASS + EESTATU + EMPREL + YEAR		
- YEAR	25	78119
- UR	27	78158
- REGION	24	78179
- EMPREL	26	78257
- EECLASS	20	78538
- AGE	27	79199
- EESTATU	20	79540
Step 3∼ AGE + UR + REGION + EECLASS + EESTATU + EMPREL		
- UR	24	78134
- REGION	21	78154
- EMPREL	23	78233
- EECLASS	17	78511
- AGE	24	79173
- EESTATU	17	79525

Table A.9: Summary of the variable selection process for Poisson regression model for admissions related to cancer in females using BIC.

Degree	# of estimated parameters	BIC	AIC
r=1	35	78194	77833
r=2	36	77837	77465
r=3	37	77734	77352
r=4	38	77734	77342
r=5	39	77735	77332

Table A.10: AIC and BIC values of the Poisson regression models for admissions due to cancer among females with different degrees ($r = 1, .., 5$) for the age polynomial.

Model	# of estimated parameters	BIC
Step 1: ∼ polynomial(AGE, degree = 3) + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP		
Full model w/ age polynomial	37	77734.17
+ EECLASS:EESTATU	101	75607.34
+ EESTATU:EMPREL	116	75529.01
+ poly(AGE, degree = 3):EMPREL	123	75516.77

Table A.11: Summary of the step-wise selection process for identifying relevant interaction terms of Poisson regression model for cancer related admissions in females using BIC.

Model	# of estimated parameters	BIC
Step 1: ∼ poly(AGE, degree = 3) + UR + REGION + EECLASS + EESTATU + EMPREL + PLANTYP + YEAR		
+ EECLASS:EESTATU	101	75607
+ EESTATU:EMPREL	52	77675
+ EESTATU:PLANTYP	91	77711
+ poly(AGE, degree = 3):EMPREL	43	77733
Full model	37	77734
+ UR:EMPREL	39	77752
+ UR:YEAR	40	77764
+ poly(AGE, degree = 3):UR	40	77767
+ REGION:UR	41	77773
+ UR:PLANTYP	44	77793
+ REGION:EMPREL	45	77796
+ EMPREL:YEAR	43	77803
+ UR:EECLASS	45	77819
+ UR:EESTATU	45	77820
+ poly(AGE, degree = 3):REGION	49	77824
+ poly(AGE, degree = 3):YEAR	46	77835
+ REGION:YEAR	49	77861
+ EMPREL:PLANTYP	51	77863
+ EECLASS:EMPREL	53	77894
+ poly(AGE, degree = 3):EESTATU	61	77912
+ PLANTYP:YEAR	58	77929
+ poly(AGE, degree = 3):PLANTYP	58	77955
+ poly(AGE, degree = 3):EECLASS	61	77972
+ EESTATU:YEAR	61	77986
+ EECLASS:YEAR	61	78001
+ REGION:EESTATU	69	78012
+ REGION:PLANTYP	65	78024
+ REGION:EECLASS	68	78036
+ EECLASS:PLANTYP	91	78232

Table A.12: Details of the step-wise selection process for identifying relevant interaction terms of Poisson regression model for admissions related to neoplasms in females using BIC.

Model	# of estimated parameters	BIC
Step 2: ∼ poly(AGE, degree = 3) + UR+ REGION + EECLASS + EESTATU + EMPREL +PLANTYP + YEAR + EECLASS : EESTATU		
+ EESTATU:EMPREL	116	75529
+ poly(AGE, degree = 3):EMPREL	107	75613
+ UR:EMPREL	103	75626
+ UR:YEAR	104	75639
+ poly(AGE, degree = 3):UR	104	75640
+ REGION:UR	105	75646
+ REGION:EMPREL	109	75666
+ UR:PLANTYP	108	75672
+ EMPREL:YEAR	107	75677
+ UR:EECLASS	109	75685
+ UR:EESTATU	109	75695
+ poly(AGE, degree = 3):REGION	113	75706
+ poly(AGE, degree = 3):YEAR	110	75710
+ EMPREL:PLANTYP	115	75733
+ REGION:YEAR	113	75734
+ EECLASS:EMPREL	117	75769
+ poly(AGE, degree = 3):EESTATU	125	75796
+ poly(AGE, degree = 3):PLANTYP	122	75831
+ PLANTYP:YEAR	122	75837
+ poly(AGE, degree = 3):EECLASS	125	75841
+ EESTATU:YEAR	125	75860
+ REGION:PLANTYP	129	75866
+ EECLASS:YEAR	125	75881
+ REGION:EECLASS	132	75931
+ REGION:EESTATU	133	75943
+ EESTATU:PLANTYP	155	76166
+ EECLASS:PLANTYP	155	76197
Step 3: ∼ poly(AGE, degree = 3) + UR+ REGION + EECLASS + EESTATU + EMPREL +PLANTYP + YEAR + EECLASS : EESTATU + EESTATU:EMPREL		
+ poly(AGE, degree = 3):EMPREL	123	75517
+ UR:YEAR	119	75560
+ UR:EMPREL	119	75561
+ poly(AGE, degree = 3):UR	119	75562
+ REGION:UR	120	75568
+ UR:PLANTYP	123	75594
+ EMPREL:YEAR	122	75598
+ REGION:EMPREL	124	75598
+ UR:EECLASS	124	75606
+ UR:EESTATU	124	75618
+ poly(AGE, degree = 3):REGION	128	75627
+ poly(AGE, degree = 3):YEAR	125	75631
+ EECLASS:EMPREL	133	75644
+ REGION:YEAR	128	75656
+ EMPREL:PLANTYP	130	75669
+ poly(AGE, degree = 3):EESTATU	140	75724
+ poly(AGE, degree = 3):PLANTYP	137	75751
+ PLANTYP:YEAR	137	75759
+ poly(AGE, degree = 3):EECLASS	140	75763
+ EESTATU:YEAR	140	75781
+ REGION:PLANTYP	144	75790
+ EECLASS:YEAR	141	75815
+ REGION:EECLASS	147	75852
+ REGION:EESTATU	148	75866
+ EESTATU:PLANTYP	170	76092
+ EECLASS:PLANTYP	171	76135

Table A.12: *Cont.*

Model	# of estimated parameters	BIC
Step 4: ∼ poly(AGE, degree = 3) + UR+ REGION + EECLASS + EESTATU + EMPREL +PLANTYP + YEAR + EECLASS : EESTATU + EESTATU:EMPREL+ poly(AGE, degree = 3):EMPREL		
+ UR:EMPREL	124	75525
+ UR:YEAR	125	75536
+ poly(AGE, degree = 3):UR	125	75537
+ REGION:UR	126	75544
+ UR:PLANTYP	129	75569
+ EMPREL:YEAR	128	75573
+ REGION:EMPREL	130	75574
+ UR:EECLASS	130	75582
+ UR:EESTATU	130	75593
+ EECLASS:EMPREL	138	75596
+ poly(AGE, degree = 3):YEAR	131	75607
+ poly(AGE, degree = 3):REGION	134	75610
+ REGION:YEAR	134	75631
+ EMPREL:PLANTYP	136	75641
+ poly(AGE, degree = 3):EESTATU	146	75719
+ poly(AGE, degree = 3):PLANTYP	143	75731
+ poly(AGE, degree = 3):EECLASS	146	75734
+ PLANTYP:YEAR	143	75734
+ EESTATU:YEAR	146	75756
+ REGION:PLANTYP	150	75766
+ EECLASS:YEAR	146	75778
+ REGION:EECLASS	153	75828
+ REGION:EESTATU	154	75841
+ EESTATU:PLANTYP	176	76067
+ EECLASS:PLANTYP	176	76098

Table A.12: *Cont.*

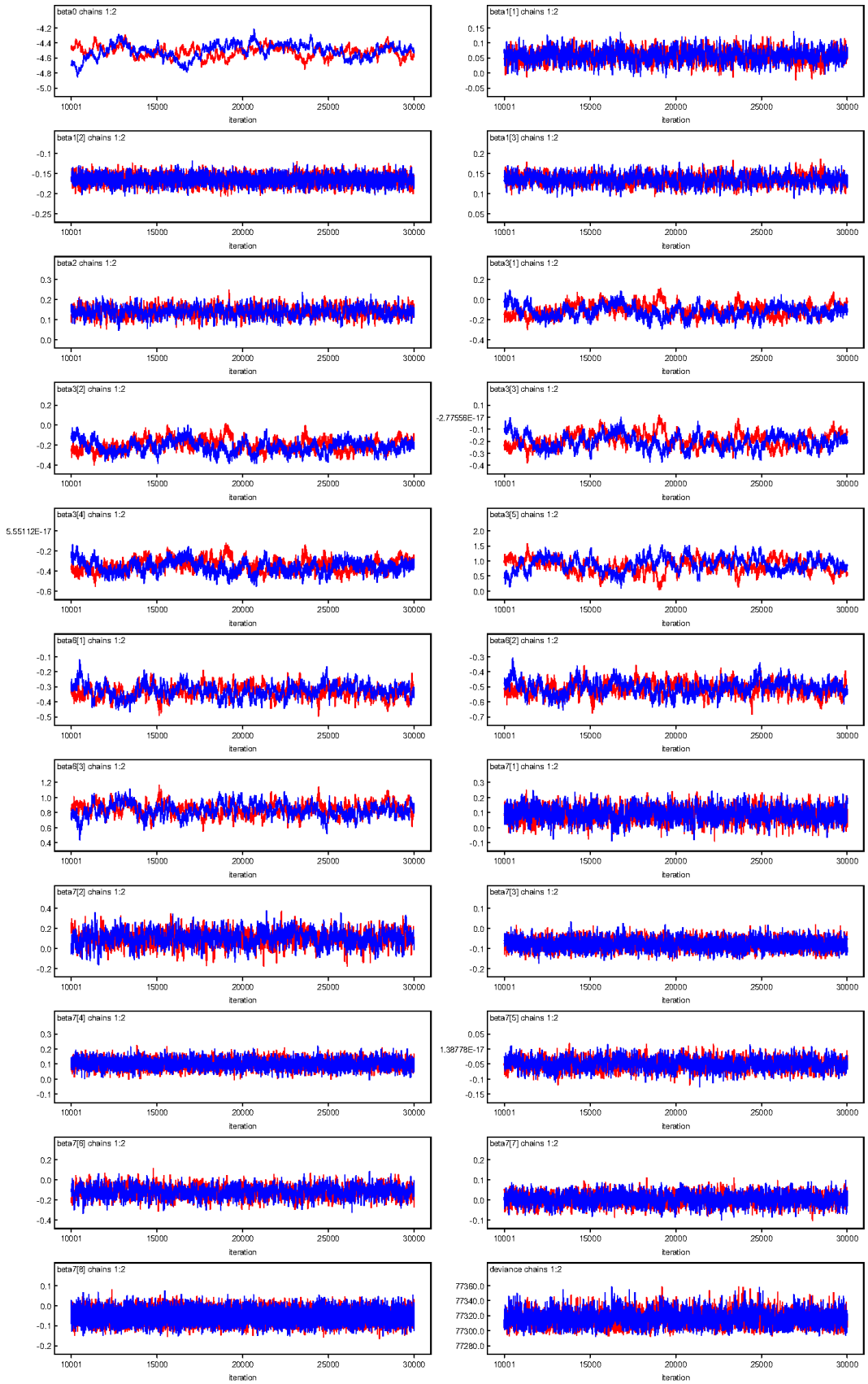


Figure A.6: Trace plots for the parameters of the BP model for admissions related to neoplasms among females.

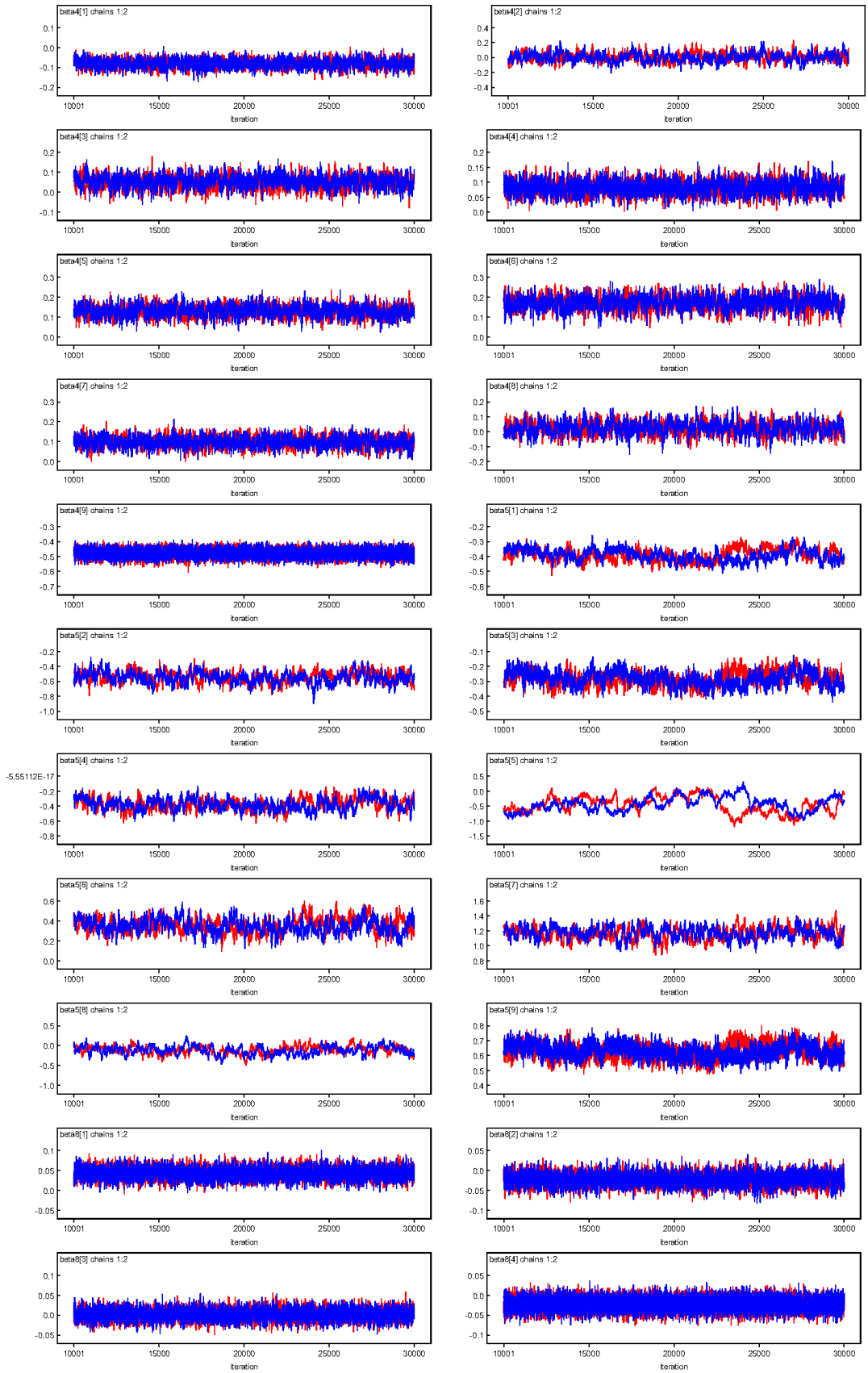


Figure A.6: *Cont.*

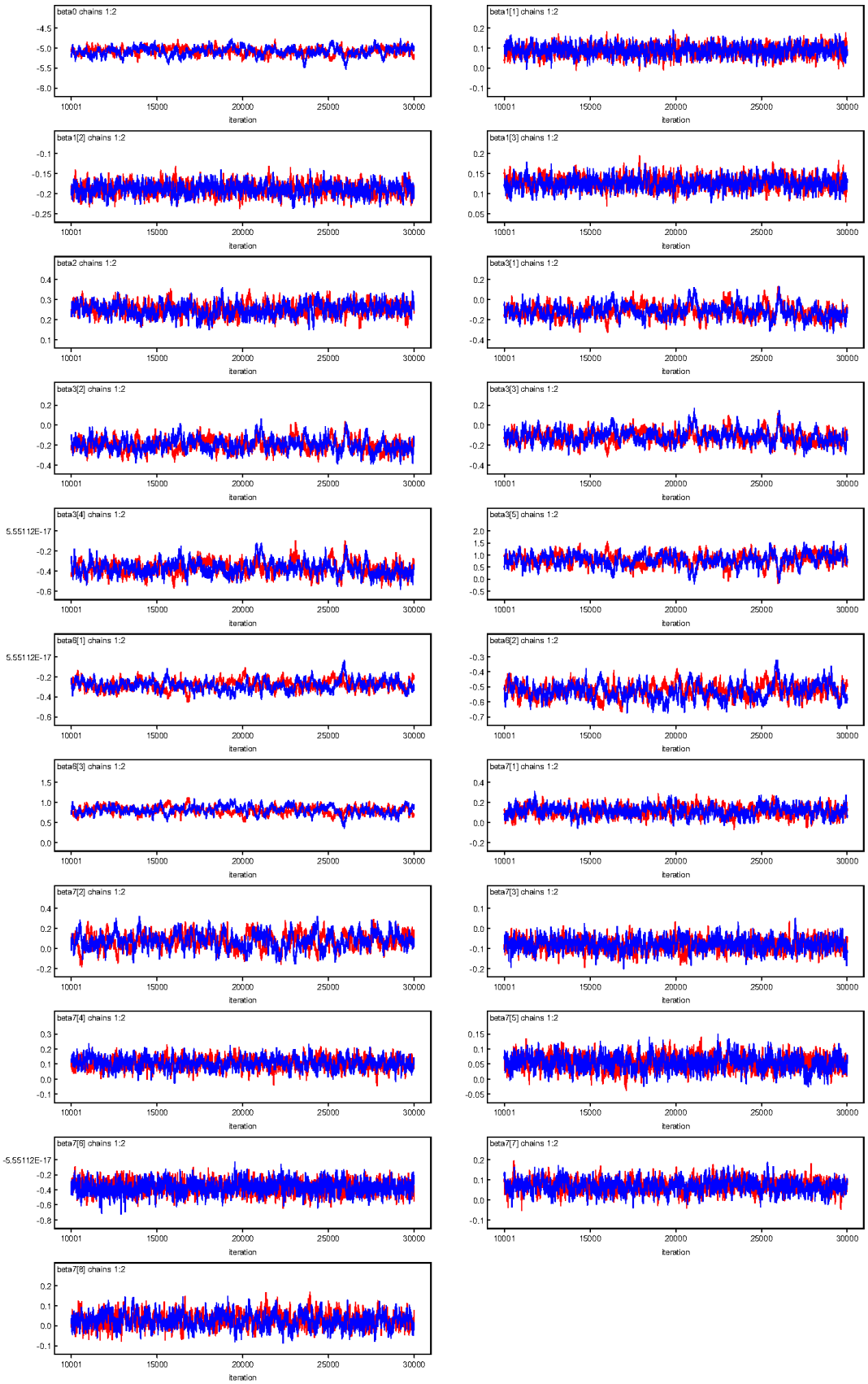


Figure A.7: Trace plots for the parameters of the BP-LN model for admissions related to neoplasms among females.

