SAPIENZA
UNIVERSITÀ DI ROMA

A Neural Network approach to measure health insurance risk

Scuola Superiore di Scienze Statistiche

Dottorato di Ricerca in Scienze Attuariali – XXXIV Ciclo

Candidate

Alessandro G. Laporta
ID number 1537371

Thesis Advisors

Prof. Susanna Levantesi
Prof. Lea Petrella

June 2022

Thesis defended on May 30, 2022
in front of a Board of Examiners composed by:

Prof. Emilia Di Lorenzo (chairman)
Prof. Luca Regis
Prof. Nicolino Ettore D'Ortona

A Neural Network approach to measure health insurance risk
Ph.D. thesis. Sapienza – University of Rome
ISBN: 000000000-0
© 2022 Alessandro G. Laporta. All rights reserved

This thesis has been typeset by LaTeX and the Sapthesis class.

Version: June 24, 2022

Author's email: alessandro.laporta@uniroma1.it

Ringraziamenti

Approfitto di questo spazio per ringraziare le persone che hanno contribuito, con il loro instancabile supporto, alla realizzazione di questa tesi di Dottorato.

In primis, un ringraziamento speciale alla Professoressa Lea Petrella e alla Professoressa Susanna Levantesi, per essere state una guida in questi anni, per i loro consigli di ricerca e per la loro pazienza.

Ringrazio i colleghi del Dottorato (Andrea, Francesco, Jacopo e Lorenzo) per essere stati dei fantastici compagni di viaggio.

Ringrazio i colleghi del Fasi per avermi permesso, tramite il loro supporto, di portare avanti l'attività lavorativa senza pregiudicare il mio percorso in questi tre anni di Dottorato.

Infine, un ringraziamento di cuore ad Annalisa per aver avuto la pazienza di sopportare e supportare un compagno che passa i weekend ad arrovellarsi su R.

Abstract

This work presents a set of neural network applications to health insurance pricing. In recent years, the actuarial literature involving machine learning in insurance pricing has flourished. However, most actuarial machine learning research focuses on car and property and casualty insurance. While, the use of such techniques in health insurance is yet to be explored. In this manuscript, we discuss the use of neural networks to set the price of an health insurance coverage following the structure of a classical frequency-severity model. We consider neural networks to estimate claim frequency and severity. In particular, we introduce Negative Multinomial Neural Networks to jointly model the frequency of possibly correlated medical claims. We then complete the frequency-severity approach proposing Gamma Neural Networks to estimate the expected claim severity.

We then go beyond the frequency-severity framework adopting a quantile approach that allows gauging the potential riskiness of a given policyholder. Namely, we discuss the estimation of conditional quantiles of aggregate claim amounts embedding the problem in a quantile regression framework using the Neural Network approach. As the first step, we consider Quantile Regression Neural Networks (QRNN) to compute quantiles for the insurance ratemaking framework. As the second step, we propose a new Quantile Regression Combined Actuarial Neural Network (Quantile-CANN) combining the traditional quantile regression approach with a Quantile Regression Neural Network. In both cases, we adopt a two-part model scheme where we fit a logistic regression to estimate the probability of positive claims and the QRNN model or the Quantile-CANN for the positive outcomes.

Through a case study based on a health insurance dataset, we highlight the overall better performances of the different neural network models with respect to more established regression models (such as GLMs and quantile regression), both in terms of accuracy and risk diversification.

Contents

Introduction and motivation

The use of statistical learning in actuarial sciences is a long-time common practice dating back to the 1980s. The actuarial profession, specifically in the pricing domain, quickly gravitated around linear models and generalized linear models (GLM), which have gradually become the go-to approach when building actuarial models for pricing. Meanwhile, the statistical learning and computer science literature continued to flourish, expanding machine learning and designing more advanced models outperforming traditional ones in several fields. Until recent years, the actuarial literature passed over these models since they mainly belonged to the computer science community. Computer scientists, compared to actuaries, have a different approach when it comes to modeling. They are largely based on the so-called *algorithmic culture*, where the modeler is primarily interested in the model's accuracy rather than explainability. Actuaries, instead, are more familiar with the *data modeling culture*, where the main concern is model interpretability. The distance between the different cultures [8] has prevented actuaries from exploring new modeling techniques for a long time.

A seminal work encompassing the use of several machine learning algorithms for insurance ratemaking was put forward by [17] where the authors compare: linear regression, generalized linear models, Tree-based models, neural networks, and supporting vector machines. As also reported in [7], [17] in their conclusion state *"We hope this paper goes a long way towards convincing actuaries to include neural networks within their set of modeling tools for ratemaking"*. A little more than a decade after this work, machine learning algorithms, particularly neural networks, have consistently started appearing in the actuarial literature and practice. This resurgence is mainly due to the vertiginous increase in the amount of data available and the rapid advances in computation and information technology that greatly benefit the accuracy of machine learning models. Such flood of information and the availability of new tools to leverage it inevitably affect the insurance industry, disrupting all aspects of insurance firms, including pricing.

The use of machine learning in insurance pricing has grown in recent years, and the related literature is increasing accordingly. For example, [27] adopts gradient boosting for auto insurance cost modeling; [55] performs insurance pricing optimization using several machine learning models; [29] use Tree-Based techniques such as GBM in order to produce car insurance tariffs; [53] employs neural networks to enhance GLMs performances in non-life insurance; [63] proposes two different techniques to overcome the unbiasedness of neural network models for insurance portfolios. Machine learning techniques have also found extensive application in the context of insurance claim reserving: [25] improve the performances of the over-dispersed Poisson model for general insurance claims reserving through neural networks embedding, and [62] propose several machine learning algorithms for individual claim reserving. For an extensive state-of-the-art review on machine learning in actuarial sciences, see [7], and [52].

While Machine learning models are slowly making their way in insurance pricing, GLMs remain the workhorse of non-life insurance pricing. Such models allow expressing the mean of a random variable of interest as a function of the linear combination of its covariates (a.k.a. features). GLMs are simple to understand, highly informative, and with good statistical properties. However, they struggle to learn complex data structures. This is often the case in non-life insurance pricing, where the linear relationship behind these models is too restrictive and prevents the model from capturing all the relevant information from the data; for example, non-linear trends and interactions between covariates may be helpful to have an accurate reflection of reality. For the modeler, it is still possible to manually plug non-linear features and interactions in a GLM. However, this feature engineering is rather tedious and time-consuming since the structural form of the relationship between the covariates, and the response variable is frequently unknown; thus, only a limited number of possibilities are explored when building a model. In this regard, the great advantage of Machine Learning models, particularly of neural networks, is that non-linear transformations and interactions between covariates are directly learned by the algorithm, without any need for a prior specification from the modeler. For this reason, after a first inception between the end of the last century and the very beginning of the 2000s, literature on pricing via Neural Networks has started growing in recent years.
Early attempts at applying neural networks for insurance pricing include [12], [49], [56], [20] and of course the aforementiond contribution by [17]. Neural Networks are also used in [13], that compared statistical learning models for estimating the pure premium. Neural Networks are also used in recent work of [51] to estimate a posteriori claim frequencies. However, the recent revival of the discussion around neural networks in actuarial pricing is due to the work put forward by M. Wutrich and the Swiss actuarial community in recent years: In [64] and [53] the authors, first propose a Poisson Neural Network to estimate the claim frequencies for a car insurance dataset, and then present an innovative technique called Combined Actuarial Neural Network (CANN) where a GLM is nested with a neural network in order to model non linear relationships not captured by the GLM; [65] deepens the discussion around the CANN approach; [63] addresses the problem portfolio biased neural networks; [39] introduce a suite of model agnostic tools that allow to extract useful and interpretable information from neural network models.

Even though the use of neural networks in insurance pricing has already been explored, most of the research has focused on car and property insurance, while little work has been done towards the application of such models in the health insurance realm. In this work, we try to fill this gap by proposing a suite of neural network applications for health insurance pricing. First, we present the use of neural networks to price a health insurance coverage following the structure of a classical frequency-severity model, where the product between the expected values of the claim frequency and claim severity returns the pure premium of a specif policy. In particular, we consider neural networks to estimate both claim frequency and claim severity. Then, following a quantile regression approach, we use neural networks to evaluate the potential riskiness of a given health insurance coverage.

This research project is organized into four chapters. The first one serves as a cornerstone for the rest of the work since we will discuss some general concepts and models that are vital to the other chapters. More specifically: Section 1.1 provides some context to our work by offering an overview of the Italian health insurance market; In Section 1.2, we briefly discuss the Frequency-Severity framework that we will consider to compute the pure premium for a health insurance policy; Section 1.3 is devoted to a short exposition of GLMs; In Section 1.4, we talk through the theory behind neural networks offering a general framework for the different models proposed in the following chapters. In Chapter 2, we introduce a novel Negative Multinomial Neural Network approach to jointly model the frequency of a class of correlated medical claims. We chose to explore the multivariate approach because health insurance policies frequently cover a wide range of medical events, which may often be strongly correlated. Such as medical visits and diagnostic testing, where the referral given by a medical visit is frequently an essential requirement to undergo a diagnostic test. The use of a Negative Multinomial approach in the context of neural networks is a novelty in actuarial sciences since all research work on neural networks for claim frequency estimation is based on the univariate Poisson distributional assumption. In order to test if the proposed model has some added value w.r.t. to a standard approach, we compare its performance against the estimates produced by a Negative Multinomial GLM (see [69]).

In Chapter 3 we complete the Frequency-Severity approach introducing Gamma Neural Networks to estimate the expected claim severity, i.e., the average cost of a given claim. The actuarial literature regarding the use of neural networks to estimate the average claim severity is still scarce (see [47] and [68]), and to the best of our knowledge, Gamma Neural Networks have never been proposed. A similar approach, but for Tree-based models, is explored in [29] where the models are trained to minimize a Gamma deviance. The premiums produced with neural networks are then compared to those issued by GLMs by means of the methodologies proposed in [15] and that we will extensively discuss in Section 3.4. Moreover, we deepen the understanding of our models by applying a set of model agnostic tools proposed in [39], that allow us to shed light on the data representation learned by the models.