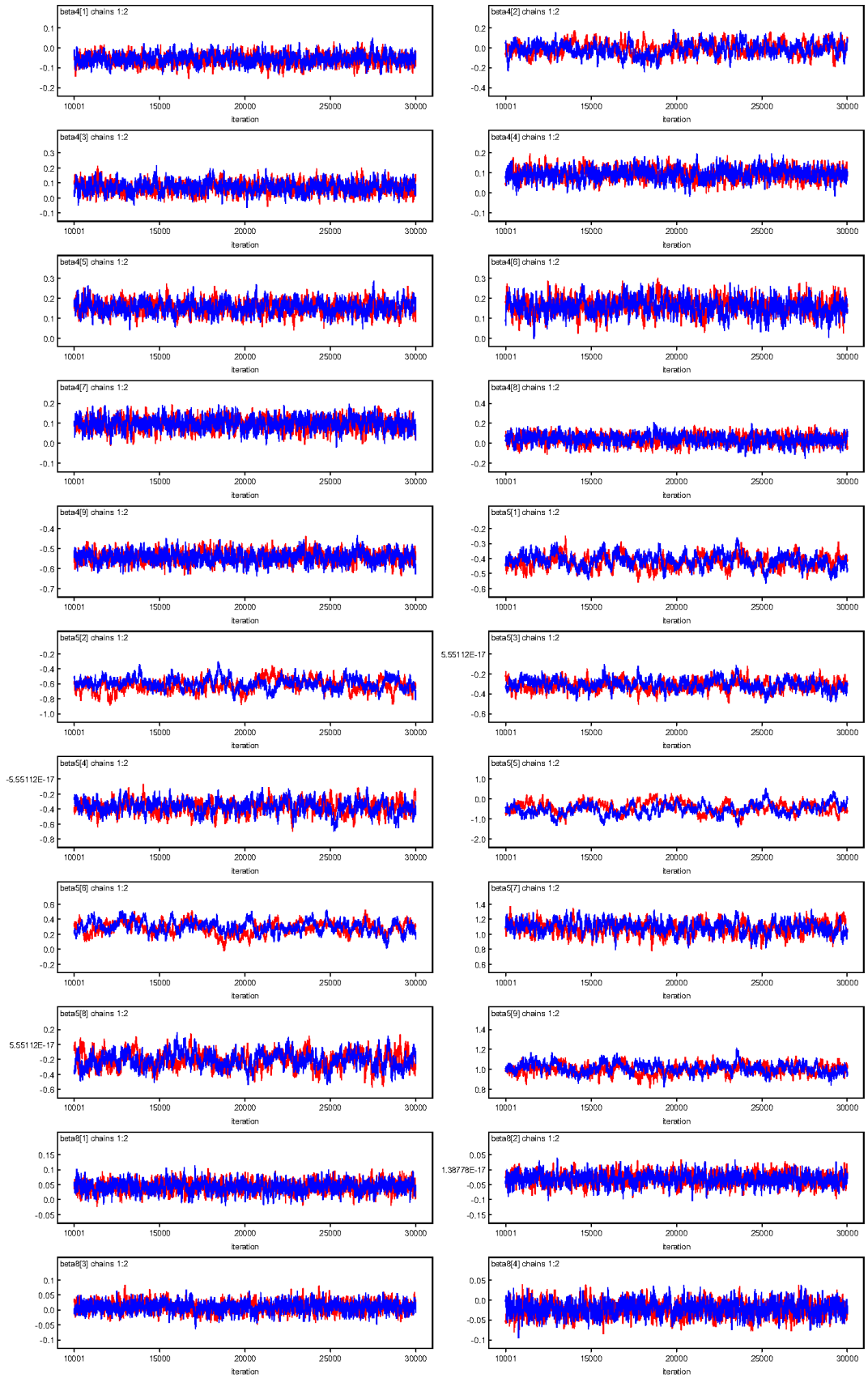


Figure A.7: *Cont.*

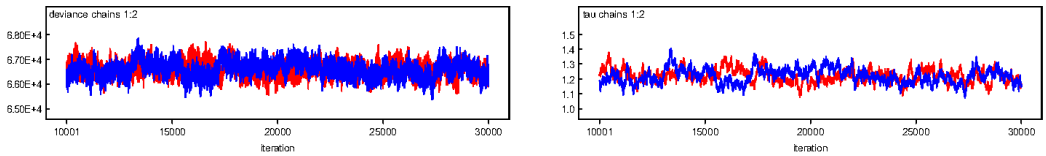


Figure A.7: *Cont.*

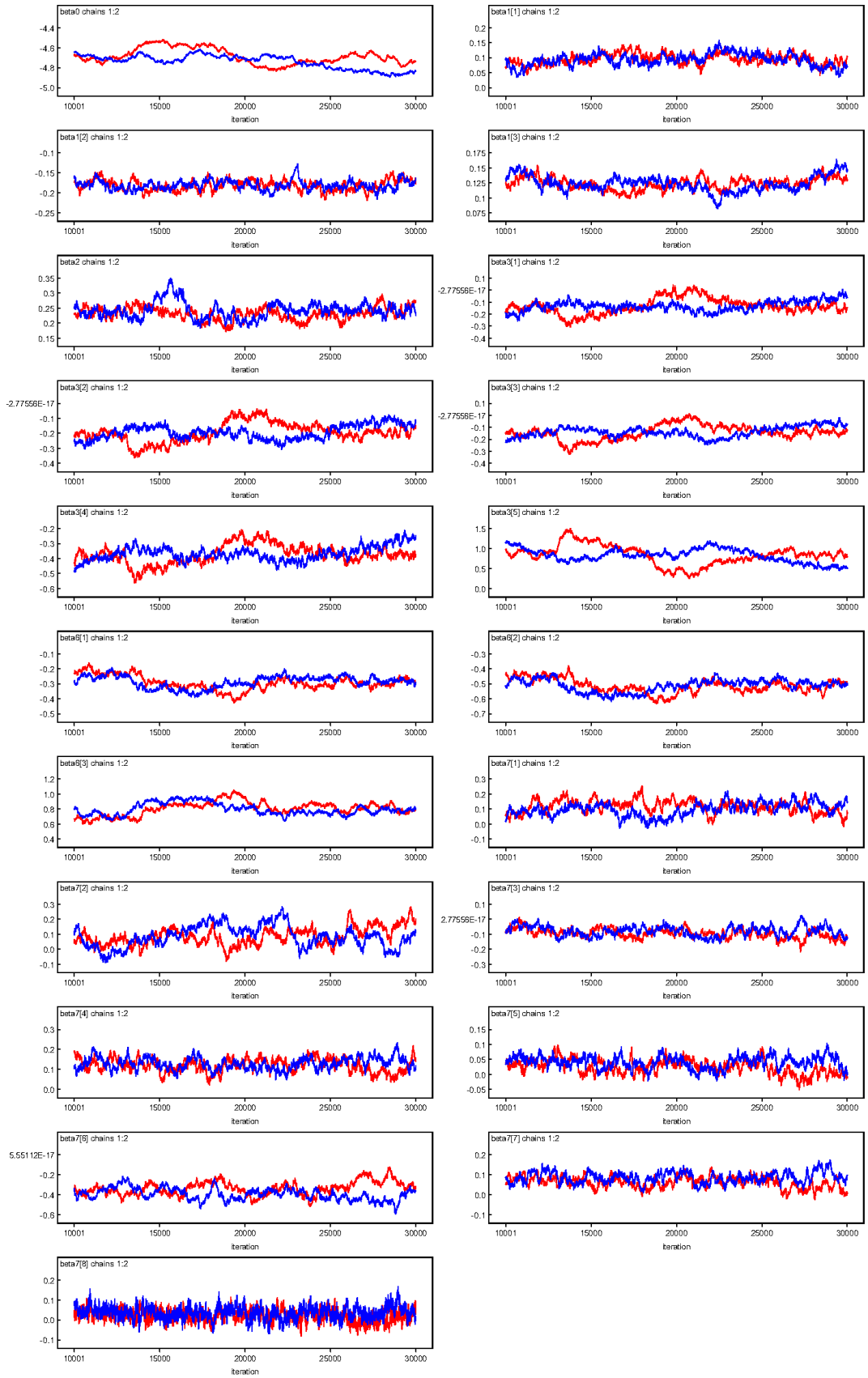


Figure A.8: Trace plots for the parameters of the BP-G model for admissions related to neoplasms among females.

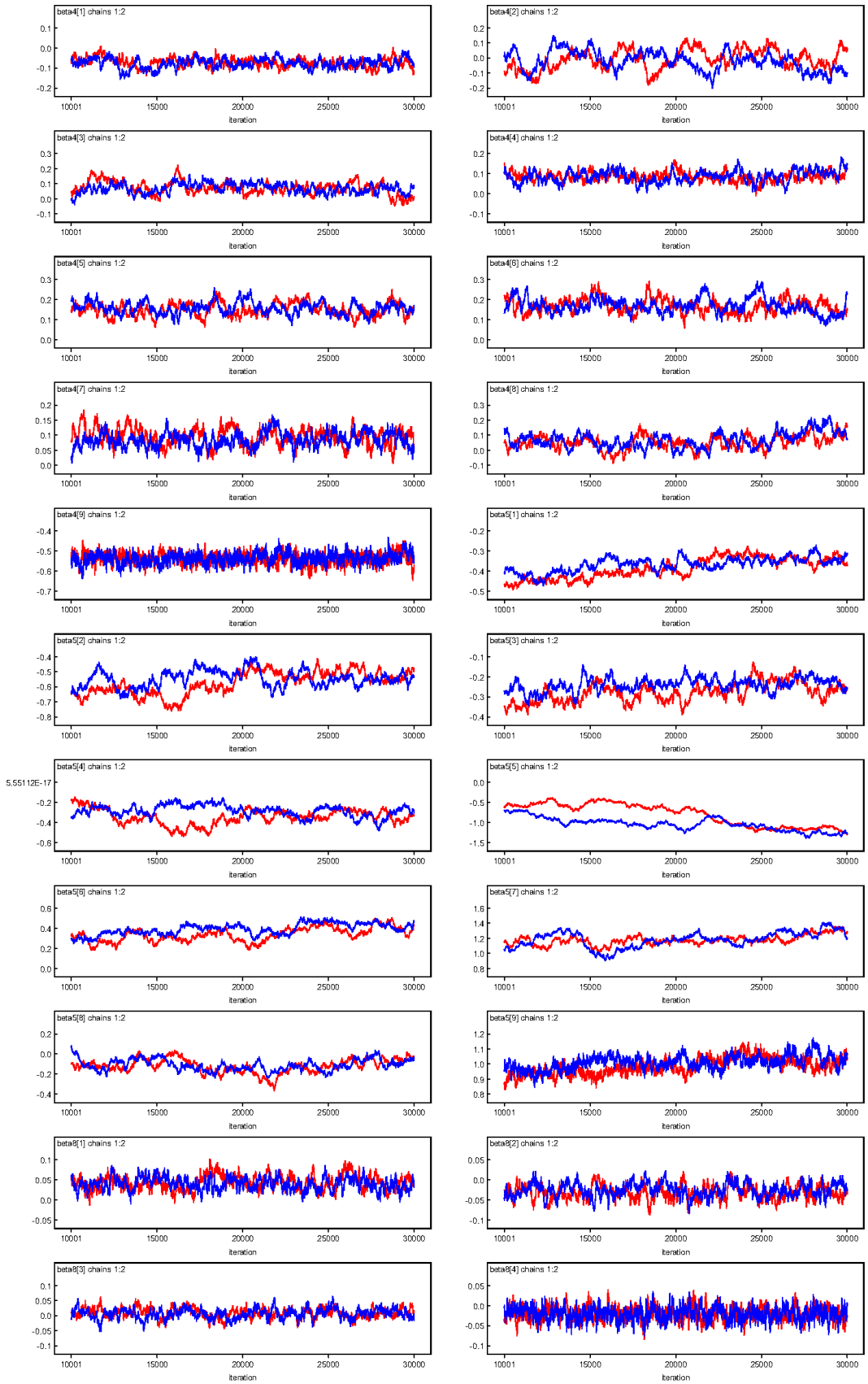


Figure A.8: *Cont.*

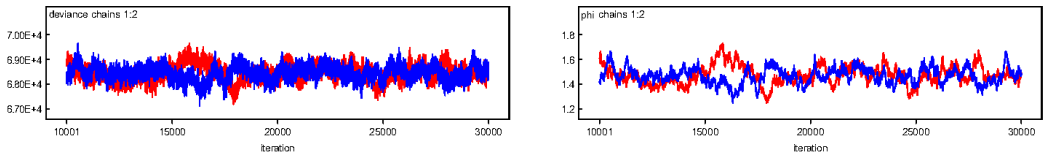


Figure A.8: *Cont.*

A.5 Code and additional information for NN models

Model	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
Data				0.0027
Pois. GLM		2.6811	2.5516	0.0027
NN (100,75,50)	10,000	2.0815	4.3194	0.0027
NN (75,50,25)	10,000	2.2348	3.3182	0.003
NN (50,35,25)	10,000	2.3546	3.0803	0.0029
NN (35,25,20)	10,000	2.4291	2.9139	0.0028
NN (25,20,15)	10,000	2.5097	2.8273	0.0029
NN (20,15,10)	10,000	2.5533	2.6676	0.0028
NN (15,10,5)	10,000	2.5902	2.6593	0.0028
NN (100,75,50)	30,000	2.1473	3.4493	0.0028
NN (75,50,25)	30,000	2.3359	2.9862	0.0027
NN (50,35,25)	30,000	2.4454	2.8612	0.0028
NN (35,25,20)	30,000	2.5116	2.6836	0.0028
NN (25,20,15)	30,000	2.5618	2.6794	0.0027
NN (20,15,10)	30,000	2.5917	2.6164	0.0026
NN (15,10,5)	30,000	2.619	2.5962	0.0026
NN (100,75,50)	50,000	2.2604	3.2272	0.0029
NN (75,50,25)	50,000	2.4708	2.7257	0.0028
NN (50,35,25)	50,000	2.5367	2.6943	0.0028
NN (35,25,20)	50,000	2.5768	2.634	0.0029
NN (25,20,15)	50,000	2.6049	2.6244	0.0029
NN (20,15,10)	50,000	2.6146	2.6135	0.0029
NN (15,10,5)	50,000	2.6316	2.5788	0.0028

Table A.13: The testing loss, learning loss, and average fitted mean of Poisson regression model and the Poisson neural network models with (100,75,50), (75,50,25), (50,35,25), (35,25,20), (25,20,15), (20,15,10), and (15,10,5) architectures for 1000 epochs, and batch sizes 10,000, 30,000, 50,000, 75,000, 100,000, 175,000, 250,000, 500,000, and 750,000.

Model	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
NN (100,75,50)	75,000	2.4316	2.8451	0.0031
NN (75,50,25)	75,000	2.5442	2.6493	0.0028
NN (50,35,25)	75,000	2.5761	2.635	0.0029
NN (35,25,20)	75,000	2.6098	2.601	0.0029
NN (25,20,15)	75,000	2.6376	2.5731	0.0027
NN (20,15,10)	75,000	2.638	2.5725	0.0027
NN (15,10,5)	75,000	2.6443	2.5701	0.0027
NN (100,75,50)	100,000	2.5945	2.6077	0.0029
NN (75,50,25)	100,000	2.6057	2.5973	0.0029
NN (50,35,25)	100,000	2.6376	2.564	0.0027
NN (35,25,20)	100,000	2.6492	2.545	0.0027
NN (25,20,15)	100,000	2.6514	2.5559	0.0028
NN (20,15,10)	100,000	2.6485	2.5516	0.0027
NN (15,10,5)	100,000	2.6551	2.548	0.0026
NN (100,75,50)	175,000	2.6591	**2.5389**	0.0028
NN (75,50,25)	175,000	2.6635	2.5484	0.0028
NN (50,35,25)	175,000	2.6666	2.5642	0.003
NN (35,25,20)	175,000	2.6725	2.5597	0.0029
NN (25,20,15)	175,000	2.6613	2.5418	0.0028
NN (20,15,10)	175,000	2.6662	2.5525	0.0028
NN (15,10,5)	175,000	2.665	2.5492	0.0028
NN (100,75,50)	250,000	2.6743	2.5565	0.0027
NN (75,50,25)	250,000	2.6757	2.5581	0.0028
NN (50,35,25)	250,000	2.6754	2.5542	0.0028
NN (35,25,20)	250,000	2.6777	2.5579	0.0028
NN (25,20,15)	250,000	2.6767	2.5592	0.0028
NN (20,15,10)	250,000	2.6836	2.5612	0.0028
NN (15,10,5)	250,000	2.7349	2.6092	0.0032
NN (100,75,50)	500,000	2.6823	2.5558	0.0028
NN (75,50,25)	500,000	2.6871	2.5631	0.0028
NN (50,35,25)	500,000	2.6877	2.5603	0.0028
NN (35,25,20)	500,000	2.69	2.564	0.0028
NN (25,20,15)	500,000	2.6934	2.5642	0.0028
NN (20,15,10)	500,000	2.7624	2.6288	0.0029
NN (15,10,5)	500,000	2.8025	2.6763	0.0029
NN (100,75,50)	750,000	2.6825	2.5552	0.0028
NN (75,50,25)	750,000	2.6892	2.5605	0.0028
NN (50,35,25)	750,000	2.6889	2.56	0.0028
NN (35,25,20)	750,000	2.7007	2.5684	0.0028
NN (25,20,15)	750,000	2.7646	2.6296	0.003
NN (20,15,10)	750,000	2.7947	2.6707	0.0033
NN (15,10,5)	750,000	2.9142	2.8013	0.0051

Table A.13: *Cont.*

Model	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
Data				0.0027
Pois.GLM		2.6811	2.5516	0.0027
NN (100,75,50)	100	2.6866	2.563	0.0031
NN (75,50,25)	100	2.6836	2.5606	0.003
NN (50,35,25)	100	2.6869	2.5649	0.0032
NN (35,25,20)	100	2.6851	2.5611	0.0028
NN (25,20,15)	100	2.6917	2.568	0.0028
NN (20,15,10)	100	2.696	2.5687	0.0028
NN (15,10,5)	100	2.8045	2.6778	0.0029
NN (100,75,50)	250	2.6518	2.5527	0.0026
NN (75,50,25)	250	2.6589	2.5448	0.0028
NN (50,35,25)	250	2.6599	2.5454	0.0026
NN (35,25,20)	250	2.6615	**2.5379**	0.0028
NN (25,20,15)	250	2.6605	2.5404	0.0026
NN (20,15,10)	250	2.6627	2.541	0.0024
NN (15,10,5)	250	2.6618	2.5396	0.0025
NN (100,75,50)	500	2.4575	2.7481	0.0029
NN (75,50,25)	500	2.5440	2.6737	0.0028
NN (50,35,25)	500	2.6074	2.5824	0.0027
NN (35,25,20)	500	2.6256	2.5703	0.0028
NN (25,20,15)	500	2.6288	2.5703	0.0027
NN (20,15,10)	500	2.6452	2.5683	0.0028
NN (15,10,5)	500	2.6459	2.5569	0.0028
NN (100,75,50)	1000	2.1653	3.4808	0.0028
NN (75,50,25)	1000	2.3453	3.0112	0.0029
NN (50,35,25)	1000	2.4357	2.8469	0.0027
NN (35,25,20)	1000	2.5345	2.7145	0.0027
NN (25,20,15)	1000	2.5661	2.6647	0.0027
NN (20,15,10)	1000	2.5983	2.5962	0.0027
NN (15,10,5)	1000	2.6194	2.5996	0.0028
NN (100,75,50)	1500	2.1123	4.2328	0.0029
NN (75,50,25)	1500	2.2884	3.4217	0.003
NN (50,35,25)	1500	2.3753	2.9564	0.0029
NN (35,25,20)	1500	2.4651	2.8431	0.0029
NN (25,20,15)	1500	2.5323	2.7397	0.0028
NN (20,15,10)	1500	2.5565	2.6925	0.0028
NN (15,10,5)	1500	2.6032	2.6234	0.0027

Table A.14: The testing loss, learning loss, and average fitted mean of the Poisson neural network models with different architectures for batch size 30,000 and different choices of epochs.