The new techniques proposed above are devoted to estimating the expected value of a given variable of choice (claim frequency and claim severity) since they are designed to return the pure premium of a specific health insurance policy in the context of a frequency-severity approach. Hence, these models, even if they offer insight into the average loss of a policy, useful to set a price, cannot provide the modeler with some valuable information about the potential riskiness of the policy, e.g., the quantile of the total claim amount. To overcome this problem, a quantile regression approach, initially introduced by [35], may be considered since it provides information on the whole distribution of a given phenomenon. The Quantile regression technique represents a robust distribution-free methodology that has been widely used in the financial literature to compute risk measures like the Value-at-Risk (see for example [58], [61], [38], [3], [50], [59]).

The quantile regression approach appears particularly suitable in the insurance context when assessing the Solvency II capital requirements and calculating the premium safety loadings. Furthermore, it enables the insurer to gauge the portfolio riskiness (i.e., computing the Value-at-Risk of a given portfolio). Modeling the quantile claim amount through the Quantile Regression (QR) has already been discussed by a handful of authors: [36] was the first introducing the use of the two-stage QR model to estimate the quantile of the total claim amount; [30] propose a refinement of the previous model since they take into account heterogeneous claim probabilities, whereas [36] only considers a single probability of having claims for each type of policyholder; [5] propose an alternative two-stage approach, where the risk margin considered in the ratemaking is calibrated on the claim's severity for each risk class in the portfolio, avoiding some of the drawbacks that characterize the technique proposed by [30].

The standard QR methods, similarly to GLMs, require the specification of a predetermined dependence structure between the dependent variable and the covariates or to elaborate its complex functional form to account for non-linearity or interactions among regressors. Unfortunately, the structural form of the dependence is often unknown to the modeler. So a different approach should be pursued in this context. Also in this context, neural networks appear to be an interesting modeling technique overcoming these limitations since they can fit a complex data structure without any apriori assumption on the dependence structure.

Therefore, Chapter 4 proposes two innovative methods to estimate the conditional quantile of the total claim amount for a group of health insurance policies employing neural networks. The first one uses the Quantile Regression Neural Network (QRNN), a particular specification of a feed-forward Neural Network originally introduced by [57], devoted to quantile estimation. Up to our knowledge, this model has never been used in the context of actuarial sciences.

The second model we propose considers a new extension of the Combined Actuarial Neural Network (CANN) proposed by [53] in a quantile regression framework. The original CANN formulation, as mentioned above, is devoted to claim frequency estimation and combines a Poisson GLM with a neural network. In our model, since we are interested in the conditional quantile of the total claim amount, we nest the QR model into the structure of a Neural Network (Quantile-CANN henceforth). This approach is able to represent additional information incorporated in the data and not captured by the simple QR model.

The structure of the approach here considered is based on a two-part model. This framework involves a model for a binary indicator variable and a model for the response variable, given that the binary indicator takes the value one. Following this approach, we fit a logistic regression for the binary variable to estimate the claim probability while we use a QRNN or a Quantile-CANN model for the positive outcomes, i.e., the total claim amount to calculate its quantile. Using the estimated quantiles of the claim amount, we finally calculate a loaded premium following the Quantile Premium Principle (QPP) considered in [30]. Then, as an analogy to Chapter 3, we compare the tariff structure produced by the different models (Quantile Regression, QRNN and Quantile-CANN) using a set of techniques put forward by [23].

Chapter 1

Pricing in health insurance

1.1 Health insurance in Italy

The Italian health insurance market serves as a complement to the public health supply provided by the National Health Service (NHS). The NHS is a regional-based healthcare system founded on the principles of universal coverage, equality in access, and solidarity in financing. It guarantees uniform health care across the country based on a national statutory benefits package (the essential levels of assistance [6]). The service is financed primarily by national and regional taxes and supplemented by co-payments. Even though the state has extensively financed the Italian NHS from its inception in 1978, an increasingly high number of families choose to subscribe some form of health insurance. Most explanations for the increase in subscriptions of health insurance policies focus on the factors related to the demand side of NHS, including long waiting lists, rising co-payments, expenditure oriented behaviour of the policyholders, perceptions of the public system's inadequacy, and changes in individual attitudes about supporting the redistributive role of public healthcare. Some numbers in order to give an idea on the increasing relevance of voluntary health insurance in Italy: at the very end of the last century in 1999 only around 2% of the population was covered by an health insurance, it increased up to a 12% in 2013 and reached 23% of the total population in 2019[1] [11].

The increase in the market share for health insurance in Italy is also due to incentives provided by the public authority in order to complement the supply given by the NHS, in particular for a class of services that are excluded or not entirely covered by the public supply, such as long term care assistance, dental care, cosmetic, thermal treatments and alternative medicine. In fact, until the late 1990s, the health insurance market in Italy was marginal. Private insurance was principally purchased by high-income, well-educated, healthy people for themselves and their families or by large companies as a benefit for their high-level employees. However, especially after the 2007-2008 economic crisis, Italy's government has progressively limited the growth of its own contributions to public health care financing. Indeed, from 2010 to

[1] In spite of its increase, the health insurance sector still plays a modest role in terms of healthcare funding. The share of current health expenditures intermediated via health insurance rose from 2.2% in 2012 (53.1 euros per capita) to 2.6% in 2017 (66 euros per capita), and recent estimates suggest that these numbers would double in the next decade

2017, public health expenditure as a percentage of GDP decreased from 7% in 2010 to 6.5% in 2017, and public health expenditure's share of total health expenditures dropped from 78.5% in 2010 to 73.9% in 2017. The cost-containment policies adopted by the Government have increased the out-of-pocket payment, which reached 23.5% of total health expenditure in 2017 [40], and the expenditure for voluntary health insurance. Another driver that contributed significantly to the increasing importance of health insurance is the spreading of employer-paid private health insurance, which is often provided as an employee benefit and as an alternative to wage increases [19], and [48]. Such seems to have been the case for Italy, where wages have seen a long stagnation period, and trade unions often exchanged more occupational welfare, particularly health care coverage, for less remuneration [28].
The voluntary health insurance market in Italy is characterized by a plethora of institutions with heterogeneous purposes, products, and regulations. From a product perspective, it is possible to define classes kind of health insurance coverage:

- Collective health insurance is mainly driven by the Integrated Health Funds (IHFs) introduced in 1992 for financing healthcare services that were supplementary and complementary to the NHS. The IHFs are entities, associations, and mutual aid companies, that are regularly registered in the National Register of Funds established by the Ministry of Health in 2008. These funds are managed as non-profit organizations based on mutuality and solidarity; they cannot carry out risk selection policies. Therefore, they must accept everyone who demands health insurance without discrimination based on age, health status, medical history, or other individual characteristics. By 2020 more than 14 million people in Italy have joined some sort of IHF. The National Register of Funds encompasses two types of funds, Type-A and Type-B funds, that provide different healthcare services. Type-A funds, which provide individual or group plans to individuals for themselves and their families, have been regulated since 1999 to complement and supplement the NHS benefits package. They cannot offer coverage for health services already provided by the NHS, but they can cover cost-sharing and offer other services listed by the law (i.e., alternative medical services, thermal treatments, and dental services) that supplement those of the NHS. Type-B Funds are group insurance schemes offered to employees, mainly as part of the occupational welfare included in collective agreements or employer-specific conditions (employer-based insurance), and may duplicate, complement, or supplement NHS coverage. At least 20 percent of the premiums collected by Type B funds must be allocated to health services supplementary to the NHS benefits package (such as dental and long-term care). Both types of IHFs can either choose to directly manage the risk coming from coverage or transfer the risk to a private insurance company. The latter are commonly known as Reinsured IHFs, while the former are called Autoinsured IHFs.
  Collective health insurance is also provided by private insurance companies. This kind of product is generally designed for firms without a category health fund to provide risk coverage for their workers. However, such products are often proposed as an alternative to employees that already have an employer-based IHF, creating competition in the market.

- Individual health insurance: this market share belongs exclusively to private insurance companies. Unlike health funds, the insurance company can carry out risk selection policies based on age, health status, etc., and the coverage price is tailored to the individual policyholder characteristics, as usual in the insurance market.

Even though insurance companies and IHF compete roughly for the same market, they have some crucial differences that should be discussed. First, insurance companies are profit-driven organizations, and IHF are non-profit ones; this leads to different market policies as discussed above. For instance, given their non-profit nature, IHF can afford to underwrite bad risks, while insurance companies cannot. A clear example is the possibility for IHF policyholders to extend their policy when they retire from work; often, this is not possible in insurance companies, where pensioners are not allowed to hold their policy. For the same reason, insurance companies need to apply individual price discrimination since this is where profits arise; in contrast, IHF generally apply the same flat premium for all the policyholders in their portfolio [2].

Another fundamental difference concerns the regulatory framework: insurance companies must ensure financial stability by meeting the capital requirements defined by Solvency II and EIOPA regulations, while IHFs are both missing a regulatory framework and a vigilance authority; still, in recent years, different guidelines for IHF have been proposed ([41]). According to such guidelines, the best practice to assess the long-term sustainability of IHF is the so-called technical balance sheet, a popular actuarial technique that consists of a ten-year projection of the fund's assets and liabilities. Despite the recent development of health funds in terms of membership, their accounts show a limited capitalization, which is likely to undermine their long-term solvency. Therefore in the coming years, it will be paramount for the Italian legislation to develop appropriate regulations for such institutions to provide additional protection to policyholders by ensuring long-term financial sustainability for IHF.

The market of voluntary health insurance in Italy is divided as follows: 38% of the market is coverered by Autoinsured IHF[3], 42% belongs to Reinsured IHF and the last 20% is covered by private insurance companies either via Individual insurance policies or Collective insurance policies.

Although insurance companies and IHF have some relevant differences, they share a common objective: carefully gauging the riskiness of their policyholders. In particular, insurance companies must evaluate the expected cost of a policyholder to make sound pricing policies resulting in effective risk control and hopefully into a profit. From an IHF perspective, estimating the potential cost of a group of policyholders is paramount to ensure its long-term sustainability. Since having an effective costing model with an excellent predictive ability allows the institution to thoroughly evaluate the consistency of its future expected liabilities. In the next

[2] Often some IHF consider a set of flat premium to be applied at different policyholder based on their working status.

[3] Also considering Mutual funds

section, we tackle the Frequency-Severity approach, which is the standard framework employed by actuaries when evaluating the expected cost (or pricing) of a non-life insurance coverage.