Model	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
NN (100,75,50)	2000	2.105	4.8078	0.0029
NN (75,50,25)	2000	2.2302	3.631	0.0029
NN (50,35,25)	2000	2.3314	3.1311	0.0029
NN (35,25,20)	2000	2.4312	2.923	0.0028
NN (25,20,15)	2000	2.4837	2.764	0.0027
NN (20,15,10)	2000	2.5611	2.6882	0.0026
NN (15,10,5)	2000	2.5997	2.6522	0.0027

Table A.14: *Cont.*

Model	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
Data				0.0027
Pois.GLM		2.6811	2.5516	0.0027
NN (75,50,25)	10,000	2.5453	2.6831	0.0031
NN (50,35,25)	10,000	2.5917	2.6302	0.003
NN (35,25,20)	10,000	2.6184	2.6045	0.003
NN (25,20,15)	10,000	2.6305	2.5834	0.0029
NN (20,15,10)	10,000	2.631	2.574	0.0029
NN (15,10,5)	10,000	2.6469	2.561	0.0028
NN (75,50,25)	30,000	2.6565	2.5493	0.0028
NN (50,35,25)	30,000	2.6546	2.5515	0.0027
NN (35,25,20)	30,000	2.6594	2.5442	0.0027
NN (25,20,15)	30,000	2.6594	2.5463	0.0027
NN (20,15,10)	30,000	2.6602	**2.5402**	0.0025
NN (15,10,5)	30,000	2.6657	2.568	0.0025
NN (75,50,25)	50,000	2.6751	2.5518	0.0027
NN (50,35,25)	50,000	2.674	2.5484	0.0027
NN (35,25,20)	50,000	2.6748	2.5572	0.0026
NN (25,20,15)	50,000	2.6758	2.5533	0.0026
NN (20,15,10)	50,000	2.6786	2.556	0.0026
NN (15,10,5)	50,000	2.6824	2.5618	0.0027
NN (75,50,25)	75,000	2.6824	2.5575	0.0027
NN (50,35,25)	75,000	2.6773	2.5551	0.0027
NN (35,25,20)	75,000	2.6851	2.5633	0.0027
NN (25,20,15)	75,000	2.6857	2.5626	0.0028
NN (20,15,10)	75,000	2.6844	2.5642	0.0027
NN (15,10,5)	75,000	2.7231	2.5933	0.003

Table A.15: The testing loss, learning loss, and average fitted mean of the Poisson neural network models with different architectures for 10,000, 30,000, 50,000, 75,000, 100,000, 175,000, 250,000, 500,000 and 750,000 batch sizes and 250 epochs.

Model	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
NN (75,50,25)	100,000	2.6839	2.5603	0.0027
NN (50,35,25)	100,000	2.6831	2.5586	0.0027
NN (35,25,20)	100,000	2.69	2.5652	0.0028
NN (25,20,15)	100,000	2.6877	2.5642	0.0028
NN (20,15,10)	100,000	2.7066	2.5724	0.0029
NN (15,10,5)	100,000	2.7639	2.6299	0.003
NN (75,50,25)	175,000	2.686	2.5582	0.0028
NN (50,35,25)	175,000	2.6912	2.5579	0.0028
NN (35,25,20)	175,000	2.6896	2.5591	0.0028
NN (25,20,15)	175,000	2.7061	2.5734	0.0028
NN (20,15,10)	175,000	2.7658	2.6331	0.0029
NN (15,10,5)	175,000	2.9795	2.8704	0.0059
NN (75,50,25)	250,000	2.6985	2.566	0.0028
NN (50,35,25)	250,000	2.702	2.5674	0.0028
NN (35,25,20)	250,000	2.7312	2.5956	0.003
NN (25,20,15)	250,000	2.7579	2.6222	0.003
NN (20,15,10)	250,000	2.8054	2.6816	0.0033
NN (15,10,5)	250,000	2.8716	2.7557	0.0046
NN (75,50,25)	500,000	2.721	2.5869	0.0029
NN (50,35,25)	500,000	2.7048	2.5666	0.0028
NN (35,25,20)	500,000	2.7868	2.6548	0.003
NN (25,20,15)	500,000	2.7821	2.6511	0.0031
NN (20,15,10)	500,000	2.8502	2.7323	0.0042
NN (15,10,5)	500,000	3.7937	3.7045	0.0131
NN (75,50,25)	750,000	2.7237	2.5909	0.003
NN (50,35,25)	750,000	2.7492	2.6152	0.0029
NN (35,25,20)	750,000	2.754	2.6209	0.0032
NN (25,20,15)	750,000	2.8334	2.7084	0.0039
NN (20,15,10)	750,000	2.8275	2.7067	0.0041
NN (15,10,5)	750,000	6.4324	6.3651	0.0317

Table A.15: *Cont.*

```
#####Feature pre-processing for regression ----------------------------
dat2 <- data
# Exposure: append the log of Exposure
dat2$logExp<- log(dat2$Exposure)

#Setting data type right for categorical variables and continous variable
dat2<-dat2 %>% mutate_at(vars(PLANTYP,UR,EGEOLOC,EECLASS,EESTATU,EMPREL,SEX),as.
    factor) %>%
                mutate_at(vars(AGE,AdmnNb),as.numeric) %>% data.frame
dat2$SEXRT <- as.integer(dat2$SEX)
dat2$URRT <- as.integer(dat2$UR)
```

```
str(dat2)

# create a loss table to record the deviance loss and the profolio average in different sets
loss.table <- data.frame(row.names = c("learning", "testing", "ptfl_avrg_learn", "ptfl_avrg_test", "ptfl_avrg"))

######Function definition for regression -------------------------------
######Poisson density definition ---------------------------------------
# Poisson density function for deviance loss calculation

dpois0 <- function(y, mu) {
  d <- rep(1, length(y))
  d[mu != 0] <- dpois(y[mu != 0], lambda = mu[mu != 0])
  d
}

######Generalized deviance loss ----------------------------------------

dev.loss <- function(y, mu, density.func = dpois0) {
  if (length(mu) != length(y)) stop("mu should have the same length as y.")
  logL.tilde <- mean(log(density.func(y, y)))
  logL.hat <- mean(log(density.func(y, mu)))
  2 * (logL.tilde - logL.hat)
}

######Record function definition ---------------------------------------
# record the predictive performance for a model (the deviance loss) and average fitted mean

result.record <- function(learn, test, loss.table, colname, density.func = dpois0){
                           # originally dnbi.gamlss
  loss.table[[colname]][1] <- 100 * dev.loss(y = learn$AdmnNb, mu = learn[[colname]], density.func) # dev.loss in learning set
  loss.table[[colname]][2] <- 100 * dev.loss(y = test$AdmnNb, mu = test[[colname]], density.func) # dev.loss in test set

  loss.table[[colname]][3] <- mean(learn[[colname]]) / mean(learn$Exposure) # average fitted mean in learning set
  loss.table[[colname]][4] <- mean(test[[colname]]) / mean(test$Exposure) # average fitted mean in test set
  loss.table[[colname]][5] <- mean(c(learn[[colname]], test[[colname]])) / mean(c(learn$Exposure, test$Exposure)) # average fitted mean in whole set
  loss.table
}
```

```
47
48 ######Choose learning and test sample ---------------------------------
49 seed<-100
50 set.seed(seed)
51 # Split the sample, 90% as the learning set, 10% as the test set
52 ll <- sample(c(1:nrow(dat2)), round(0.9*nrow(dat2)), replace = FALSE) # learning
       sample index
53 learn <- dat2[ll,] # learning sample
54 test <- dat2[-ll,] # testing sample
55
56 n_l <- nrow(learn) # length of the learning set
57 n_t <- nrow(test) # length of the testing set
58 ######Data preparation for neural network ------------------------------
59 ######Feature pre-processing for neural network ------------------------
60
61 # Define Min-Max scalar to rescale the continous predictors between [-1, 1]
62 PreProcess.Continuous <- function(data) 2*(data-min(data))/(max(data)-min(data))-1
63
64 dat3 <- data.frame(AdmnNb = dat2$AdmnNb)
65 dat3$AGE <- PreProcess.Continuous(dat2$AGE)
66 #For Binary levels
67 dat3$SEX <- PreProcess.Continuous(dat2$SEXRT)
68 dat3$UR <- PreProcess.Continuous(dat2$URRT)
69
70 ######Choose learning and test sample ---------------------------------
71 # divided into learning and testing set
72
73 learn2 <- dat3[ll,] # learning set
74 test2 <- dat3[-ll,] # testing set
75
76 ######Data structure set-up for neural network ------------------------
77 # Continuous predictors (1 in total)
78
79 Design.learn <- matrix(as.numeric(as.matrix(learn2[,c(2:ncol(learn2))])), nrow =
       n_l, ncol = ncol(learn2) - 1)
80 Design.test <- matrix(as.numeric(as.matrix(test2[,c(2:ncol(test2))])), nrow = n_t,
       ncol = ncol(test2) - 1)
81
82 AGE.learn <- as.matrix(learn2$AGE)
83 AGE.test <- as.matrix(test2$AGE)
84
85 # EECLASS(categorical predictor)
86 Eec.learn <- as.matrix(as.integer(learn$EECLASS)-1)
87 Eec.test <- as.matrix(as.integer(test$EECLASS)-1)
88 # EESTATU (categorical predictor)
```

```
Ees.learn <- as.matrix(as.integer(learn$EESTATU)-1)
Ees.test <- as.matrix(as.integer(test$EESTATU)-1)
# Emprel (categorical predictor)
Emp.learn <- as.matrix(as.integer(learn$EMPREL)-1)
Emp.test <- as.matrix(as.integer(test$EMPREL)-1)
# PLANTYP (categorical predictor)
Pln.learn <- as.matrix(as.integer(learn$PLANTYP)-1)
Pln.test <- as.matrix(as.integer(test$PLANTYP)-1)
# EGEOLOC (categorical predictor)
Ege.learn <- as.matrix(as.integer(learn$EGEOLOC)-1)
Ege.test <- as.matrix(as.integer(test$EGEOLOC)-1)

# Exposure
Vol.learn <- as.matrix(learn$Exposure)
Vol.test <- as.matrix(test$Exposure)
LogVol.learn <- as.matrix(learn$logExp)
LogVol.test <- as.matrix(test$logExp)
LogVol_EDF.learn <- as.matrix(cbind(learn$logExp, rep(0, n_l)))
LogVol_EDF.test <- as.matrix(cbind(test$logExp, rep(0, n_t)))

# Admn number (response)
Ylearn <- as.matrix(learn2$AdmnNb)
Ytest <- as.matrix(test2$AdmnNb)
```

Listing A.1: Code for data preparation, feature pre-processing and other supporting functions.

```
#Loading additional libraries
library(MASS)
library(keras)
library(tensorflow)
library(stats)
library(gamlss)

seed<-100
set.seed(seed)
Sys.setenv(PYTHONHASHSEED = seed)
reticulate::py_set_seed(seed)
tensorflow::tf$random$set_seed(seed)
set_random_seed(seed)
######Neural Network ------------------------------------------------------
######Hyperparameters initialization ---------------------------------------

q1 <- c(20) # the dimension of the 1st hidden layer
q2 <- c(15) # the dimension of the 2nd hidden layer
```

```
q3 <- c(10) # the dimension of the 3rd hidden layer
epochs <- c(250) # number of epochs to train the model
batchsize <- c(30000) # number of samples per gradient update
qEmb <- c(2)# the dimension of the embedded layer for categorical variable

######Input layer ------------------------------------------------------
# Input layer for continuous features

DesignShape <- ncol(learn2) - 1 # the number of the continuous predictors
Design <- layer_input(shape = DesignShape, dtype = 'float32', name = 'Design')

# Input layer for categorical features

EecLabel<-length(unique(learn$EECLASS)) # number of eeclass = 9
EesLabel<-length(unique(learn$EESTATU)) # number of eestatu = 9
EmpLabel<-length(unique(learn$EMPREL)) # number of emprel = 3
PlnLabel<-length(unique(learn$PLANTYP)) # number of PLANTYP = 8
EgeLabel<-length(unique(learn$EGEOLOC)) # number of EGEOLOC = 53

EECLASS<-layer_input(shape=c(1), dtype = 'int32', name = 'EECLASS')
EESTATU<-layer_input(shape=c(1), dtype = 'int32', name = 'EESTATU')
EMPREL<-layer_input(shape=c(1), dtype = 'int32', name = 'EMPREL')
PLANTYP<-layer_input(shape=c(1), dtype = 'int32', name = 'PLANTYP')
EGEOLOC<-layer_input(shape=c(1), dtype = 'int32', name = 'EGEOLOC')

# Input layer for Exposure (as the offset)
LogVol<-layer_input(shape=c(1), dtype = 'float32', name = 'LogVol')
Vol<-layer_input(shape=c(1), dtype = 'float32', name = 'Vol')

#######Embedding layer ------------------------------------------------
#Create embedding layer for categorical predictors (dimension: qEmb=2)

EecEmb = EECLASS %>%
  layer_embedding(input_dim = EecLabel, output_dim = qEmb, input_length = 1, name =
       'EecEmb') %>%
  layer_flatten(name = 'Eec_flat')
EesEmb = EESTATU %>%
  layer_embedding(input_dim = EesLabel, output_dim = qEmb, input_length = 1, name =
       'EesEmb') %>%
  layer_flatten(name = 'Ees_flat')
EmpEmb = EMPREL %>%
  layer_embedding(input_dim = EmpLabel, output_dim = qEmb, input_length = 1, name =
       'EmpEmb') %>%
  layer_flatten(name = 'Emp_flat')
PlnEmb = PLANTYP %>%
```

```
  layer_embedding(input_dim = PlnLabel, output_dim = qEmb, input_length = 1, name =
      'PlnEmb') %>%
  layer_flatten(name = 'Pln_flat')
EgeEmb = EGEOLOC %>%
  layer_embedding(input_dim = EgeLabel, output_dim = qEmb, input_length = 1, name =
      'EgeEmb') %>%
  layer_flatten(name = 'Ege_flat')

######Main architecture and Output layer
    ----------------------------------------------------
# Main architecture with 3 hidden layers

Network <- list(Design, EecEmb, EesEmb, EmpEmb, PlnEmb,EgeEmb) %>%
    layer_concatenate(name = 'concate') %>%
  # 1st hidden layer
  layer_dense(units = q1, activation = 'tanh', name = 'hidden1') %>%
  # 2nd hidden layer
  layer_dense(units = q2, activation = 'tanh', name = 'hidden2') %>%
  # 3rd hidden layer
  layer_dense(units = q3, activation = 'tanh', name = 'hidden3') %>%
  # provide one neuron in the output layer
  layer_dense(units = 1, activation = 'linear', name = 'Network')

#Output layer to combine the main architecture and the offset layer
Response = list(Network, LogVol) %>%
  # add the exposure and the last neuron
  layer_add(name = 'Add') %>%
  # give the response
  layer_dense(units = 1,
              activation = 'exponential',
              name = 'Response', trainable = FALSE,
              weights = list(array(1, dim = c(1,1)), array(0, dim = c(1))))

#######Model configuration and fitting
    ----------------------------------------------------
# Model assembly

model <- keras_model(inputs = c(Design, EECLASS, EESTATU, EMPREL, PLANTYP,EGEOLOC,
    LogVol), outputs = c(Response))
summary(model)
# Model configuration

model %>% compile(
  loss = 'poisson', # set poisson deviance loss function as the objective loss
      function
```

```
  optimizer = 'nadam')

# Model fitting by running gradient descent method to minimize the objective loss function
{
  t1 <- proc.time()
  fit <- model %>% fit(
    list(Design.learn, Eec.learn, Ees.learn, Emp.learn, Pln.learn, Ege.learn, LogVol.learn), # all predictors
    Ylearn, # response
    verbose = 1, # verbose = 0 silences the progress bar for the process
    # verbose = 1 shows the fitting process, incl. learning loss and validation loss, epoch by epoch
    epochs = epochs, # epochs = 1,000
    batch_size = batchsize, # batchsize = 10,000
    validation_split = 0.2 # 20% as validation set
    )
    print(proc.time()-t1)
}
# Predicted value of the admission numbers
learn$nn0 <- as.vector(model %>% predict(list(Design.learn, Eec.learn, Ees.learn, Emp.learn, Pln.learn,Ege.learn, LogVol.learn)))
test$nn0 <- as.vector(model %>% predict(list(Design.test, Eec.test, Ees.test, Emp.test, Pln.test,Ege.test, LogVol.test)))

# Record the loss in both learning and testing set
loss.table <- result.record(learn, test, loss.table, "nn0", dpois0)
```

Listing A.2: code for fitting a sample Poisson neural network model with architecture (20,15,10) model with batch size 30,000 and epochs 250

Algorithm A.1 Pseudo code for neural network.

Step 0: Read data

Step 1: Data preparation and feature pre-processing

// setting data types and structure right, min- max scaler transformation for continuous and binary variables//

Step 2: Choose learning and test sample

  input: seed value

  learning data← 90% of data randomly sampled

  testing data← 10% remaining data

Step 3: Constructing neural network

initialise hyper parameters

    input: batch size, epochs, number of hidden layers

    input: number of nodes in each layer

    input: validation split // 20%//

create the initial layer

    input: dimension of the embedded layer for categorical variable

for all features in the dataset **do**

    create input layer

end for

for all categorical features in the dataset **do**

    create embedding layer

end for

outcome: initial layer

create main architecture and output node

for $i = 1$ to number of hidden layers **do**

    input: activation function

    create a layer with the required number of nodes in i^{th} layer

end for

create offset ← log(exposure)

create output node to combine the main architecture and the offset

  set activation function for output layer ← exponential

Step 4: Model assembly and configuration

assemble the model defining inputs and outputs

  set inputs ← input layer of all features, offset

  set outputs ← output node

configure the model // compile//

  input: loss function ← 'poisson'

  input: optimizer ← 'nadam'

Algorithm A.1 *Cont.*

Step 5: Model fitting
split learning data into training and validation set
  training set← 80% of *'learning data'*
  validation set← 20% *'learning data'*
no. of batches ← rounded to next integer$\left(\frac{\text{(no. of records in learn}\times\text{(1-validation split))}}{\text{batch size}}\right)$
split *'training set'* into *'no. of batches'* batches
// last batch might contain fewer samples//

fitting model
initialize model weights w randomly
for each epoch **do**
  for each batch of data **do**
    update the model weights using 'nadam' optimizer
  end for
  calculate validation loss
end for
return model output// fitted values for the testing data//

Step 6: Model evaluation
predict for testing data and **calculate** testing loss

```
# Negative binomial regression using gamlss
NBI.reg <- gamlss(AdmnNb ~ offset(logExp) + AGE + EGEOLOC + SEXRT + URRT + EECLASS
    + EESTATU + EMPREL + PLANTYP, data = learn, family = NBI)

learn$NBI.reg <- fitted(NBI.reg)
test$NBI.reg <- predict(NBI.reg, newdata = test, type="response")

#replacing the offset term with working weight obtained from regression
LogVol.learn <- as.matrix(log(learn$NBI.reg))
LogVol.test <- as.matrix(log(test$NBI.reg))
 model <- keras_model(inputs = c(Design, EECLASS, EESTATU, EMPREL, PLANTYP,EGEOLOC,
      LogVol), outputs = c(Response))
```

Listing A.3: Code for implementing $CANN_{NB}$.

```
# Main architecture with three hidden layers
Network <- list(Design, EecEmb, EesEmb, EmpEmb, PlnEmb,EgeEmb) %>%
    layer_concatenate(name = 'concate') %>%
  layer_dense(units = q1, kernel_regularizer = regularizer_l2(0.00001),activation =
       'tanh', name = 'hidden1') %>%
  layer_dense(units = q2, kernel_regularizer = regularizer_l2(0.00001),activation =
       'tanh', name = 'hidden2') %>%
```

```
  layer_dense(units = q3, kernel_regularizer = regularizer_l2(0.00001),activation =
      'tanh', name = 'hidden3') %>%
  layer_dense(units = 1, activation = 'linear', name = 'Network')
```

Listing A.4: Code for implementing ridge regularisation with $\eta = 10^{-5}$.

```
CBs<-callback_model_checkpoint("path0",
                               monitor = "val_loss",
                               save_best_only = TRUE,
                               verbose = 1,
                               save_weights_only = TRUE)
fit <- model %>% fit(
            list(Design.learn, Eec.learn, Ees.learn, Emp.learn, Pln.learn, Ege.
                learn, LogVol.learn), # all predictors
            Ylearn,
            verbose = 1,
            epochs = epochs,
            batch_size = batchsize,
            validation_split = 0.2,
            callbacks = CBs)
load_model_weights_hdf5(model,"path0")
```

Listing A.5: Code for implementing early stopping using callback

```
p<-0.02
Network <- list(Design, EecEmb, EesEmb, EmpEmb, PlnEmb,EgeEmb) %>%
    layer_concatenate(name = 'concate') %>%
  layer_dense(units = q1,activation = 'tanh', name = 'hidden1') %>% layer_dropout(
      rate = p) %>%
  layer_dense(units = q2,activation = 'tanh', name = 'hidden2') %>% layer_dropout(
      rate = p) %>%
  layer_dense(units = q3,activation = 'tanh', name = 'hidden3') %>% layer_dropout(
      rate = p) %>%
  layer_dense(units = 1, activation = 'linear', name = 'Network')
```

Listing A.6: Code for implementing dropout with dropout rate $p = 2\%$.

```
glm.fmla <- function(qq){
  string <- "AdmnNb ~ z1"
  if (qq>1){for (z in 2:qq){ string <- paste(string, "+z",z, sep="")}}
  string
}

zz <- keras_model(inputs=model$input, outputs=get_layer(model, 'hidden3')$output)
zz.learn <- data.frame(zz %>% predict(list(Design.learn, Eec.learn, Ees.learn, Emp.
    learn, Pln.learn,Ege.learn, LogVol.learn)))
```

```
colnames(zz.learn)<-c(paste0("z",1:q3))
zz.learn$AdmnNb <- learn$AdmnNb
zz.learn$Exposure <- learn$Exposure
zz.test <- data.frame(zz %>%predict(list(Design.test, Eec.test, Ees.test, Emp.test,
     Pln.test,Ege.test, LogVol.test)))
colnames(zz.test)<-c(paste0("z",1:q3))
zz.test$Exposure<-test$Exposure
# perform GLM step on the last hidden layer
glm1 <- glm(as.formula(glm.fmla(q3)), data=zz.learn,offset = log(Exposure), family=
    poisson())

nn<-paste("nnreg",batchsize,epochs,q1,q2,q3,sep="_")
# Predicted value of the admn numbers
learn[,nn] <- fitted(glm1)
test[,nn] <- predict(glm1, newdata=zz.test, type="response")
# Record the loss in both learning and testing set
loss.table <- result.record(learn, test, loss.table, nn, dpois0)
```

Listing A.7: Code for implementing the GLM bias regularisation approach for Poisson neural network model.

Algorithm A.2 Pseudo code for bias regularisation.

Step 0: Read data
Step 1: Data preparation and feature pre-processing
Step 2: Choose learning and test sample
Step 3: Constructing neural network
Step 4: Model assembly and configuration
Step 5: Model fitting
// no change till this point//

Step 6: bias regularisation
$q \leftarrow$ number of nodes in the last hidden layer
$z_{1,..,q} \leftarrow$ output from the last hidden layer
if neural network model **then**
    bias regularised model$\leftarrow$ fit GLM to $(\boldsymbol{z})$// offset=log(exposure)//
else if CANN **then**
    bias regularised model$\leftarrow$ fit GLM to $(\boldsymbol{z})$ // *offset* $\leftarrow$ *(Pois.glm)* // // *(Pois.glm)* is the GLM model used in CANN//
end if
Step 7: Model evaluation
predict for testing data using *bias regularised model*
calculate testing loss

Algorithm A.3 Pseudo code for nagging predictor.

Step 0: Read data
Step 1: Data preparation and feature pre-processing
Step 2: Choose learning and test sample
Step 3: Constructing neural network
Step 4: Model assembly and configuration

for $i = 1$ to index (M) of nagging predictor **do**
  set seed value // different seed value for each iteration//

  Step 5: Model fitting
  split learning data into training and validation set
    training set← 80% of *'learning data'*
    validation set← 20% *'learning data'*
  no. of batches ← rounded to next integer$\left(\frac{\text{(no. of records in learn}\times\text{(1-validation split))}}{\text{batch size}}\right)$
  split *'training set'* into *'no. of batches'* batches
// last batch might contain fewer samples//

  fitting model
  initialize model weights w randomly
  for each epoch **do**
    for each batch of data **do**
      update the model weights using 'nadam' optimizer
    end for
    calculate validation loss
  end for
  return model output // fitted values for learning data//

  Step 6: Model evaluation
  $predictor_i$ ←**predict** for testing data
end for
$nagging\ predictor_M \leftarrow avg(predictor_{j=1\ldots M})$
calculate testing loss using $nagging\ predictor_M$

Model	**Epochs**	**Batch Size**	**Learning Loss**	**Testing Loss**	**Average Fitted Mean**
Pois.reg			2.4666	2.5260	0.0025
NN (50,35,25)	100	10,000	2.4458	2.5110	0.0025
NN (35,25,20)	100	10,000	2.4478	2.5110	0.0026
NN (25,20,15)	100	10,000	2.4464	2.5145	0.0025
NN (20,15,10)	100	10,000	2.4441	2.5108	0.0025
NN (15,10,5)	100	10,000	2.4467	2.5109	0.0026
NN (50,35,25)	100	30,000	2.4459	2.5107	0.0025
NN (35,25,20)	100	30,000	2.4467	2.5134	0.0025
NN (25,20,15)	100	30,000	2.4481	2.5124	0.0025
NN (20,15,10)	100	30,000	2.4472	2.5115	0.0026
NN (15,10,5)	100	30,000	2.4493	2.5117	0.0024
NN (50,35,25)	100	50,000	2.4478	2.5127	0.0024
NN (35,25,20)	100	50,000	2.4496	2.5151	0.0025
NN (25,20,15)	100	50,000	2.4513	2.5148	0.0024
NN (20,15,10)	100	50,000	2.4511	2.5143	0.0023
NN (15,10,5)	100	50,000	2.4535	2.5184	0.0024
NN (50,35,25)	100	75,000	2.4578	2.5223	0.0024
NN (35,25,20)	100	75,000	2.4628	2.5260	0.0024
NN (25,20,15)	100	75,000	2.4668	2.5285	0.0024
NN (20,15,10)	100	75,000	2.4678	2.5309	0.0024
NN (15,10,5)	100	75,000	2.4721	2.5371	0.0023
NN (50,35,25)	100	100,000	2.4660	2.5276	0.0024
NN (35,25,20)	100	100,000	2.4679	2.5302	0.0024
NN (25,20,15)	100	100,000	2.4705	2.5335	0.0024
NN (20,15,10)	100	100,000	2.4753	2.5397	0.0023
NN (15,10,5)	100	100,000	2.4899	2.5571	0.0024
NN (50,35,25)	250	10,000	2.4458	2.5110	0.0025
NN (35,25,20)	250	10,000	2.4478	2.5110	0.0026
NN (25,20,15)	250	10,000	2.4464	2.5145	0.0025
NN (20,15,10)	250	10,000	2.4441	2.5108	0.0025
NN (15,10,5)	250	10,000	2.4467	2.5109	0.0026
NN (50,35,25)	250	30,000	2.4459	2.5107	0.0025
NN (35,25,20)	250	30,000	2.4426	2.5117	0.0026
NN (25,20,15)	250	30,000	2.4437	2.5114	0.0025
NN (20,15,10)	250	30,000	2.4446	2.5107	0.0025
NN (15,10,5)	250	30,000	2.4463	2.5110	0.0026

Table A.16: The testing loss, learning loss, and average fitted mean of the Poisson regression model and the Poisson neural network models with different architectures, as well as for different combinations of batch size and epoch, in the case of individual-level data for admissions due to respiratory diseases in years 2016-2019.

Model	Epochs	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
NN (50,35,25)	250	50,000	2.4463	2.5106	0.0025
NN (35,25,20)	250	50,000	2.4462	2.5115	0.0026
NN (25,20,15)	250	50,000	2.4449	2.5113	0.0026
NN (20,15,10)	250	50,000	2.4460	2.5094	0.0026
NN (15,10,5)	250	50,000	2.4471	2.5115	0.0025
NN (50,35,25)	250	75,000	2.4434	2.5105	0.0026
NN (35,25,20)	250	75,000	2.4441	2.5109	0.0026
NN (25,20,15)	250	75,000	2.4458	2.5126	0.0025
NN (20,15,10)	250	75,000	2.4464	**2.5089**	0.0026
NN (15,10,5)	250	75,000	2.4478	2.5111	0.0025
NN (50,35,25)	250	100,000	2.4457	2.5099	0.0026
NN (35,25,20)	250	100,000	2.4459	2.5124	0.0025
NN (25,20,15)	250	100,000	2.4475	2.5130	0.0025
NN (20,15,10)	250	100,000	2.4474	2.5100	0.0025
NN (15,10,5)	250	100,000	2.4493	2.5128	0.0025
NN (50,35,25)	500	10,000	2.4458	2.5110	0.0025
NN (35,25,20)	500	10,000	2.4478	2.5110	0.0026
NN (25,20,15)	500	10,000	2.4464	2.5145	0.0025
NN (20,15,10)	500	10,000	2.4441	2.5108	0.0025
NN (15,10,5)	500	10,000	2.4467	2.5109	0.0026
NN (50,35,25)	500	30,000	2.4459	2.5107	0.0025
NN (35,25,20)	500	30,000	2.4426	2.5117	0.0026
NN (25,20,15)	500	30,000	2.4437	2.5114	0.0025
NN (20,15,10)	500	30,000	2.4446	2.5107	0.0025
NN (15,10,5)	500	30,000	2.4463	2.5110	0.0026
NN (50,35,25)	500	50,000	2.4463	2.5106	0.0025
NN (35,25,20)	500	50,000	2.4462	2.5115	0.0026
NN (25,20,15)	500	50,000	2.4449	2.5113	0.0026
NN (20,15,10)	500	50,000	2.4460	2.5094	0.0026
NN (15,10,5)	500	50,000	2.4471	2.5115	0.0025
NN (50,35,25)	500	75,000	2.4434	2.5105	0.0026
NN (35,25,20)	500	75,000	2.4441	2.5109	0.0026
NN (25,20,15)	500	75,000	2.4444	2.5113	0.0026
NN (20,15,10)	500	75,000	2.4464	2.5089	0.0026
NN (15,10,5)	500	75,000	2.4468	2.5110	0.0025
NN (50,35,25)	500	100,000	2.4457	2.5099	0.0026
NN (35,25,20)	500	100,000	2.4437	2.5117	0.0026
NN (25,20,15)	500	100,000	2.4423	2.5096	0.0025
NN (20,15,10)	500	100,000	2.4461	2.5095	0.0025
NN (15,10,5)	500	100,000	2.4472	2.5114	0.0025

Table A.16: *Cont.*

Model	**Epochs**	**Batch Size**	**Learning Loss**	**Testing Loss**	**Average Fitted Mean**
Pois.reg			3.0656	2.9932	0.0032
NN (35,25,20)	100	10,000	3.0095	2.9668	0.0033
NN (25,20,15)	100	10,000	3.0087	2.9661	0.0031
NN (20,15,10)	100	10,000	3.0105	2.9651	0.0032
NN (15,10,5)	100	10,000	3.0104	2.9630	0.0032
NN (35,25,20)	100	30,000	3.0108	2.9620	0.0032
NN (25,20,15)	100	30,000	3.0124	2.9603	0.0032
NN (20,15,10)	100	30,000	3.0137	2.9628	0.0032
NN (15,10,5)	100	30,000	3.0179	2.9622	0.0031
NN (35,25,20)	100	50,000	3.0147	2.9610	0.0033
NN (25,20,15)	100	50,000	3.0234	2.9663	0.0032
NN (20,15,10)	100	50,000	3.0388	2.9757	0.0031
NN (15,10,5)	100	50,000	3.1329	3.0558	0.0031
NN (35,25,20)	100	75,000	3.0510	2.9836	0.0031
NN (25,20,15)	100	75,000	3.0679	2.9957	0.0031
NN (20,15,10)	100	75,000	3.1452	3.0706	0.0035
NN (15,10,5)	100	75,000	3.2551	3.1809	0.0031
NN (35,25,20)	250	10,000	3.0087	2.9637	0.0033
NN (25,20,15)	250	10,000	3.0087	2.9661	0.0031
NN (20,15,10)	250	10,000	3.0105	2.9651	0.0032
NN (15,10,5)	250	10,000	3.0091	2.9621	0.0032
NN (35,25,20)	250	30,000	3.0108	2.9620	0.0032
NN (25,20,15)	250	30,000	3.0087	2.9640	0.0032
NN (20,15,10)	250	30,000	3.0096	2.9630	0.0032
NN (15,10,5)	250	30,000	3.0100	2.9628	0.0032
NN (35,25,20)	250	50,000	3.0101	2.9612	0.0032
NN (25,20,15)	250	50,000	3.0118	2.9600	0.0033
NN (20,15,10)	250	50,000	3.0106	2.9623	0.0032
NN (15,10,5)	250	50,000	3.0117	2.9617	0.0032
NN (35,25,20)	250	75,000	3.0121	2.9617	0.0033
NN (25,20,15)	250	75,000	3.0119	**2.9591**	0.0032
NN (20,15,10)	250	75,000	3.0128	2.9619	0.0032
NN (15,10,5)	250	75,000	3.0174	2.9632	0.003
NN (35,25,20)	500	10,000	3.0087	2.9637	0.0033
NN (25,20,15)	500	10,000	3.0087	2.9661	0.0031
NN (20,15,10)	500	10,000	3.0105	2.9651	0.0032
NN (15,10,5)	500	10,000	3.0091	2.9621	0.0032

Table A.17: The testing loss, learning loss, and average fitted mean of the Poisson regression model and the Poisson neural network models with different architectures, as well as for different combinations of batch size and epoch, in the case of individual-level data for admissions due to neoplasms in males from the years 2016 to 2019.

Model	Epochs	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
NN (35,25,20)	500	30,000	3.0108	2.9620	0.0032
NN (25,20,15)	500	30,000	3.0087	2.9640	0.0032
NN (20,15,10)	500	30,000	3.0096	2.9630	0.0032
NN (15,10,5)	500	30,000	3.0100	2.9628	0.0032
NN (35,25,20)	500	50,000	3.0101	2.9612	0.0032
NN (25,20,15)	500	50,000	3.0087	2.9631	0.0032
NN (20,15,10)	500	50,000	3.0095	2.9626	0.0032
NN (15,10,5)	500	50,000	3.0096	2.9631	0.0032
NN (35,25,20)	500	75,000	3.0121	2.9617	0.0033
NN (25,20,15)	500	75,000	3.0118	2.9594	0.0032
NN (20,15,10)	500	75,000	3.0095	2.9627	0.0032
NN (15,10,5)	500	75,000	3.0104	2.9616	0.0032

Table A.17: *Cont.*

Model	Epochs	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg			3.9780	4.0140	0.0043
NN (35,25,20)	100	10,000	3.9000	3.9390	0.0042
NN (25,20,15)	100	10,000	3.8960	3.9390	0.0043
NN (20,15,10)	100	10,000	3.9010	3.9390	0.0043
NN (15,10,5)	100	10,000	3.8990	3.9390	0.0043
NN (35,25,20)	100	30,000	3.9050	3.9380	0.0041
NN (25,20,15)	100	30,000	3.9060	3.9380	0.0041
NN (20,15,10)	100	30,000	3.9070	3.9430	0.0043
NN (15,10,5)	100	30,000	3.9150	3.9480	0.0042
NN (35,25,20)	100	50,000	3.9120	3.9470	0.0042
NN (25,20,15)	100	50,000	3.9150	3.9520	0.0041
NN (20,15,10)	100	50,000	3.9390	3.9720	0.0041
NN (15,10,5)	100	50,000	4.0510	4.0600	0.0041
NN (35,25,20)	100	75,000	3.9250	3.9590	0.0041
NN (25,20,15)	100	75,000	3.9350	3.9740	0.0043
NN (20,15,10)	100	75,000	4.0430	4.0520	0.0042
NN (15,10,5)	100	75,000	4.0680	4.0710	0.0042

Table A.18: The testing loss, learning loss, and average fitted mean of the Poisson regression model and the Poisson neural network models with different architectures, as well as for different combinations of batch size and epoch, in the case of individual-level data for admissions due to neoplasms in females from the years 2016 to 2019.