1.2 Frequency-Severity approach

Setting the price of an insurance policy is strictly related to evaluating the cost associated with the risk coverage provided by the insurance contract. In other terms, in order to price a contract, insurers predict the expected total claim amount S_i for each policyholder based on his or her observable characteristics $\boldsymbol{x}_i$. The insurer therefore develops a predictive model $f(.)$, mapping the risk factors $\boldsymbol{x}_i$ to the predicted loss cost with $E(S_i/\boldsymbol{x}_i) = f(\boldsymbol{x}_i)$. Therefore, determining the price of a health insurance coverage is strictly connected to determining the medical expenditure of the policyholders during the time of coverage.

Models for medical expenditure are usually designed using a two-part approach, where the medical expenditure of a group of individuals is modeled in two steps. First, the population is split between users of medical services and non-users using a logit model. While in the second part, the expenditure amount is modeled using a classical regression model.

The use of a two-part model approach to model an individual's expenditure during a year is only suitable when analyzing a specific illness. However, over one year, individuals may face multiple illness events, see [21]. Hence, a more viable approach is to consider frequency-severity modeling, which is the primary statistical approach for modeling non-life insurance claims. This approach splits the total claim amount[4] for a given insured i, into a compound sum that accounts for the number of filed claims and determines the individual medical claim sizes. Thus, the total claim amount S_i is represented using a compound random variable with N_i describing the number of claims that occur over one year and $\tilde{S}_{i,1}$, …,$\tilde{S}_{i,N_i}$ describing the individual claim sizes defined as claim severities. More formally:

$$S_i = \tilde{S}_{i,1} + \tilde{S}_{i,2} + \cdots + \tilde{S}_{i,N_i} = \sum_{j=1}^{N_i} \tilde{S}_{i,j} \tag{1.1}$$

this approach is built upon three standard assumptions:

- N_i is a discrete random variable which takes values in $\mathbb{N}$;
- $\tilde{S}_{i,1}, \ldots, \tilde{S}_{i,j}, \ldots, \tilde{S}_{i,N_i}$ are independent and identically distributed (i.i.d.);
- N_i and $\tilde{S}_{i,1}, \ldots, \tilde{S}_{i,N_i}$ are independent.

for such properties we have that:

$$E(S_i) = E(N_i) \cdot E(\tilde{S}_{i,j}), \tag{1.2}$$

where $\tilde{S}_{i,j}$ is the cost for a generic claim filed by policyholder i, for proofs see [34]. Basically, the goal of a health insurance company, given a set of risk drivers $\boldsymbol{x}_i$, is

[4] from now on we will use an insurance based vocabulary, hence we will use terms such as claim amount rather than health expenditure

to asses the expected total claim amount $E(S_i/\boldsymbol{x}_i)$ of an insurance coverage over one year, for costing or pricing purposes. In Chapters 2 and 3 we explore how to model the claim frequency[5] $E(N_i/\boldsymbol{x}_i)$ and the claim severity $E(\tilde{S}_{i,1}/\boldsymbol{x}_i)$ for a set of different claim types using neural networks and drawing a comparison with GLMs.

1.3 GLM

To assess the possible merits of neural networks for health insurance pricing, we first have to establish a benchmark model that meets industry standards. Generalized linear models, first introduced by [45], provide a unified procedure to model responses with a distribution belonging to the exponential family with respect to a given transformation of a set of covariates. Today, GLMs are state-of-the-art statistical models in many applied fields; in actuarial science, these models are mainly used for predictive modeling, and they represent by far the most popular choice in the industry when it comes to developing pricing models. They are commonly employed to model claim frequency and claim severity. In this Section, without aiming to be exhaustive, we briefly highlight the main features characterizing GLM.

We start by defining the set of observations $\{\boldsymbol{Y}, \boldsymbol{x}\}$, where $\boldsymbol{Y} = (Y_1, \ldots, Y_I)$ is the vector gathering the observations for the phenomenon of interest and $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_I)$ is the $I \times p$ matrix of covariates.

A GLM specifying the relationship between $\boldsymbol{Y}$ and $\boldsymbol{x}$ is built upon the following hypothesis:

- Distributional hypothesis: the responses $Y_1, \ldots, Y_I$ are i.i.d. following a distribution belonging to the exponential family;
- Structural hypothesis: the relationship between the expectation $\mu_i = E(Y_i/\boldsymbol{x}_i)$ and $\boldsymbol{x}_i$ is represented as:

  $$g(\mu_i) = \boldsymbol{x}_i'\boldsymbol{\beta}, \qquad \text{for } i = 1, 2, \ldots, I, \tag{1.3}$$

  where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ is the vector of parameters to be estimated via Maximum Likelihood by the model and $g(.)$ is a monotonic, differentiable and invertible link function such that:

  $$E(Y_i/\boldsymbol{x}_i) = \mu_i = g^{-1}(\boldsymbol{x}_i'\boldsymbol{\beta}), \qquad \text{for } i = 1, 2, \ldots, I. \tag{1.4}$$

The choice of the distribution for Y_i depends on the specific task at hand. For instance, in a claim frequency model, the response variable typically follows a count distribution such as the Poisson or the Negative Binomial. While in a claim severity model, the response variable generally follows a right-skewed distribution with a long right tail such as the Gamma or the Log-normal. In the following lines, we provide some details on the exponential distribution family that characterizes the distributional assumption on Y_i.