Model	**Epochs**	**Batch Size**	**Learning Loss**	**Testing Loss**	**Average Fitted Mean**
NN (35,25,20)	250	10,000	3.8950	3.9370	0.0042
NN (25,20,15)	250	10,000	3.8950	3.9390	0.0043
NN (20,15,10)	250	10,000	3.8870	3.9410	0.0043
NN (15,10,5)	250	10,000	3.8990	3.9360	0.0041
NN (35,25,20)	250	30,000	3.8930	3.9370	0.0042
NN (25,20,15)	250	30,000	3.8950	3.9410	0.0042
NN (20,15,10)	250	30,000	3.9010	3.9370	0.0041
NN (15,10,5)	250	30,000	3.8970	3.9380	0.0042
NN (35,25,20)	250	50,000	3.8980	3.9380	0.0042
NN (25,20,15)	250	50,000	3.8990	3.9390	0.0042
NN (20,15,10)	250	50,000	3.9020	3.9390	0.0042
NN (15,10,5)	250	50,000	3.9010	3.9370	0.0042
NN (35,25,20)	250	75,000	3.9030	3.9390	0.0041
NN (25,20,15)	250	75,000	3.9030	3.9400	0.0042
NN (20,15,10)	250	75,000	3.9060	3.9440	0.0042
NN (15,10,5)	250	75,000	3.9130	3.9460	0.0041
NN (35,25,20)	500	10,000	3.8950	3.9370	0.0042
NN (25,20,15)	500	10,000	3.8950	3.9390	0.0043
NN (20,15,10)	500	10,000	3.8820	3.9460	0.0042
NN (15,10,5)	500	10,000	3.8990	3.9360	0.0041
NN (35,25,20)	500	30,000	3.8930	3.9370	0.0042
NN (25,20,15)	500	30,000	3.8890	3.9430	0.0043
NN (20,15,10)	500	30,000	3.8850	3.9420	0.0043
NN (15,10,5)	500	30,000	3.8970	3.9380	0.0042
NN (35,25,20)	500	50,000	3.8900	3.9410	0.0042
NN (25,20,15)	500	50,000	3.8960	3.9370	0.0043
NN (20,15,10)	500	50,000	3.8960	3.9370	0.0042
NN (15,10,5)	500	50,000	3.8950	**3.9340**	0.0043
NN (35,25,20)	500	75,000	3.8930	3.9370	0.0042
NN (25,20,15)	500	75,000	3.8960	3.9360	0.0043
NN (20,15,10)	500	75,000	3.8990	3.9370	0.0042
NN (15,10,5)	500	75,000	3.8970	3.9360	0.0043

Table A.18: *Cont.*

Fold	Model	Learning Loss	Testing Loss	Average Fitted Mean
1	Pois.reg	16.759	16.711	0.003
	BP-G	16.781	16.680	0.003
	NN (20,15,10)	22.784	23.012	0.006
	CANN (20,15,10): variant 1	20.089	20.250	0.005
	CANN (20,15,10): variant 2	16.759	16.725	0.003
2	Pois.reg	16.673	17.503	0.003
	BP-G	16.696	17.484	0.003
	NN (20,15,10)	22.589	23.798	0.006
	CANN (20,15,10): variant 1	19.925	21.101	0.005
	CANN (20,15,10): variant 2	16.673	17.501	0.003
3	Pois.reg	16.648	17.738	0.003
	BP-G	16.669	17.734	0.003
	NN (20,15,10)	22.581	24.065	0.006
	CANN (20,15,10): variant 1	19.922	20.953	0.005
	CANN (20,15,10): variant 2	16.650	17.745	0.003
4	Pois.reg	16.730	17.010	0.003
	BP-G	16.760	16.891	0.003
	NN (20,15,10)	22.834	22.592	0.006
	CANN (20,15,10): variant 1	20.073	20.113	0.005
	CANN (20,15,10): variant 2	16.732	17.013	0.003
5	Pois.reg	16.650	17.746	0.003
	BP-G	16.674	17.665	0.003
	NN (20,15,10)	22.840	22.556	0.006
	CANN (20,15,10): variant 1	20.084	20.330	0.005
	CANN (20,15,10): variant 2	16.653	17.749	0.003
6	Pois.reg	16.847	15.957	0.003
	BP-G	16.876	15.831	0.003
	NN (20,15,10)	22.853	21.104	0.006
	CANN (20,15,10): variant 1	20.251	18.733	0.005
	CANN (20,15,10): variant 2	16.849	15.957	0.003
7	Pois.reg	16.924	15.236	0.003
	BP-G	16.949	15.216	0.003
	NN (20,15,10)	22.894	22.173	0.006
	CANN (20,15,10): variant 1	20.255	18.975	0.005
	CANN (20,15,10): variant 2	16.927	15.245	0.003

Table A.19: The testing loss, learning loss, and average fitted mean of classical, Bayesian regression and network-based models under the Poisson distributional assumptions for 10 different folds.

Fold	Model	Learning Loss	Testing Loss	Average Fitted Mean
8	Pois.reg	16.769	16.667	0.003
	BP-G	16.786	16.608	0.003
	NN (20,15,10)	22.868	22.440	0.006
	CANN (20,15,10): variant 1	20.158	19.866	0.005
	CANN (20,15,10): variant 2	16.772	16.673	0.003
9	Pois.reg	16.790	16.459	0.003
	BP-G	16.812	16.400	0.003
	NN (20,15,10)	22.722	22.270	0.006
	CANN (20,15,10): variant 1	20.131	19.486	0.005
	CANN (20,15,10): variant 2	16.791	16.467	0.003
10	Pois.reg	16.682	17.461	0.003
	BP-G	16.707	17.342	0.003
	NN (20,15,10)	22.661	23.638	0.006
	CANN (20,15,10): variant 1	19.997	21.140	0.005
	CANN (20,15,10): variant 2	16.683	17.452	0.003

Table A.19: *Cont.*

A.6 Hyper-parameter tuning for NN models for risk profile level data

Model	Epochs	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg			16.759	16.711	0.003
NN (100,75,50)	100	5,000	16.918	16.979	0.003
NN (75,50,25)	100	5,000	16.984	17.056	0.003
NN (50,35,25)	100	5,000	16.950	16.986	0.003
NN (35,25,20)	100	5,000	17.262	17.435	0.003
NN (25,20,15)	100	5,000	17.469	17.611	0.003
NN (20,15,10)	100	5,000	18.988	19.070	0.003
NN (15,10,5)	100	5,000	21.495	21.666	0.005
NN (100,75,50)	100	30,000	17.322	17.555	0.003
NN (75,50,25)	100	30,000	18.107	18.295	0.003
NN (50,35,25)	100	30,000	17.940	18.103	0.003
NN (35,25,20)	100	30,000	19.668	19.921	0.004
NN (25,20,15)	100	30,000	22.504	22.717	0.006
NN (20,15,10)	100	30,000	46.838	47.553	0.016
NN (15,10,5)	100	30,000	118.170	120.732	0.041
NN (100,75,50)	100	50,000	17.670	17.952	0.003
NN (75,50,25)	100	50,000	19.294	19.495	0.004
NN (50,35,25)	100	50,000	19.149	19.319	0.004
NN (35,25,20)	100	50,000	22.935	23.142	0.006
NN (25,20,15)	100	50,000	30.992	31.234	0.010
NN (20,15,10)	100	50,000	78.646	79.683	0.028
NN (15,10,5)	100	50,000	184.929	188.641	0.064
NN (100,75,50)	250	5,000	16.918	16.972	0.003
NN (75,50,25)	250	5,000	16.890	16.921	0.003
NN (50,35,25)	250	5,000	16.858	16.878	0.003
NN (35,25,20)	250	5,000	17.014	17.114	0.003
NN (25,20,15)	250	5,000	17.014	17.097	0.003
NN (20,15,10)	250	5,000	17.472	17.641	0.003
NN (15,10,5)	250	5,000	18.917	18.976	0.003

Table A.20: The testing loss, learning loss, and average fitted mean of the Poisson regression model and the Poisson neural network models with different architectures, as well as for different combinations of batch size and epoch, in the case of risk profile level data for admissions due to respiratory diseases in 2016.

Model	Epochs	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
NN (100,75,50)	250	30,000	17.046	17.170	0.003
NN (75,50,25)	250	30,000	17.192	17.300	0.003
NN (50,35,25)	250	30,000	17.121	17.184	0.003
NN (35,25,20)	250	30,000	17.640	17.883	0.003
NN (25,20,15)	250	30,000	18.340	18.496	0.003
NN (20,15,10)	250	30,000	22.464	22.671	0.006
NN (15,10,5)	250	30,000	39.697	40.409	0.014
NN (100,75,50)	250	50,000	17.152	17.319	0.003
NN (75,50,25)	250	50,000	17.426	17.578	0.003
NN (50,35,25)	250	50,000	17.293	17.392	0.003
NN (35,25,20)	250	50,000	18.238	18.516	0.003
NN (25,20,15)	250	50,000	19.110	19.280	0.004
NN (20,15,10)	250	50,000	28.911	29.287	0.009
NN (15,10,5)	250	50,000	63.636	64.966	0.023
NN (100,75,50)	500	5,000	16.680	16.728	0.002
NN (75,50,25)	500	5,000	16.650	16.759	0.003
NN (50,35,25)	500	5,000	16.494	**16.464**	0.003
NN (35,25,20)	500	5,000	16.642	16.731	0.003
NN (25,20,15)	500	5,000	16.855	16.904	0.003
NN (20,15,10)	500	5,000	16.733	16.747	0.003
NN (15,10,5)	500	5,000	16.902	16.958	0.003
NN (100,75,50)	500	30,000	16.936	17.027	0.003
NN (75,50,25)	500	30,000	16.994	17.071	0.003
NN (50,35,25)	500	30,000	16.940	16.984	0.003
NN (35,25,20)	500	30,000	17.271	17.450	0.003
NN (25,20,15)	500	30,000	17.437	17.583	0.003
NN (20,15,10)	500	30,000	19.047	19.131	0.003
NN (15,10,5)	500	30,000	22.107	22.300	0.006
NN (100,75,50)	500	50,000	16.990	17.098	0.003
NN (75,50,25)	500	50,000	17.113	17.217	0.003
NN (50,35,25)	500	50,000	17.039	17.093	0.003
NN (35,25,20)	500	50,000	17.461	17.680	0.003
NN (25,20,15)	500	50,000	17.943	18.085	0.003
NN (20,15,10)	500	50,000	20.238	20.373	0.005
NN (15,10,5)	500	50,000	29.675	30.104	0.010
NN (100,75,50)	1,000	5,000	16.353	**16.576**	0.002
NN (75,50,25)	1,000	5,000	16.321	**16.493**	0.002
NN (50,35,25)	1,000	5,000	16.481	**16.448**	0.002
NN (35,25,20)	1,000	5,000	16.343	**16.569**	0.003
NN (25,20,15)	1,000	5,000	16.318	**16.434**	0.003
NN (20,15,10)	1,000	5,000	16.356	**16.483**	0.003
NN (15,10,5)	1,000	5,000	16.428	**16.452**	0.002

Table A.20: *Cont.*

Model	Epochs	Batch Size	Learning Loss	Testing Loss	Average Fitted Mean
NN (100,75,50)	1,000	30,000	16.901	16.971	0.003
NN (75,50,25)	1,000	30,000	16.894	16.943	0.003
NN (50,35,25)	1,000	30,000	16.862	16.894	0.003
NN (35,25,20)	1,000	30,000	17.052	17.174	0.003
NN (25,20,15)	1,000	30,000	17.102	17.208	0.003
NN (20,15,10)	1,000	30,000	18.412	18.545	0.003
NN (15,10,5)	1,000	30,000	18.967	19.028	0.003
NN (100,75,50)	1,000	50,000	16.921	17.002	0.003
NN (75,50,25)	1,000	50,000	16.947	17.011	0.003
NN (50,35,25)	1,000	50,000	16.896	16.937	0.003
NN (35,25,20)	1,000	50,000	17.168	17.324	0.003
NN (25,20,15)	1,000	50,000	17.240	17.389	0.003
NN (20,15,10)	1,000	50,000	18.835	18.909	0.003
NN (15,10,5)	1,000	50,000	19.838	19.945	0.004
NN (100,75,50)	1,500	5,000	16.353	**16.576**	0.002
NN (75,50,25)	1,500	5,000	16.321	**16.493**	0.002
NN (50,35,25)	1,500	5,000	16.481	**16.448**	0.002
NN (35,25,20)	1,500	5,000	16.343	**16.569**	0.003
NN (25,20,15)	1,500	5,000	16.318	**16.434**	0.003
NN (20,15,10)	1,500	5,000	16.356	**16.483**	0.003
NN (15,10,5)	1,500	5,000	16.428	**16.452**	0.002
NN (100,75,50)	1,500	30,000	16.886	16.943	0.003
NN (75,50,25)	1,500	30,000	16.866	16.907	0.003
NN (50,35,25)	1,500	30,000	16.826	16.851	0.003
NN (35,25,20)	1,500	30,000	16.989	17.085	0.003
NN (25,20,15)	1,500	30,000	16.965	17.030	0.003
NN (20,15,10)	1,500	30,000	17.218	17.343	0.003
NN (15,10,5)	1,500	30,000	18.676	18.780	0.003
NN (100,75,50)	1,500	50,000	16.901	16.976	0.003
NN (75,50,25)	1,500	50,000	16.896	16.947	0.003
NN (50,35,25)	1,500	50,000	16.858	16.893	0.003
NN (35,25,20)	1,500	50,000	17.056	17.178	0.003
NN (25,20,15)	1,500	50,000	17.101	17.202	0.003
NN (20,15,10)	1,500	50,000	18.402	18.536	0.003
NN (15,10,5)	1,500	50,000	18.966	19.027	0.003

Table A.20: *Cont.*

Model	**Batch Size**	**Epochs**	**Learning Loss**	**Testing Loss**	**Average Fitted Mean**
Pois.reg			16.759	16.711	0.0027
CANN(100,75,50)	5,000	100	16.917	16.959	0.0026
CANN (75,50,25)	5,000	100	17.074	17.096	0.0027
CANN (50,35,25)	5,000	100	17.056	17.059	0.0026
CANN (35,25,20)	5,000	100	17.221	17.250	0.0027
CANN (25,20,15)	5,000	100	17.546	17.576	0.0027
CANN (20,15,10)	5,000	100	18.067	18.081	0.0029
CANN (15,10,5)	5,000	100	18.767	18.851	0.0037
CANN(100,75,50)	30,000	100	17.823	18.035	0.0029
CANN (75,50,25)	30,000	100	18.402	18.565	0.0033
CANN (50,35,25)	30,000	100	18.490	18.645	0.0033
CANN (35,25,20)	30,000	100	20.047	20.205	0.0046
CANN (25,20,15)	30,000	100	22.565	22.883	0.0062
CANN (20,15,10)	30,000	100	39.666	40.608	0.0138
CANN (15,10,5)	30,000	100	105.000	108.902	0.0371
CANN(100,75,50)	50,000	100	18.528	18.778	0.0031
CANN (75,50,25)	50,000	100	19.947	20.149	0.0045
CANN (50,35,25)	50,000	100	20.277	20.440	0.0047
CANN (35,25,20)	50,000	100	25.489	25.697	0.0075
CANN (25,20,15)	50,000	100	32.763	33.267	0.0108
CANN (20,15,10)	50,000	100	72.493	74.065	0.0258
CANN (15,10,5)	50,000	100	176.045	182.288	0.0609
CANN(100,75,50)	5,000	250	16.880	16.892	0.0025
CANN (75,50,25)	5,000	250	16.854	16.864	0.0026
CANN (50,35,25)	5,000	250	16.850	16.849	0.0026
CANN (35,25,20)	5,000	250	16.896	16.884	0.0026
CANN (25,20,15)	5,000	250	16.966	16.985	0.0027
CANN (20,15,10)	5,000	250	17.204	17.259	0.0027
CANN (15,10,5)	5,000	250	17.395	17.387	0.0027
CANN(100,75,50)	30,000	250	17.137	17.209	0.0028
CANN (75,50,25)	30,000	250	17.424	17.485	0.0028
CANN (50,35,25)	30,000	250	17.401	17.451	0.0028
CANN (35,25,20)	30,000	250	17.689	17.780	0.0029
CANN (25,20,15)	30,000	250	18.063	18.168	0.0031
CANN (20,15,10)	30,000	250	19.965	20.147	0.0047
CANN (15,10,5)	30,000	250	33.158	34.015	0.0112

Table A.21: The testing loss, learning loss, and average fitted mean of the Poisson regression model and the Poisson CANN model: (variant 1) with different architectures, as well as for different combinations of batch size and epoch, in the case of risk profile level data for admissions due to respiratory diseases in 2016.

Model	Batch Size	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
CANN(100,75,50)	50,000	250	17.351	17.478	0.0028
CANN (75,50,25)	50,000	250	17.737	17.858	0.0029
CANN (50,35,25)	50,000	250	17.743	17.860	0.0028
CANN (35,25,20)	50,000	250	18.201	18.310	0.0032
CANN (25,20,15)	50,000	250	18.815	18.981	0.0038
CANN (20,15,10)	50,000	250	24.336	24.754	0.0073
CANN (15,10,5)	50,000	250	55.153	57.028	0.0197
CANN(100,75,50)	5,000	500	16.638	**16.703**	0.0024
CANN (75,50,25)	5,000	500	16.685	16.785	0.0026
CANN (50,35,25)	5,000	500	16.677	16.726	0.0026
CANN (35,25,20)	5,000	500	16.672	16.718	0.0026
CANN (25,20,15)	5,000	500	16.843	16.859	0.0026
CANN (20,15,10)	5,000	500	16.862	16.899	0.0026
CANN (15,10,5)	5,000	500	16.985	17.043	0.0027
CANN(100,75,50)	30,000	500	16.915	16.966	0.0027
CANN (75,50,25)	30,000	500	17.084	17.103	0.0027
CANN (50,35,25)	30,000	500	17.065	17.067	0.0027
CANN (35,25,20)	30,000	500	17.246	17.272	0.0027
CANN (25,20,15)	30,000	500	17.556	17.598	0.0027
CANN (20,15,10)	30,000	500	18.060	18.086	0.003
CANN (15,10,5)	30,000	500	19.091	19.190	0.004
CANN(100,75,50)	50,000	500	17.025	17.083	0.0027
CANN (75,50,25)	50,000	500	17.266	17.290	0.0028
CANN (50,35,25)	50,000	500	17.241	17.249	0.0027
CANN (35,25,20)	50,000	500	17.459	17.522	0.0028
CANN (25,20,15)	50,000	500	17.823	17.907	0.0028
CANN (20,15,10)	50,000	500	18.720	18.807	0.0037
CANN (15,10,5)	50,000	500	23.668	24.043	0.0069
CANN(100,75,50)	5,000	1,000	16.128	**16.534**	0.0024
CANN (75,50,25)	5,000	1,000	16.290	**16.593**	0.0023
CANN (50,35,25)	5,000	1,000	16.312	**16.516**	0.0023
CANN (35,25,20)	5,000	1,000	16.643	**16.693**	0.0025
CANN (25,20,15)	5,000	1,000	16.644	16.744	0.0026
CANN (20,15,10)	5,000	1,000	16.635	**16.709**	0.0026
CANN (15,10,5)	5,000	1,000	16.985	17.043	0.0027
CANN(100,75,50)	30,000	1,000	16.858	16.888	0.0027
CANN (75,50,25)	30,000	1,000	16.876	16.896	0.0027
CANN (50,35,25)	30,000	1,000	16.870	16.879	0.0027
CANN (35,25,20)	30,000	1,000	16.945	16.948	0.0027
CANN (25,20,15)	30,000	1,000	17.063	17.098	0.0027
CANN (20,15,10)	30,000	1,000	17.424	17.448	0.0027
CANN (15,10,5)	30,000	1,000	17.827	17.830	0.0028

Table A.21: *Cont.*

Model	Batch Size	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
CANN(100,75,50)	50,000	1,000	16.884	16.929	0.0027
CANN (75,50,25)	50,000	1,000	16.976	17.001	0.0027
CANN (50,35,25)	50,000	1,000	16.955	16.965	0.0027
CANN (35,25,20)	50,000	1,000	17.103	17.119	0.0027
CANN (25,20,15)	50,000	1,000	17.347	17.370	0.0027
CANN (20,15,10)	50,000	1,000	17.776	17.783	0.0028
CANN (15,10,5)	50,000	1,000	18.283	18.315	0.0031
CANN(100,75,50)	5,000	1500	16.128	**16.534**	0.0024
CANN (75,50,25)	5,000	1500	16.290	**16.593**	0.0023
CANN (50,35,25)	5,000	1500	16.312	**16.516**	0.0023
CANN (35,25,20)	5,000	1500	16.643	**16.693**	0.0025
CANN (25,20,15)	5,000	1500	16.644	16.744	0.0026
CANN (20,15,10)	5,000	1500	16.635	**16.709**	0.0026
CANN (15,10,5)	5,000	1500	16.985	17.043	0.0027
CANN(100,75,50)	30,000	1500	16.847	16.867	0.0027
CANN (75,50,25)	30,000	1500	16.837	16.852	0.0027
CANN (50,35,25)	30,000	1500	16.835	16.833	0.0026
CANN (35,25,20)	30,000	1500	16.853	16.850	0.0027
CANN (25,20,15)	30,000	1500	16.914	16.938	0.0027
CANN (20,15,10)	30,000	1500	17.077	17.163	0.0027
CANN (15,10,5)	30,000	1500	17.327	17.299	0.0027
CANN(100,75,50)	50,000	1500	16.856	16.891	0.0027
CANN (75,50,25)	50,000	1500	16.875	16.897	0.0027
CANN (50,35,25)	50,000	1500	16.865	16.875	0.0027
CANN (35,25,20)	50,000	1500	16.938	16.945	0.0027
CANN (25,20,15)	50,000	1500	17.060	17.097	0.0027
CANN (20,15,10)	50,000	1500	17.423	17.448	0.0027
CANN (15,10,5)	50,000	1500	17.824	17.825	0.0028

Table A.21: *Cont.*

Model	Batch Size	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
Pois.reg			16.759	16.711	0.0027
CANN(100,75,50)	5,000	100	16.532	16.763	0.0028
CANN (75,50,25)	5,000	100	16.521	16.664	0.0028
CANN (50,35,25)	5,000	100	16.765	16.726	0.0027
CANN (35,25,20)	5,000	100	16.577	16.657	0.0028
CANN (25,20,15)	5,000	100	16.759	16.737	0.0027
CANN (20,15,10)	5,000	100	16.763	16.714	0.0027
CANN (15,10,5)	5,000	100	16.763	16.713	0.0027
CANN(100,75,50)	30,000	100	16.774	16.697	0.0027
CANN (75,50,25)	30,000	100	16.759	16.706	0.0027
CANN (50,35,25)	30,000	100	16.819	16.777	0.0028
CANN (35,25,20)	30,000	100	16.760	16.697	0.0027
CANN (25,20,15)	30,000	100	16.761	16.729	0.0027
CANN (20,15,10)	30,000	100	16.761	16.727	0.0027
CANN (15,10,5)	30,000	100	16.759	16.716	0.0027
CANN(100,75,50)	50,000	100	16.764	16.706	0.0026
CANN (75,50,25)	50,000	100	16.762	16.708	0.0026
CANN (50,35,25)	50,000	100	16.790	16.750	0.0027
CANN (35,25,20)	50,000	100	16.759	16.706	0.0027
CANN (25,20,15)	50,000	100	16.758	16.733	0.0027
CANN (20,15,10)	50,000	100	16.762	16.728	0.0027
CANN (15,10,5)	50,000	100	16.759	16.722	0.0027
CANN(100,75,50)	5,000	250	16.532	16.763	0.0028
CANN (75,50,25)	5,000	250	16.330	**16.500**	0.0028
CANN (50,35,25)	5,000	250	16.765	16.726	0.0027
CANN (35,25,20)	5,000	250	16.381	16.551	0.0029
CANN (25,20,15)	5,000	250	16.759	16.737	0.0027
CANN (20,15,10)	5,000	250	16.525	16.685	0.0027
CANN (15,10,5)	5,000	250	16.763	16.713	0.0027
CANN(100,75,50)	30,000	250	16.774	16.697	0.0027
CANN (75,50,25)	30,000	250	16.759	16.706	0.0027
CANN (50,35,25)	30,000	250	16.819	16.777	0.0028
CANN (35,25,20)	30,000	250	16.760	16.697	0.0027
CANN (25,20,15)	30,000	250	16.761	16.729	0.0027
CANN (20,15,10)	30,000	250	16.761	16.727	0.0027
CANN (15,10,5)	30,000	250	16.759	16.716	0.0027

Table A.22: The testing loss, learning loss, and average fitted mean of the Poisson regression model and the Poisson CANN model: (variant 2) with different architectures, as well as for different combinations of batch size and epoch, in the case of risk profile level data for admissions due to respiratory diseases in 2016.

Model	Batch Size	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
CANN(100,75,50)	50,000	250	16.764	16.706	0.0026
CANN (75,50,25)	50,000	250	16.762	16.708	0.0026
CANN (50,35,25)	50,000	250	16.790	16.750	0.0027
CANN (35,25,20)	50,000	250	16.759	16.706	0.0027
CANN (25,20,15)	50,000	250	16.758	16.733	0.0027
CANN (20,15,10)	50,000	250	16.762	16.728	0.0027
CANN (15,10,5)	50,000	250	16.759	16.722	0.0027
CANN(100,75,50)	5,000	500	16.532	16.763	0.0028
CANN (75,50,25)	5,000	500	16.330	**16.500**	0.0028
CANN (50,35,25)	5,000	500	16.765	16.726	0.0027
CANN (35,25,20)	5,000	500	16.381	16.551	0.0029
CANN (25,20,15)	5,000	500	16.093	16.590	0.0026
CANN (20,15,10)	5,000	500	16.045	16.608	0.0027
CANN (15,10,5)	5,000	500	16.763	16.713	0.0027
CANN(100,75,50)	30,000	500	16.774	16.697	0.0027
CANN (75,50,25)	30,000	500	16.759	16.706	0.0027
CANN (50,35,25)	30,000	500	16.819	16.777	0.0028
CANN (35,25,20)	30,000	500	16.760	16.697	0.0027
CANN (25,20,15)	30,000	500	16.279	16.735	0.0027
CANN (20,15,10)	30,000	500	16.761	16.727	0.0027
CANN (15,10,5)	30,000	500	16.759	16.716	0.0027
CANN(100,75,50)	50,000	500	16.764	16.706	0.0026
CANN (75,50,25)	50,000	500	16.235	16.550	0.0028
CANN (50,35,25)	50,000	500	16.201	16.822	0.0027
CANN (35,25,20)	50,000	500	16.759	16.706	0.0027
CANN (25,20,15)	50,000	500	16.758	16.733	0.0027
CANN (20,15,10)	50,000	500	16.448	16.645	0.0027
CANN (15,10,5)	50,000	500	16.759	16.722	0.0027
CANN(100,75,50)	5,000	1,000	16.532	16.763	0.0028
CANN (75,50,25)	5,000	1,000	16.330	**16.500**	0.0028
CANN (50,35,25)	5,000	1,000	16.765	16.726	0.0027
CANN (35,25,20)	5,000	1,000	16.381	16.551	0.0029
CANN (25,20,15)	5,000	1,000	16.093	16.590	0.0026
CANN (20,15,10)	5,000	1,000	16.045	16.608	0.0027
CANN (15,10,5)	5,000	1,000	16.763	16.713	0.0027
CANN(100,75,50)	30,000	1,000	16.774	16.697	0.0027
CANN (75,50,25)	30,000	1,000	16.759	16.706	0.0027
CANN (50,35,25)	30,000	1,000	16.819	16.777	0.0028
CANN (35,25,20)	30,000	1,000	16.760	16.697	0.0027
CANN (25,20,15)	30,000	1,000	16.279	16.735	0.0027
CANN (20,15,10)	30,000	1,000	16.761	16.727	0.0027
CANN (15,10,5)	30,000	1,000	16.759	16.716	0.0027

Table A.22: *Cont.*

Model	Batch Size	Epochs	Learning Loss	Testing Loss	Average Fitted Mean
CANN(100,75,50)	50,000	1,000	16.764	16.706	0.0026
CANN (75,50,25)	50,000	1,000	16.235	16.550	0.0028
CANN (50,35,25)	50,000	1,000	16.147	16.836	0.0027
CANN (35,25,20)	50,000	1,000	16.105	16.623	0.0027
CANN (25,20,15)	50,000	1,000	16.758	16.733	0.0027
CANN (20,15,10)	50,000	1,000	16.407	16.652	0.0027
CANN (15,10,5)	50,000	1,000	16.759	16.722	0.0027
CANN(100,75,50)	5,000	1,500	16.532	16.763	0.0028
CANN (75,50,25)	5,000	1,500	16.330	**16.500**	0.0028
CANN (50,35,25)	5,000	1,500	16.765	16.726	0.0027
CANN (35,25,20)	5,000	1,500	16.381	16.551	0.0029
CANN (25,20,15)	5,000	1,500	16.093	16.590	0.0026
CANN (20,15,10)	5,000	1,500	16.045	16.608	0.0027
CANN (15,10,5)	5,000	1,500	16.763	16.713	0.0027
CANN(100,75,50)	30,000	1,500	16.774	16.697	0.0027
CANN (75,50,25)	30,000	1,500	16.759	16.706	0.0027
CANN (50,35,25)	30,000	1,500	16.819	16.777	0.0028
CANN (35,25,20)	30,000	1,500	16.760	16.697	0.0027
CANN (25,20,15)	30,000	1,500	16.279	16.735	0.0027
CANN (20,15,10)	30,000	1,500	16.761	16.727	0.0027
CANN (15,10,5)	30,000	1,500	16.759	16.716	0.0027
CANN(100,75,50)	50,000	1,500	16.764	16.706	0.0026
CANN (75,50,25)	50,000	1,500	16.235	16.550	0.0028
CANN (50,35,25)	50,000	1,500	16.147	16.836	0.0027
CANN (35,25,20)	50,000	1,500	16.105	16.623	0.0027
CANN (25,20,15)	50,000	1,500	16.758	16.733	0.0027
CANN (20,15,10)	50,000	1,500	16.407	16.652	0.0027
CANN (15,10,5)	50,000	1,500	16.759	16.722	0.0027

Table A.22: *Cont.*

Fold	Model	Learning Loss	Testing Loss	Average Fitted Mean
1	Pois.reg	16.759	16.711	0.0030
	BP-G	16.781	16.680	0.0030
	Pois.reg w/ interaction	16.723	16.678	0.0027
	BP-G w/ interaction	16.752	16.702	0.0028
	NN (20,15,10)	16.343	16.406	0.0030
	CANN (20,15,10): variant 1	16.864	16.866	0.0026
	CANN (20,15,10): variant 2	16.761	16.709	0.0027
	NN (25,20,15)	16.340	16.445	0.0025
	CANN(50,35,25): variant 1	16.312	16.516	0.0023
	CANN(75,50,25): variant 2	16.330	16.500	0.0028
2	Pois.reg	16.673	17.503	0.0030
	BP-G	16.696	17.484	0.0030
	Pois.reg w/ interaction	16.636	17.473	0.0027
	BP-G w/ interaction	16.671	17.604	0.0028
	NN (20,15,10)	16.302	17.200	0.0030
	CANN (20,15,10): variant 1	16.788	17.645	0.0026
	CANN (20,15,10): variant 2	16.251	17.390	0.0027
	NN (25,20,15)	16.317	17.148	0.0026
	CANN(50,35,25): variant 1	16.364	17.238	0.0025
	CANN(75,50,25): variant 2	16.668	17.494	0.0027
3	Pois.reg	16.648	17.738	0.0030
	BP-G	16.669	17.734	0.0030
	Pois.reg w/ interaction	16.608	17.738	0.0027
	BP-G w/ interaction	16.643	17.859	0.0028
	NN (20,15,10)	16.348	17.465	0.0030
	CANN (20,15,10): variant 1	16.702	17.777	0.0026
	CANN (20,15,10): variant 2	16.191	17.627	0.0027
	NN (25,20,15)	16.231	17.573	0.0025
	CANN(50,35,25): variant 1	16.268	17.631	0.0026
	CANN(75,50,25): variant 2	16.396	17.959	0.0029

Table A.23: Testing loss, learning loss, and average fitted mean for classical, Bayesian regression, and network-based models under Poisson distributional assumption after hyper-parameter tuning of neural network models for risk profile level data across 10 different folds.

Fold	Model	Learning Loss	Testing Loss	Average Fitted Mean
4	Pois.reg	16.730	17.010	0.0030
	BP-G	16.760	16.891	0.0030
	Pois.reg w/ interaction	16.688	17.026	0.0027
	BP-G w/ interaction	16.727	17.009	0.0028
	NN (20,15,10)	16.427	17.065	0.0020
	CANN (20,15,10): variant 1	16.791	17.215	0.0026
	CANN (20,15,10): variant 2	16.735	17.035	0.0027
	NN (25,20,15)	16.296	17.039	0.0026
	CANN(50,35,25): variant 1	16.395	17.091	0.0024
	CANN(75,50,25): variant 2	16.285	16.912	0.0027
5	Pois.reg	16.650	17.746	0.0030
	BP-G	16.674	17.665	0.0030
	Pois.reg w/ interaction	16.616	17.686	0.0027
	BP-G w/ interaction	16.658	17.747	0.0028
	NN (20,15,10)	16.286	17.409	0.0020
	CANN (20,15,10): variant 1	16.761	17.726	0.0026
	CANN (20,15,10): variant 2	16.660	17.771	0.0026
	NN (25,20,15)	16.239	17.461	0.0025
	CANN(50,35,25): variant 1	16.308	17.434	0.0025
	CANN(75,50,25): variant 2	16.649	17.760	0.0026
6	Pois.reg	16.847	15.957	0.0030
	BP-G	16.876	15.831	0.0030
	Pois.reg w/ interaction	16.813	15.898	0.0027
	BP-G w/ interaction	16.842	15.926	0.0028
	NN (20,15,10)	16.456	15.865	0.0030
	CANN (20,15,10): variant 1	16.920	16.161	0.0026
	CANN (20,15,10): variant 2	16.464	15.922	0.0027
	NN (25,20,15)	16.527	15.977	0.0026
	CANN(50,35,25): variant 1	16.530	15.845	0.0025
	CANN(75,50,25): variant 2	16.881	16.095	0.0026
7	Pois.reg	16.924	15.236	0.0030
	BP-G	16.949	15.216	0.0030
	Pois.reg w/ interaction	16.893	15.160	0.0027
	BP-G w/ interaction	16.943	15.247	0.0028
	NN (20,15,10)	16.472	15.134	0.0030
	CANN (20,15,10): variant 1	16.992	15.314	0.0026
	CANN (20,15,10): variant 2	16.925	15.256	0.0028
	NN (25,20,15)	16.538	15.122	0.0026
	CANN(50,35,25): variant 1	16.572	15.143	0.0025
	CANN(75,50,25): variant 2	16.551	15.105	0.0027

Table A.23: *Cont.*

Fold	Model	Learning Loss	Testing Loss	Average Fitted Mean
8	Pois.reg	16.769	16.667	0.0030
	BP-G	16.786	16.608	0.0030
	Pois.reg w/ interaction	16.733	16.624	0.0027
	BP-G w/ interaction	16.781	16.658	0.0028
	NN (20,15,10)	16.366	16.165	0.0030
	CANN (20,15,10): variant 1	16.855	16.647	0.0026
	CANN (20,15,10): variant 2	16.338	16.435	0.0028
	NN (25,20,15)	16.439	16.178	0.0025
	CANN(50,35,25): variant 1	16.399	16.118	0.0025
	CANN(75,50,25): variant 2	16.366	16.445	0.0028
9	Pois.reg	16.790	16.459	0.0030
	BP-G	16.812	16.400	0.0030
	Pois.reg w/ interaction	16.750	16.451	0.0027
	BP-G w/ interaction	16.783	16.472	0.0028
	NN (20,15,10)	16.370	16.400	0.0030
	CANN (20,15,10): variant 1	16.883	16.607	0.0026
	CANN (20,15,10): variant 2	16.114	16.663	0.0027
	NN (25,20,15)	16.426	16.437	0.0025
	CANN(50,35,25): variant 1	16.502	16.466	0.0025
	CANN(75,50,25): variant 2	16.799	16.439	0.0027
10	Pois.reg	16.682	17.461	0.0030
	BP-G	16.707	17.342	0.0030
	Pois.reg w/ interaction	16.645	17.423	0.0027
	BP-G w/ interaction	16.695	17.424	0.0028
	NN (20,15,10)	16.414	17.406	0.0020
	CANN (20,15,10): variant 1	16.800	17.602	0.0026
	CANN (20,15,10): variant 2	16.315	17.495	0.0027
	NN (25,20,15)	16.266	17.483	0.0025
	CANN(50,35,25): variant 1	16.385	17.471	0.0023
	CANN(75,50,25): variant 2	15.981	17.432	0.0027

Table A.23: *Cont.*

A.7 Additional details for interpretable zero-inflated network models

Algorithm A.4 Pseudo code for ZIPNN model.

Step 0: Read data
Step 1: Data preparation and feature pre-processing
Step 2: Choose learning and test sample
Step 3: Constructing neural network

initialise hyper parameters
create the initial layer
create main architecture
for $i = 1$ to number of hidden layers **do**
    input: activation function
    create a layer with the required number of nodes in i^{th} layer
end for
create intermediate layer
create a hidden layer with two nodes
  set activation function ← NULL
create distribution layer and output node
create offset ← log(exposure)
$x_i \leftarrow i^{th}$ node in the intermediate layer
    $(1 - \pi) \leftarrow x_1$
    $\lambda \leftarrow x_2$+ offset
define mixture distribution
    component one← Poisson process with rate λ with probability $(1 - \pi)$
    component two← zero generating process with probability π
output node ← mean() of the distribution

Step 4: Model assembly and configuration
assemble the model defining inputs and outputs
configure the model // compile//
  input: loss function ← 'negative log likelihood'
  input: optimizer ← 'nadam'

Step 5: Model fitting
Step 6: Model evaluation
predict for testing data and **calculate** testing loss

```
# Main architecture with 3 hidden layers with q1,q2,q3 nodes in each layer
Network<- Design %>%
  # 1st hidden layer
  layer_dense(units = q1, activation = 'tanh', name = 'hidden1') %>% layer_dropout(
      rate = p) %>%
```

```
   # 2nd hidden layer
  layer_dense(units = q2, activation = 'tanh', name = 'hidden2') %>% layer_dropout(
      rate = p) %>%
  # 3rd hidden layer
  layer_dense(units = q3, activation = 'tanh', name = 'hidden3') %>% layer_dropout(
      rate = p) %>%
  # provide two neuron in the intermediate layer
  layer_dense(units = 2, name = 'Net')

Network<-list(Network,LogVol) %>% layer_concatenate(name='concate1')# adding offset
      node
#function used in lambda layer to add the exposure and to define the zero inflated
    distribution
zero_inf<-function(out){
  # rate/lambda= exp(node 1 + offset) and the squeeze function is used to flatten
      the tensor
  rate=list(out[,1],out[,3]) %>% layer_add(name = 'Add') %>% k_exp() %>% tf$squeeze
      ()
  #probability of count component = node 2 and the sigmoid function ensures value
      between 0 and 1
  s = tf$math$sigmoid(out[,2]) %>% layer_flatten()
  probs = layer_concatenate(list(1-s,s),axis = -1) # probabilities of two
      components in the mixture distribution

# Defining zero inflated distribution as a mixture distribution
  return (tfd_mixture(  #(I)
    cat = tfd_categorical(probs = probs), #(II)
    components = list(
      tfd_deterministic(loc = tf$zeros_like(rate)), #(III)
      tfd_poisson(rate = rate)) #(IV)
  )
  )
}
------------------------------------------------------------------------------
#(I): Creates a mixture distribution with two components: one for zero and
#one for a Poisson distribution.
#(II): Represents the probabilities of these two components using an
#instance of a categorical distribution.
#(III) and (IV) correspond to the two components:
#(III): Represents a definite value of zero using a scalar deterministic
#distribution.
#(IV): Represents a value derived from a Poisson distribution.
------------------------------------------------------------------------------

#negative loglikelihood loss function
```

```
nll<-function(y_true,y_pred)
{
  -y_pred$log_prob(k_reshape(y_true,shape = c(-1)))# k_shape for flattening the
      tensor
  #y_pred is the ZIP distribution layer
  #y_true is the observed admission count
}
--------------------------------------------------------------------------
#------------------Defining the distribution layer-------------------------
#Model is defined in such a way that the output layer is distribution layer.
#The 'convert_to_tensor_fn' is set to 'tfd_mean' in order to extract the
#fitted mean E(y) from the distribution.
#Samples from the distribution could be generated using 'tfd_sample'.
#Input to the 'make_distribution_fn' is the function used to define the ZIP
#distribution.

#Constructs a distribution layer with an underlying ZIP distribution
#assumption, which is defined as a mixture distribution.

p_y_zi =layer_distribution_lambda(make_distribution_fn = zero_inf,
    convert_to_tensor_fn = tfd_mean)#tfd_sample for sample
--------------------------------------------------------------------------

# ------------------ Model configuration and fitting -------------------------
# Model assembly
model_zi = keras_model(inputs= c(Design,LogVol), outputs=p_y_zi(Network))
summary(model_zi)

#callback to avoid overfitting
CBs<-callback_model_checkpoint("path0",monitor = "val_loss",save_best_only = TRUE,
                   verbose = 1, save_weights_only = TRUE)

# Model configuration
model_zi %>% compile(
  loss = nll, # set poisson NLL  loss function as the objective loss function
  optimizer = 'nadam'
)
# Model fitting by running gradient descent method to minimize the objective loss
    function
  fit <- model_zi %>% fit(
    list(Design.learn,LogVol.learn), # all predictors and the offset term
    Ylearn, # response
    verbose = 1, # verbose = 0 silences the progress bar for the process
    # verbose = 1 shows the fitting process, incl. learning loss and validation
        loss, epoch by epoch
```

```
    epochs = epochs, # epochs = 250
    batch_size = batchsize, # batchsize = 30,000
    validation_split = 0.2, # 20% as validation set
    callbacks = CBs)
  load_model_weights_hdf5(model_zi,"path0")
```