[5] A proper definition for the claim frequency involves scaling the claim counts N_i using the exposure at risk during the year $e_i \in (0, 1]$ for policyholder i. Resulting in the claim frequency $F_i = \frac{N_i}{e_i}$. However, as we will discuss in Section 2.1, in this work we consider a portfolio where each insured is exposed for the entire year, i.e. $e_i = 1$. Therefore we neglect the exposure e_i.

Exponential distribution family

As discussed above $Y_1, \dots, Y_I$ are distributed accordingly to the same exponential family, such that:

$$Y_i \sim f(y; \theta_i, \phi, \omega_i) = \exp\left\{\frac{\omega_i}{\phi}\left[y\theta_i - b(\theta_i)\right]\right\} c(y, \phi, \omega_i), \qquad i = 1, \dots, I, \tag{1.5}$$

where θ_i is the canonical parameter which is strictly connected to the mean of the distribution, ϕ is the dispersion parameter (also known as nuisance), $b(.)$ and $c(.)$ are known functions and $\omega_i > 0$ is the weight. It is also worth mentioning that:

- Given an exponential family, $b(.)$ does not depend on i;
- ϕ does not depend on i;
- both θ_i and ω_i do depend on i.

For the properties of such family of distributions, as proven in [45], we have:

$$\mathrm{E}(Y_i) = b^{'}(\theta_i) \qquad \mathrm{var}(Y_i) = \frac{\phi}{\omega_i} b^{''}(\theta_i) \tag{1.6}$$

As shown in Table 1.1 by changing the parameters displayed in Eq.(1.5) we obtain different popular distributions, that can be particularly useful to describe specif phenomena in the context of insurance pricing.

Family distribution	θ_i	ϕ/ω_i	$b(\theta_i)$	$c(y, \phi, \omega_i)$
Normal	μ_i	σ_i^2/ω_i	$\theta_i^2/2$	$(2\pi\phi/\omega_i)^{-0.5}\mathrm{exp}(-y^2/2(\phi/\omega_i))$
Poisson	$\log(\mu_i)$	$1/\omega_i$	$\exp(\theta_i)$	$1/y!$
Negative Binomial	$\log(\mu_i/(\alpha+\mu_i))$	$1/\omega_i$	$-\alpha\log(1-\exp^{\theta})$	$\Gamma(\alpha+y)/(\Gamma(\alpha)y!)$
Gamma	$-1/\mu_i$	$1/(\alpha \cdot \omega_i)$	$-\log(-\theta)$	$(1/(\phi/\omega_i))^{1/(\phi/\omega_i)} y^{1/(\phi/\omega_i)-1}/\Gamma(1/(\phi/\omega_i))$

Table 1.1. Some possible parametrizations for the exponential distribution family

In the next section, we present the theoretical background for neural networks. We will not dwell on using such models for insurance pricing since this issue will be addressed in the following chapters. Still, the section serves as a cornerstone for the different network models proposed in this work.

1.4 Neural networks

Neural networks are popular machine learning models inspired by brains' functionality, having their roots in the 1940s. In essence, neural networks can be used as high-dimensional nonlinear regression functions, and the resulting model can be seen as a parametric regression model. However, due to their high dimensionality and

non-interpretability of parameters, they are often called non-parametric regression models[6].
Let us begin with the description of the elements characterizing the generic architecture of a neural network.

1.4.1 Neural network architecture

Activation function
The first important element of a neural network architecture is the choice of the activation function $\phi : \mathbb{R} \to \mathbb{R}$. There are plenty of choices available in the literature for such function, but since we would like to approximate non-linear regression functions, the activation function ϕ should be non-linear, too. The most popular choices of activation functions are:

- exponential activation function:
$$\phi(x) = \exp(x); \tag{1.7}$$
- hyperbolic tangent activation function:
$$\phi(x) = \tanh(x); \tag{1.8}$$
- rectified linear unit (ReLU) activation function:
$$\phi(x) = x \cdot \mathbb{I}_{x \leq 0}; \tag{1.9}$$
- sigmoid activation function:
$$\phi(x) = (1 + \exp(-x))^{-1}. \tag{1.10}$$

Layers
A neural network consist of several network layers. A network layer $\boldsymbol{z}^{(s)}$, given an activation function ϕ, is a mapping from dimension q_{s-1} to dimension q_s defined as:

$$\boldsymbol{z}^{(s)} : \mathbb{R}^{q_{s-1}} \to \mathbb{R}^{q_s}, \qquad \boldsymbol{z}^{(s)}(\boldsymbol{\theta}^{(s)}) = \left(z_1^{(s)}(\boldsymbol{\theta}_1^{(s)}), \cdots, z_{q_s}^{(s)}(\boldsymbol{\theta}_{q_s}^{(s)})\right)' \tag{1.11}$$

where $z_j^{(s)}(\boldsymbol{\theta}_1^{(s)})$ with $j = 1, \ldots, q_s$ are the neurons composing the network layer, defined as:

$$z_j^{(s)}(\boldsymbol{\theta}_j^{(s)}) = \phi\left(\theta_{j,0}^{(s)} + \sum_{l=1}^{q_{s-1}} \theta_{j,l}^{(s)} \cdot z_l^{(s-1)}(\boldsymbol{\theta}_l^{(s-1)})\right) = \phi(a_j^{(s)}) \tag{1.12}$$

[6] In this work, we only focus on Feed-forward Neural Networks, nonetheless, the computer science literature has developed a plethora of Network architectures, such as Recurrent Neural Networks, Convolutional Neural Networks, Self Organizing Maps, Generative Adversial Networks etc.

where $a_j^{(s)}$ represents the neuron value before the activation is applied and $\boldsymbol{\theta}_j^{(s)} = (\theta_{j0}^{(s)}, \theta_{j,1}^{(s)}, \dots, \theta_{j,q_{s-1}}^{(s)})'$ is the vector of parameters belonging to j-th neuron in the s-th hidden layer. More specifically $\theta_{j,0}^{(s)}$ is the parameter (known as *bias*), for the j-th neuron in the s-th hidden layer, that works as a sort of intercept. While $\theta_{j,l}^{(s)}$ is the parameter (a.k.a. *weight*) connecting the l-st neuron in the $(s-1)$-th layer to the j-th neuron in the s-th layer. Considering the vector of parameters $\boldsymbol{\theta}_1^{(s)}, \dots, \boldsymbol{\theta}_{q_s}^{(s)}$ for each neuron in Eq.(1.11), we can define the matrix of parameters for the s-th hidden layer as $\boldsymbol{\theta}^{(s)} = (\boldsymbol{\theta}_1^{(s)}, \dots, \boldsymbol{\theta}_j^{(s)}, \dots, \boldsymbol{\theta}_{q_s}^{(s)})'$ of dimension $q_s \times (1 + q_{s-1})$. It is interesting to observe that neuron $z_j^{(s)}$ describes a regression function with respect to the neurons in layer $\boldsymbol{z}^{(s-1)}$, that consists of a scalar product (similar to GLMs) and then measures the resulting activation of this scalar product in a non-linear fashion via the activation function ϕ. All things considered, a neural network layer can be abbreviated in a vectorial notation as:

$$\boldsymbol{z}^{(s)}(\boldsymbol{\theta}^{(s)}) = \phi(\boldsymbol{a}^{(s)}) \tag{1.13}$$

As mentioned above, a feed-forward neural network architecture is a composition of several neural network layers. Namely, we define the number of layers composing the network K (a.k.a. the network *dept*), the network architecture starts with the input layer $\boldsymbol{z}^0$, which coincides with the set of covariates $\boldsymbol{z}^0 = \boldsymbol{x}$, where $\boldsymbol{x}$ is the $I \times p$ matrix of observations used as the training set. Then, the network continues with the composition of K hidden layers. The layer in the last hidden layer K is defined as:

$$\boldsymbol{z}^{(K:1)}(\boldsymbol{x}_i) = \left(\boldsymbol{z}^{(K)}(\boldsymbol{\theta}^{(K)}) \circ \dots \circ \boldsymbol{z}^{(s)}(\boldsymbol{\theta}^{(s)}) \circ \dots \circ \boldsymbol{z}^{(1)}(\boldsymbol{\theta}^{(1)})\right)(\boldsymbol{x}_i), \qquad \text{for } i = 1, 2, \dots, I, \tag{1.14}$$

The neurons in the last hidden layer are transformed in the output layer using a final activation function ψ and a set of parameters for the output layer.

$$z^F(\boldsymbol{x}_i)(\boldsymbol{\theta}) = \psi(\theta_0^{K+1} + \textstyle\sum_{l=1}^{q_K} \theta_l^{K+1} z_l^{(K:1)}(\boldsymbol{x}_i)) = \psi(a^{(F)}(\boldsymbol{x}_i)), \qquad \text{for } i = 1, 2, \dots, I, \tag{1.15}$$

Figure 1.1 provides a graphical representation of the architecture for a neural network with $K = 2$. It is possible to denote with $\boldsymbol{\theta}$ the full set of parameters for the network gathering the matrix of parameters of each layer:

$$\boldsymbol{\theta} = \left\{\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(s)}, \cdots, \boldsymbol{\theta}^{(K)}, \boldsymbol{\theta}^{(K+1)}\right\} \tag{1.16}$$

In other terms $\boldsymbol{\theta}$ encompasses all weights and biases in the network.

1.4.2 Training neural networks: gradient descent and backpropagation

Now that we have drawn a general framework to define neural networks, here we discuss the gradient descent method employed to obtain the optimal set of parameters

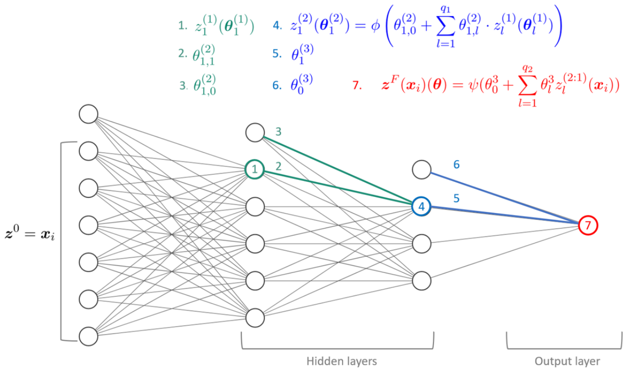

1. $z_1^{(1)}(\boldsymbol{\theta}_1^{(1)})$
2. $\theta_{1,1}^{(2)}$
3. $\theta_{1,0}^{(2)}$
4. $z_1^{(2)}(\boldsymbol{\theta}_1^{(2)}) = \phi\left(\theta_{1,0}^{(2)} + \sum_{l=1}^{q_1} \theta_{1,l}^{(2)} \cdot z_l^{(1)}(\boldsymbol{\theta}_l^{(1)})\right)$
5. $\theta_1^{(3)}$
6. $\theta_0^{(3)}$
7. $\boldsymbol{z}^F(\boldsymbol{x}_i)(\boldsymbol{\theta}) = \psi(\theta_0^3 + \sum_{l=1}^{q_2} \theta_l^3 z_l^{(2:1)}(\boldsymbol{x}_i))$
$\boldsymbol{z}^0 = \boldsymbol{x}_i$
1
2
3
4
5
6
7
Hidden layers
Output layer