Listing A.8: Code for implementing $ZIPNN$.

```
#skip connection for lambda
Skip_1<-Design %>% layer_dense(units = 1, activation = 'linear', name = 'Skip_1')
#skip connection for probability
Age<-layer_input(shape = c(1), dtype = 'float32', name = 'Age')
Skip_2<-Age %>% layer_dense(units = 1, activation = 'linear', name = 'Skip_2')

add_skip<-function(out,skip_1,skip_2){
  r<-list(out[,1],skip_1) %>% layer_add(name = 'rate_skip')
  s<-list(out[,2],skip_2) %>% layer_add(name = 'p_skip')
  net_skip<-list(r,s) %>% layer_concatenate(name='net_skip')
  return(net_skip)
}
Network<-add_skip(Network,Skip_1,Skip_2)
Network<-list(Network,LogVol) %>% layer_concatenate(name='concate1')# adding offset
      node
# Model assembly
model_zi = keras_model(inputs= c(Design,LogVol,Age), outputs=p_y_zi(Network))
```

Listing A.9: Code for implementing $ZIPCANN$.

```
Model: " ZIPCANN model"
_________________________________________________________________
 Layer (type)                Output Shape Param#  Connected to
=================================================================
 Design (InputLayer)         [(None, 37)]  0      []
 hidden1 (Dense)             (None, 20)    760    ['Design[0][0]']
 dropout_2 (Dropout)         (None, 20)    0      ['hidden1[0][0]']
 hidden2 (Dense)             (None, 15)    315    ['dropout_2[0][0]']
 dropout_1 (Dropout)         (None, 15)    0      ['hidden2[0][0]']
 hidden3 (Dense)             (None, 10)    160    ['dropout_1[0][0]']
 dropout (Dropout)           (None, 10)    0      ['hidden3[0][0]']
 Net (Dense)                 (None, 2)     22     ['dropout[0][0]']
 Age (InputLayer)            [(None, 1)]   0      []
 tf.__operators__.getitem    (None,)       0      ['Net[0][0]']
 (SlicingOpLambda)
 Skip_1 (Dense)              (None, 1)     38     ['Design[0][0]']
 tf.__operators__.getitem_1  (None,)       0      ['Net[0][0]']
```

```
 (SlicingOpLambda)
 Skip_2 (Dense)              (None, 1)     2       ['Age[0][0]']
 rate_skip (Add)             (None, 1)     0       ['tf.__operators__.
                                                    getitem[0][0]',
                                                    'Skip_1[0][0]']
 p_skip (Add)                (None, 1)     0       ['tf.__operators__.
                                                    getitem_1[0][0]',
                                                    'Skip_2[0][0]']
 net_skip (Concatenate)      (None, 2)     0       ['rate_skip[0][0]',
                                                     'p_skip[0][0]']
 LogVol (InputLayer)         [(None, 1)]   0       []
 concate1 (Concatenate)      (None, 3)     0       ['net_skip[0][0]',
                                                     'LogVol[0][0]']
 distribution_lambda         (None, None)  0       ['concate1[0][0]']
 (DistributionLambda)
==================================================================
Total params: 1,297
Trainable params: 1,297
Non-trainable params: 0
```

Listing A.10: Structure of the $ZIPCANN$ model.

Algorithm A.5 Pseudo code for ZIPCANN model.

Step 0: Read data
Step 1: Data preparation and feature pre-processing
Step 2: Choose learning and test sample
Step 3: Constructing neural network
initialise hyper parameters
create the initial layer
create main architecture
for $i = 1$ to number of hidden layers **do**
    input: activation function
    create a layer with the required number of nodes in i^{th} layer
end for
create skip connection
    $n \leftarrow$ dimension of initial layer
    $x_i \leftarrow i^{th}$ node in initial layer
    create skip connection 1 $\leftarrow \beta_i x_i + ...\beta_n x_n$
    create skip connection 2 $\leftarrow \beta_i^{'} x_i + ...\beta_n^{'} x_n$
create intermediate layer
create a hidden layer with two nodes
  set activation function $\leftarrow$ NULL
    $z_i \leftarrow i^{th}$ node in the intermediate layer
    $z_1^{'} \leftarrow z_1$ + skip connection 1
    $z_2^{'} \leftarrow z_2$+ skip connection 2
create distribution layer and output node
create offset $\leftarrow log$(exposure)
$z_i^{'} \leftarrow i^{th}$ node in the intermediate layer with skip connection
    $(1 - \pi) \leftarrow z_1^{'}$
    $\lambda \leftarrow z_2^{'}$+ offset
define mixture distribution
    component one$\leftarrow$ Poisson process with rate λ with probability $(1 - \pi)$
    component two$\leftarrow$ zero generating process with probability π
output node $\leftarrow$ mean() of the distribution
Step 4: Model assembly and configuration
assemble the model defining inputs and outputs
configure the model // compile//
  input: loss function $\leftarrow$ 'negative log likelihood'
  input: optimizer $\leftarrow$ 'nadam'
Step 5: Model fitting
Step 6: Model evaluation
predict for testing data and **calculate** testing loss

```
# Main architecture with 3 hidden layers and attention layer
Attention <- Design %>%
# first three hidden layers
  layer_dense(units = q1, activation = 'tanh', name = 'hidden1') %>%layer_dropout(
```

```
      rate = p) %>%
  layer_dense(units = q2, activation = 'tanh', name = 'hidden2') %>%layer_dropout(
      rate = p) %>%
  layer_dense(units = q3, activation = 'tanh', name = 'hidden3') %>%layer_dropout(
      rate = p) %>%
  #layer for attention weights
  layer_dense(units = q0, activation = 'linear', name = 'Attention')

Network<-list(Design,Attention) %>%
  # taking dot product of attention weights and feature set
  layer_dot(name='localglm',axes = 1) %>%
  # provide one neuron in the output layer
  layer_dense(units = 1, activation = 'linear', name = 'Network')
Response = list(Network, LogVol) %>%
        # add the exposure and the last neuron
        layer_add(name = 'Add') %>%
        # give the response
        layer_dense(units = 1,
                    activation = 'exponential',
                    name = 'Response',
                    trainable = FALSE,
                    weights = list(array(1, dim = c(1,1)), array(0, dim = c(1))))

model <- keras_model(inputs = c(Design,LogVol), outputs = c(Response))
```

Listing A.11: Code for implementing *LocalGLMnet*.

Algorithm A.6 Pseudo code for LocalGLMnet approach

Step 0: Read data
Step 1: Data preparation and feature pre-processing
Step 2: Choose learning and test sample
Step 3: Constructing neural network
initialise hyper parameters
create the initial layer
create main architecture
for $i = 1$ to number of hidden layers **do**
    input: activation function
    create a layer with the required number of nodes in i^{th} layer
end for
create additional layer
    no. of nodes ($layer_{additional}$) $\leftarrow$ size of input layer
    set activation function $\leftarrow$ NULL
    $z \leftarrow (layer_{additional} \cdot input\ layer)$
create output node
create offset $\leftarrow log$(exposure)
output node $\leftarrow \beta_0 + z +$ offset
Step 4: Model assembly and configuration
assemble the model defining inputs and outputs
configure the model // compile//
Step 5: Model fitting
Step 6: Model evaluation
predict for testing data and **calculate** testing loss

```
# Main architecture with 3 hidden layers
Attention<- Design %>%
  layer_dense(units = q1, activation = 'tanh', name = 'hidden1') %>% layer_dropout(
      rate = p) %>%
  layer_dense(units = q2, activation = 'tanh', name = 'hidden2') %>% layer_dropout(
      rate = p) %>%
  layer_dense(units = q3, activation = 'tanh', name = 'hidden3') %>% layer_dropout(
      rate = p) %>%
  #layer for attention weights with dimension= 2*q_0
```

```
  layer_dense(units = c(2*q0), activation = 'linear', name = 'Attention')

# rate parameter from attention weights
lamda_local<-list(Design,Attention[,1:q0])%>% layer_dot(name='localglm_1',axes = 1)
     %>%    layer_dense(units = 1, activation = 'linear', name = 'lamda_local')

# probability parameter from attention weights
p_local   <-list(Design,Attention[,(q0+1):(2*q0)])%>% layer_dot(name='localglm_2',
    axes = 1)  %>%  layer_dense(units = 1, activation = 'linear', name = 'p_local')

Network<-list(lamda_local,p_local,LogVol) %>% layer_concatenate(name='concate1')
#the rest of the steps is same as that of the ZIPNN model without attention layer
```

Listing A.12: Code for interpretable $ZIPNN$ model.

```
Model: "Interpretable ZIPNN model"
__________________________________________________________________________
 Layer (type)               Output Shape Param # Connected to
==========================================================================
 Design (InputLayer)        [(None, 37)]  0      []
 hidden1 (Dense)            (None, 20)    760    ['Design[0][0]']
 dropout_2 (Dropout)        (None, 20)    0      ['hidden1[0][0]']
 hidden2 (Dense)            (None, 15)    315    ['dropout_2[0][0]']
 dropout_1 (Dropout)        (None, 15)    0      ['hidden2[0][0]']
 hidden3 (Dense)            (None, 10)    160    ['dropout_1[0][0]']
 dropout (Dropout)          (None, 10)    0      ['hidden3[0][0]']
 Attention (Dense)          (None, 74)    814    ['dropout[0][0]']
 tf.__operators__.getitem   (None, 37)    0      ['Attention[0][0]']
 (SlicingOpLambda)
 tf.__operators__.getitem_1 (None, 37)    0      ['Attention[0][0]']
 (SlicingOpLambda)
 localglm_1 (Dot)           (None, 1)     0      ['Design[0][0]',
                                                 'tf.__operators__.
                                                 getitem[0][0]']
 localglm_2 (Dot)           (None, 1)      0     ['Design[0][0]',
                                                 'tf.__operators__.
                                                  getitem_1[0][0]']
 lamda_local (Dense)        (None, 1)      2     ['localglm_1[0][0]']
 p_local (Dense)            (None, 1)      2     ['localglm_2[0][0]']
 LogVol (InputLayer)        [(None, 1)]    0     []
 concate1 (Concatenate)     (None, 3)      0     ['lamda_local[0][0]',
                                                  'p_local[0][0]',
                                                  'LogVol[0][0]']
 distribution_lambda        (None, None)   0     ['concate1[0][0]']
 (DistributionLambda)
```

```
31 ================================================================================
32 Total params: 2,053
33 Trainable params: 2,053
34 Non-trainable params: 0
```

Listing A.13: Structure of the interpretable $ZIPNN$ model.

Model	Learning loss	Testing loss	Average fitted mean
ZIPNN(50,35,25)	26191.8	2823.2	0.0026
ZIPNN(35,25,20)	26218.9	2827.9	0.0025
ZIPNN(25,20,15)	26319.2	2811.8	0.0026
ZIPNN(20,15,10)	26428.3	2803.3	0.0026
ZIPNN(15,10,5)	26430.2	2806.1	0.0026
ZIPCANN(50,35,25)	26358.8	2815.1	0.0027
ZIPCANN(35,25,20)	26319.3	2810.1	0.0028
ZIPCANN(25,20,15)	26286.8	2816.5	0.0027
ZIPCANN(20,15,10)	26366.3	2813.6	0.0027
ZIPCANN(15,10,5)	26383.0	2817.5	0.0027

Table A.24: The testing loss, learning loss, and average fitted mean of the ZIPNN and ZIPCANN models with (50,35,25), (35,25,20), (25,20,15), (20,15,10) and (15,10,5) neurons in the initial three hidden layers.

A.7.1 Interpretable CANN and ZIPCANN models

In case of a CANN model, an additional skip connection is used for estimating the attention weights as shown in Figure A.9.

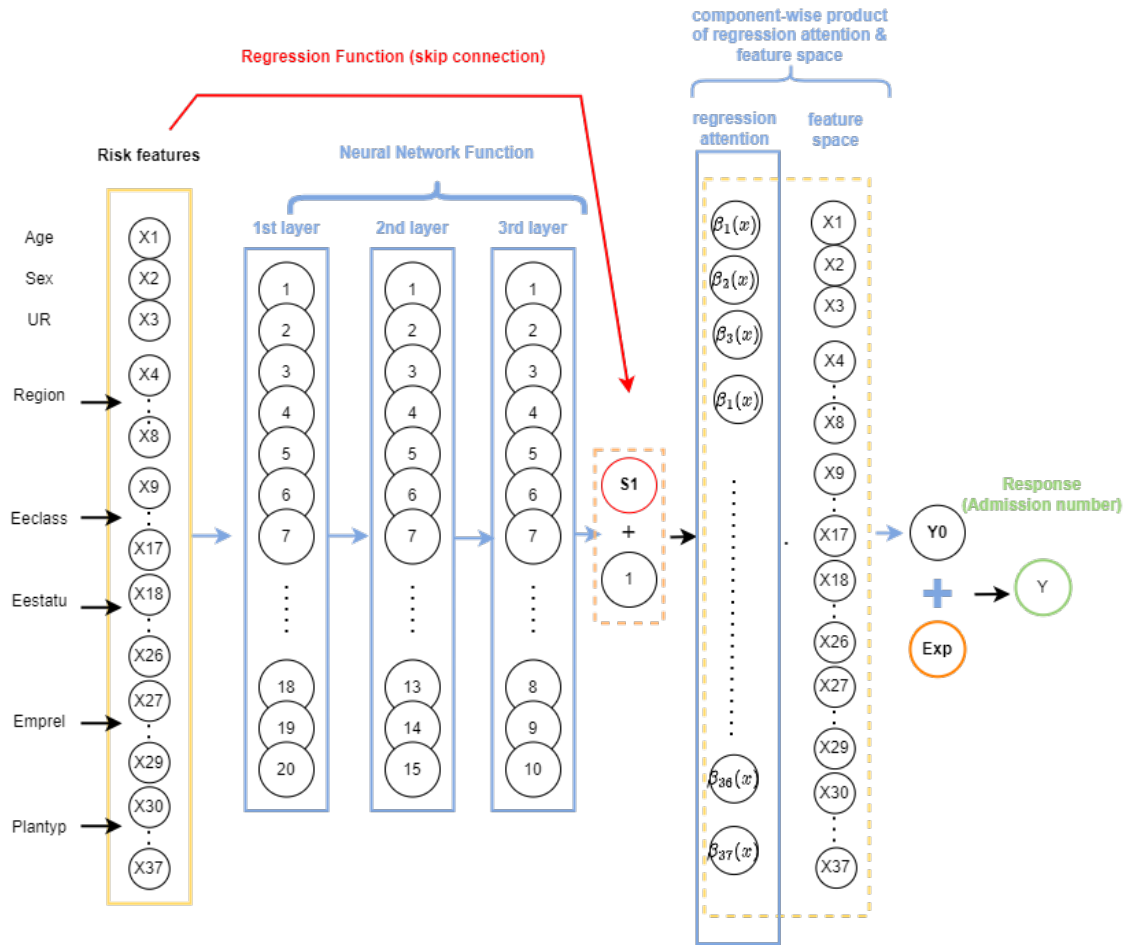


Figure A.9: An illustration of an interpretable CANN model with regression attention, one-hot encoding and 20,15,10 neurons in each of the hidden layers.

A.7.1.1 Regression attention for ZIPCANN

Contrary to the ZIPNN model, two skip connections are introduced in the case of the ZIPCANN model for estimating the rate and probability parameters (Equations 6.24 and 6.26). For accommodating the skip connections, the attention layer is introduced after the last hidden layer with two neurons and before the distribution layer (see Figure A.10). Although the specification of the distribution layer and the associated μ and p parameters using the attention weights is comparable to the ZIPNN model, the specification of the attention weights differs for the ZIPCANN model due to the skip connections. The attention weights corresponding to both parameters are estimated separately due to the distinct skip connection present for

each case. The distribution layer specification and covariate contribution analysis are then carried out, similar to the ZIPNN model.

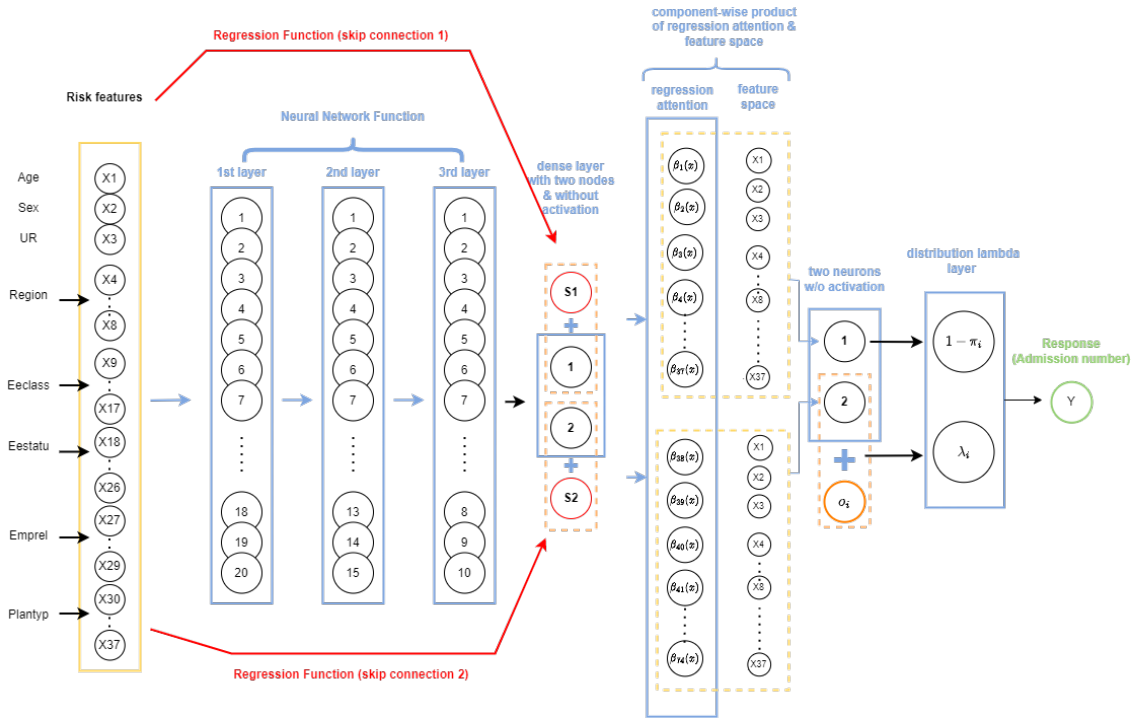


Figure A.10: An illustration of an interpretable ZIPCANN model with regression attention, one-hot encoding and 20,15,10,2 neurons in each of the layers.

Bibliography

Agresti, A. (2015). *Foundations of linear and generalized linear models*. John Wiley & Sons.

Allaire, J., & Chollet, F. (2021). *Keras: R interface to 'keras'* [R package version 2.7.0]. https://CRAN.R-project.org/package=keras

Allaire, J., & Tang, Y. (2021). *Tensorflow: R interface to 'tensorflow'* [R package version 2.6.0]. https://CRAN.R-project.org/package=tensorflow

Andersen, P. K., Borgan, O., Gill, R. D., & Keiding, N. (2012). *Statistical models based on counting processes*. Springer Science & Business Media.

Arik, A., Cairns, A. J., Dodd, E., Macdonald, A. S., Shao, A., & Streftaris, G. (2023). Insurance pricing for breast cancer under different multiple state models. *arXiv preprint arXiv:2311.15975*.

Arık, A., Dodd, E., Cairns, A., & Streftaris, G. (2021). Socioeconomic disparities in cancer incidence and mortality in england and the impact of age-at-diagnosis on cancer mortality. *Plos one*, *16*(7), e0253854.

Arık, A., Dodd, E., & Streftaris, G. (2020). Cancer morbidity trends and regional differences in england—a bayesian analysis. *PLoS One*, *15*(5), e0232844.

Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics surveys*, *4*, 40–79.

Aveyard, P., Gao, M., Lindson, N., Hartmann-Boyce, J., Watkinson, P., Young, D., Coupland, C. A., San Tan, P., Clift, A. K., Harrison, D., et al. (2021). Association between pre-existing respiratory disease and its treatment, and

severe covid-19: A population cohort study. *The lancet Respiratory medicine*, *9*(8), 909–923.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bengio, Y., Ducharme, R., & Vincent, P. (2000). A neural probabilistic language model. *Advances in neural information processing systems*, *13*.

Blanc, P. D., Annesi-Maesano, I., Balmes, J. R., Cummings, K. J., Fishwick, D., Miedinger, D., Murgia, N., Naidoo, R. N., Reynolds, C. J., Sigsgaard, T., et al. (2019). The occupational burden of nonmalignant respiratory diseases. an official american thoracic society and european respiratory society statement. *American journal of respiratory and critical care medicine*, *199*(11), 1312–1334.

Blier-Wong, C., Cossette, H., Lamontagne, L., & Marceau, E. (2020). Machine learning in p&c insurance: A review for pricing and reserving. *Risks*, *9*(1), 4. https://doi.org/10.3390/risks9010004

Bousquet, J., Khaltaev, N. G., & Cruz, A. A. (2007). *Global surveillance, prevention and control of chronic respiratory diseases*. World Health Organization.

Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control*. John Wiley & Sons.

Box, G. E., & Tiao, G. C. (1992). *Bayesian inference in statistical analysis* (Vol. 40). John Wiley & Sons.

Brooks, S. (1998). Markov chain monte carlo method and its application. *Journal of the royal statistical society: series D (the Statistician)*, *47*(1), 69–100.

Brooks, S., Gelman, A., Jones, G., & Meng, X.-L. (2011). *Handbook of markov chain monte carlo*. CRC press.

Cameron, A. C., Li, T., Trivedi, P. K., & Zimmer, D. M. (2004). Modelling the differences in counted outcomes using bivariate copula models with application to mismeasured counts. *The Econometrics Journal*, *7*(2), 566–584.

Cameron, A. C., & Trivedi, P. K. (2013). *Regression analysis of count data* (Vol. 53). Cambridge university press.

Carlin, B. P., & Louis, T. A. (2008). *Bayesian methods for data analysis*. CRC press.

CDC. (2012). Chronic obstructive pulmonary disease among adults–united states, 2011. *MMWR. Morbidity and mortality weekly report*, *61*(46), 938–943.

CDC. (2023). Health and economic costs of chronic diseases. https://www.cdc.gov/chronicdisease/about/costs/index.htm

CDC. (2021). Health effects of cigarette smoking. https://www.cdc.gov/tobacco/data_statistics/fact_sheets/health_effects/effects_cig_smoking/index.htm

CDC. (2016). Icd- 10 - cm international classification of diseases, tenth revision, clinical modification (icd-10-cm) [https://www.cdc.gov/nchs/icd/icd-10-cm.htm].

Celeux, G., Forbes, F., Robert, C. P., & Titterington, D. M. (2006). Deviance information criteria for missing data models.

Christiansen, M., Denuit, M., Lucas, N., & Schmidt, J.-P. (2018). Projection models for health expenses. *Annals of Actuarial Science*, *12*(1), 185–203.

Collett, D. (2023). *Modelling survival data in medical research*. CRC press.

Congdon, P. D. (2019). *Bayesian hierarchical models: With applications using r*. CRC Press.

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, *34*(2), 187–202.

De Jong, P., Heller, G. Z., et al. (2008). *Generalized linear models for insurance data*. Cambridge University Press.

Denuit, M., Charpentier, A., & Trufin, J. (2021). Autocalibration and tweedie-dominance for insurance pricing with machine learning. *Insurance: Mathematics and Economics*, *101*, 485–497.

Denuit, M., Hainaut, D., & Trufin, J. (2019). *Effective statistical learning methods for actuaries iii: Neural networks and extensions*. Springer International Publishing.

Dillon, J. V., Langmore, I., Tran, D., Brevdo, E., Vasudevan, S., Moore, D., Patton, B., Alemi, A., Hoffman, M., & Saurous, R. A. (2017). Tensorflow distributions. *arXiv preprint arXiv:1711.10604*.

Dodd, E. O., Streftaris, G., Waters, H. R., & Stott, A. D. (2015). The effect of model uncertainty on the pricing of critical illness insurance. *Annals of Actuarial Science*, *9*(1), 108–133.

Doney, B., Hnizdo, E., Syamlal, G., Kullman, G., Burchfiel, C., Martin, C. J., & Mujuru, P. (2014). Prevalence of chronic obstructive pulmonary disease among us working adults aged 40 to 70 years: National health interview survey data 2004 to 2011. *Journal of occupational and environmental medicine/American College of Occupational and Environmental Medicine*, *56*(10), 1088.

Dozat, T. (2016). Incorporating nesterov momentum into adam.

Draper, N. R., & Smith, H. (1998). *Applied regression analysis* (Vol. 326). John Wiley & Sons.

Duncan, I. G. (2011). *Healthcare risk adjustment and predictive modeling*. Actex Publications.

Dürr, O., Sick, B., & Murina, E. (2020). *Probabilistic deep learning: With python, keras and tensorflow probability*. Manning Publications.

Famoye, F., & Singh, K. P. (2006). Zero-inflated generalized poisson regression model with an application to domestic violence data. *Journal of Data Science*, *4*(1), 117–130.

Feng, C. (2022). Zero-inflated models for adjusting varying exposures: A cautionary note on the pitfalls of using offset. *Journal of Applied Statistics*, *49*(1), 1–23.

Ferrario, A., Noll, A., & Wuthrich, M. V. (2020). Insights from inside neural networks. *Available at SSRN 3226852*.

Frees, E. W. (2009). *Regression modeling with actuarial and financial applications*. Cambridge University Press.

Frees, E. W., Derrig, R. A., & Meyers, G. (2014). *Predictive modeling applications in actuarial science* (Vol. 1). Cambridge University Press.

Gamerman, D., & Lopes, H. F. (2006). *Markov chain monte carlo: Stochastic simulation for bayesian inference*. CRC press.

Gao, G., Meng, S., & Wüthrich, M. V. (2019). Claims frequency modeling using telematics car driving data. *Scandinavian Actuarial Journal*, *2019*(2), 143–162.

Geisser, S. (1975). The predictive sample reuse method with applications. *Journal of the American statistical Association*, *70*(350), 320–328.

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

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1995). *Bayesian data analysis*. Chapman; Hall/CRC.

Gibson, G. J., Streftaris, G., & Thong, D. (2018). Comparison and assessment of epidemic models. *Statistical Science*, *33*(1), 19–33.

Goldstein, B. A., Navar, A. M., Pencina, M. J., & Ioannidis, J. P. (2017). Opportunities and challenges in developing risk prediction models with electronic health records data: A systematic review. *Journal of the American Medical Informatics Association: JAMIA*, *24*(1), 198.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Gurmu, S. (1997). Semi-parametric estimation of hurdle regression models with an application to medicaid utilization. *Journal of applied econometrics*, *12*(3), 225–242.

Gurmu, S., & Trivedi, P. K. (1996). Excess zeros in count models for recreational trips. *Journal of Business & Economic Statistics*, *14*(4), 469–477.

Haberman, S., & Renshaw, A. E. (1996). Generalized linear models and actuarial science. *Journal of the Royal Statistical Society: Series D (The Statistician)*, *45*(4), 407–436.

Hainaut, D. (2018). A neural-network analyzer for mortality forecast. *ASTIN Bulletin: The Journal of the IAA*, *48*(2), 481–508.

Hardin, J. W., & Hilbe, J. M. (2007). *Generalized linear models and extensions*. Stata press.

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

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hejazi, S. A., & Jackson, K. R. (2016). A neural network approach to efficient valuation of large portfolios of variable annuities. *Insurance: Mathematics and Economics*, *70*, 169–181.

Hilbe, J. M. (2011). *Negative binomial regression*. Cambridge University Press.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Ibrahim, J. G., Chen, M.-H., Lipsitz, S. R., & Herring, A. H. (2005). Missing-data methods for generalized linear models: A comparative review. *Journal of the American Statistical Association*, *100*(469), 332–346.

Jeong, H., Tzougas, G., & Fung, T. C. (2023). Multivariate claim count regression model with varying dispersion and dependence parameters. *Journal of the Royal Statistical Society Series A: Statistics in Society*, *186*(1), 61–83.

Jose, A., Macdonald, A. S., Tzougas, G., & Streftaris, G. (2022). A combined neural network approach for the prediction of admission rates related to respiratory diseases. *Risks*, *10*(11), 217.

Jose, A., Macdonald, A. S., Tzougas, G., & Streftaris, G. (2024). Interpretable zero-inflated neural network models for predicting admission counts. *Accepted for publication in Annals of Actuarial Science*.

Jung, Y. (2018). Multiple predicting k-fold cross-validation for model selection. *Journal of Nonparametric Statistics*, *30*(1), 197–215.

Keydana, S. (2022). *Tfprobability: Interface to 'tensorflow probability'* [R package version 0.15.0]. https://CRAN.R-project.org/package=tfprobability

Kocherlakota, S., & Kocherlakota, K. (2001). Regression in the bivariate poisson distribution.

Kuo, K. (2019). Deeptriangle: A deep learning approach to loss reserving. *Risks*, *7*(3), 97.

Kurt, I., Ture, M., & Kurum, A. T. (2008). Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease. *Expert systems with applications*, *34*(1), 366–374.

Lambert, D. (1992). Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, *34*(1), 1–14.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Lee, A. H., Wang, K., & Yau, K. K. (2001). Analysis of zero-inflated poisson data incorporating extent of exposure. *Biometrical Journal*, *43*(8), 963–975.

Little, R. J., & Rubin, D. B. (2019). *Statistical analysis with missing data* (Vol. 793). John Wiley & Sons.

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

Lunn, D., Jackson, C., Best, N., Thomas, A., & Spiegelhalter, D. (2012). *The bugs book: A practical introduction to bayesian analysis*. CRC press.

Lunn, D., Thomas, A., Best, N., & Spiegelhalter, D. (2000). Winbugs-a bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and computing*, *10*, 325–337.

Macdonald, A. S., Richards, S. J., & Currie, I. D. (2018). *Modelling mortality with actuarial applications*. Cambridge University Press.

MarketScan, M. (2017). Marketscan ccae mdcr user guide.

Martin, J., & Hall, D. B. (2016). R 2 measures for zero-inflated regression models for count data with excess zeros. *Journal of Statistical Computation and Simulation*, *86*(18), 3777–3790.

McCullagh, P., & Nelder, J. A. (1989). *Generalized linear models* (2nd ed.). Chapman & Hall.

Millar, R. B. (2009). Comparison of hierarchical bayesian models for overdispersed count data using dic and bayes' factors. *Biometrics*, *65*(3), 962–969.

Mullahy, J. (1986). Specification and testing of some modified count data models. *Journal of econometrics*, *33*(3), 341–365.

NCI. (2021). Nci dictionary of cancer terms. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/morbidity

Nelder, J. A., & Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, *135*(3), 370–384.

Noll, A., Salzmann, R., & Wuthrich, M. V. (2020). Case study: French motor third-party liability claims. *Available at SSRN 3164764*.

Ntzoufras, I. (2011). *Bayesian modeling using winbugs* (Vol. 698). John Wiley & Sons.

Ntzoufras, I., & Dellaportas, P. (2002). Bayesian modelling of outstanding liabilities incorporating claim count uncertainty. *North American Actuarial Journal*, *6*(1), 113–125.

Ohlsson, E., & Johansson, B. (2010). *Non-life insurance pricing with generalized linear models* (Vol. 2). Springer.

Ozkok, E., Streftaris, G., Waters, H. R., & Wilkie, A. D. (2014a). Modelling critical illness claim diagnosis rates i: Methodology. *Scandinavian Actuarial Journal*, *2014*(5), 439–457.

Ozkok, E., Streftaris, G., Waters, H. R., & Wilkie, A. D. (2014b). Modelling critical illness claim diagnosis rates i: Methodology. *Scandinavian Actuarial Journal*, *2014*(5), 439–457.

Peters, G., Zhu, R., Tzougas, G., Rabitti, G., & Yusuf, I. (2022). The role and significance of green bonds in funding transition to a low carbon economy: A case study forecasting portfolios of green bond instrument returns. *Available at SSRN 4299196*.

Pitacco, E. (2014). Health insurance. *Basic Actuarial Models, Cham, Switzerland: Springer Verlag*.

Prechelt, L. (1998). Early stopping-but when? In *Neural networks: Tricks of the trade* (pp. 55–69). Springer.

R Core Team. (2021). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Richman, R. (2021). Ai in actuarial science–a review of recent advances–part 2. *Annals of Actuarial Science*, *15*(2), 230–258.

Richman, R., & Wüthrich, M. V. (2022). Localglmnet: Interpretable deep learning for tabular data. *Scandinavian Actuarial Journal*, 1–25.

Richman, R., & Wüthrich, M. V. (2020). Nagging predictors. *Risks*, *8*(3), 83.

Richman, R., & Wüthrich, M. V. (2021). A neural network extension of the lee–carter model to multiple populations. *Annals of Actuarial Science*, *15*(2), 346–366.

Ridout, M., Demétrio, C. G., & Hinde, J. (1998). Models for count data with many zeros. *Proceedings of the XIXth international biometric conference*, *19*, 179–192.

Rigby, R. A., & Stasinopoulos, D. M. (2005). Generalized additive models for location, scale and shape,(with discussion). *Applied Statistics*, *54*, 507–554.

RStudio Team. (2021). *Rstudio: Integrated development environment for r*. RStudio, PBC. Boston, MA. http://www.rstudio.com/

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Rue, H., Martino, S., & Chopin, N. (2009). Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *71*(2), 319–392.

Russell, S. J. (2010). *Artificial intelligence a modern approach*. Pearson Education, Inc.

Schelldorfer, J., & Wuthrich, M. V. (2019). Nesting classical actuarial models into neural networks. *Available at SSRN 3320525*.

Seber, G. A., & Lee, A. J. (2003). *Linear regression analysis* (Vol. 330). John Wiley & Sons.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Linde, A. (2014). The deviance information criterion: 12 years on. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *76*(3), 485–493.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *64*(4), 583–639.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929–1958.

Stephens, D. A., Crowder, M. J., & Dellaportas, P. (2004). Quantification of automobile insurance liability: A bayesian failure time approach. *Insurance: Mathematics and Economics*, *34*(1), 1–21.

Streftaris, G., & Gibson, G. J. (2002). Statistical inference for stochastic epidemic models. *Proc. 17th international Workshop on Statistical Modeling*, 609–616.

Tzougas, G., & Li, Z. (2021). Neural network embedding of the mixed poisson regression model for claim counts [https://insurancedatascience.org/project/2021_london/].

Tzougas, G., Vrontos, S., & Frangos, N. (2014). Optimal bonus-malus systems using finite mixture models. *ASTIN Bulletin: The Journal of the IAA*, *44*(2), 417–444.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Vercruysse, W., Dhaene, J., Denuit, M., Pitacco, E., & Antonio, K. (2012). Premium indexing in lifelong health insurance. *FEB Research Report AFI_1274*, 1–15.

von Bortkiewicz, L. (1898). *Das gesetz der kleinen zahlen*. BG Teubner.

Vrieze, S. I. (2012). Model selection and psychological theory: A discussion of the differences between the akaike information criterion (aic) and the bayesian information criterion (bic). *Psychological methods*, *17*(2), 228.

Walters, M., & Mishel, L. (2003). How unions help all workers. *Briefing paper*, *143*.

Wang, Z., Dellaportas, P., & Kosmidis, I. (2023). Bayesian tensor factorisations for time series of counts. *Machine Learning*, 1–20.

Ward, E. J. (2008). A review and comparison of four commonly used bayesian and maximum likelihood model selection tools. *Ecological Modelling*, *211*(1-2), 1–10.

WHO. (2022a). Asthma and covid-19: Scientific brief, 19 april 2021 [https://www.who.int/publications-detail-redirect/who-2019-ncov-sci-brief-asthma-2021.1].

WHO. (2023a). Cancer. https://www.who.int/news-room/fact-sheets/detail/cancer

WHO. (2022b). Heath topics- chronic respiratory diseases. https://www.who.int/health-topics/chronic-respiratory-diseases#tab=tab_1

WHO. (2023b). Non communicable diseases. https://www.who.int/news-room/fact-sheets/detail/noncommunicable-diseases

Wüthrich, M. V. (2021). The balance property in neural network modelling. *Statistical Theory and Related Fields*, 1–9.

Wüthrich, M. V. (2020). Bias regularization in neural network models for general insurance pricing. *European Actuarial Journal*, *10*(1), 179–202.

Wüthrich, M. V., & Merz, M. (2023). *Statistical foundations of actuarial learning and its applications*. Springer Nature.

Wüthrich, M. V., & Merz, M. (2019). Yes, we cann! *ASTIN Bulletin: The Journal of the IAA*, *49*(1), 1–3.

Yip, K. C., & Yau, K. K. (2005). On modeling claim frequency data in general insurance with extra zeros. *Insurance: Mathematics and Economics*, *36*(2), 153–163.

Zeileis, A., Kleiber, C., & Jackman, S. (2008a). Regression models for count data in R. *Journal of Statistical Software*, *27*(8). http://www.jstatsoft.org/v27/i08/

Zeileis, A., Kleiber, C., & Jackman, S. (2008b). Regression models for count data in R. *Journal of Statistical Software*, *27*(8). http://www.jstatsoft.org/v27/i08/