Figure 1.1. Neural network architecture for a network with $K = 2$, $q_0 = 7, q_1 = 5, q_2 = 3$, and $\boldsymbol{z}^0 = \boldsymbol{x}_i$

$\hat{\boldsymbol{\theta}}$ for Eq.(1.16). Assume we have a training set of observations Y_i and a training set of covariates $\boldsymbol{x}_i$ for $i = 1, \dots, I$, the neural network defined in Eq.(1.15) is fitted in order to find the set of parameters $\hat{\boldsymbol{\theta}}$ that minimizes a loss function (or deviance) of choice, defined w.r.t. to the set of observations $\boldsymbol{Y} = (Y_1, \dots, Y_I)$. A common choice in general regression problems is to use to classical sum-of-squared errors as the loss function. As we will see in the following chapters different loss functions can be used to meet the specific requirements of the modeler. However, for this section we stick with the sum-of-squared error for ease of discussion. Therefore, let us define the loss function as:

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{I} L_i = \sum_{i=1}^{I} (Y_i - \boldsymbol{z}^F(\boldsymbol{x}_i)(\boldsymbol{\theta}))^2 = [\boldsymbol{Y} - \boldsymbol{z}^F(\boldsymbol{x})(\boldsymbol{\theta})]^2, \tag{1.17}$$

the objective when training the network is:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \qquad \boldsymbol{L}(\boldsymbol{\theta}) \tag{1.18}$$

The generic approach to minimize $L(\boldsymbol{\theta})$ is by gradient descent, called back-propagation in this setting. The goal of backpropagation is to compute the partial derivative $\partial \boldsymbol{L}/\partial \theta_{j,l}^{(s)}$ and $\partial \boldsymbol{L}/\partial \theta_{j,0}^{(s)}$ of the loss function with respect to any weight or bias in the network, where $\boldsymbol{L} = (L_1, \dots, L_i, \dots, L_I)$. These derivatives are computed iteratively by the algorithm in order to update the parameters in the network the algorithm iterates over until an optimum is found. Because of the compositional form of the model, the gradient can be easily derived using the chain rule for differentiation.
In order to properly discuss the Backpropagation algorithm we introduce $\boldsymbol{\delta}_j^{(s)} = \partial \boldsymbol{L}/\partial a_j^{(s)}$ defined as the error in the j-th neuron in s-th hidden layer before the activation ϕ is applied. As we will see in the following this error is crucial in order to compute the desired derivatives $\partial \boldsymbol{L}/\partial \theta_{j,l}^{(s)}$ and $\partial \boldsymbol{L}/\partial \theta_{j,0}^{(s)}$. For ease of discussion we drop the $\boldsymbol{x}$ notation from $\boldsymbol{z}^{(K:1)}(\boldsymbol{x}_i)$, however all the operations we will discuss are meant to be performed by feeding the entire training set $\boldsymbol{x}$ into the the input layer. The Backpropagation algorithm is based around four fundamental equations, that provide a way to compute $\boldsymbol{\delta}_j^{(s)}$ and the derivatives of the loss mentioned above.

- The first equation defines the error produced in the in the output layer, as:

$$\boldsymbol{\delta}^F = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}^F} = 2 \cdot [\boldsymbol{Y} - \boldsymbol{z}^F(\boldsymbol{\theta})] \cdot \psi'(\boldsymbol{a}^F), \tag{1.19}$$

  for a proof of Eq.(1.19) see Appendix A. The interpretation of Eq.(1.19) is straightforward, the first term in the right hand side measures how fast the loss is changing as a function of the output layer after the activation ψ is applied. While the second term measures how fast the activation reacts to the output neuron $\boldsymbol{a}^F$.

- The second equation consists in expressing the error $\boldsymbol{\delta}_j^{(s)}$ in terms of the error in the next layer:

$$\boldsymbol{\delta}_j^{(s)} = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}_j^{(s)}} = \sum_{l=1}^{q_s} \boldsymbol{\delta}_l^{(s+1)} \theta_{l,j}^{(s+1)} \phi'(\boldsymbol{a}_j^{(s)}), \tag{1.20}$$

  for proof see Appendix A. In words, Eq.(1.20), moves the error backward through the network from layer $(s+1)$ to the j-th neuron in layer s, using the vector of weights and biases connecting the neuron to the next layer.
  By combining Eq.(1.19) and Eq.(1.20) we can compute the error $\boldsymbol{\delta}^{(s)} = (\boldsymbol{\delta}_1^{(s)}, \ldots, \boldsymbol{\delta}_j^{(s)}, \ldots, \boldsymbol{\delta}_{q_s}^{(s)})$ for any layer in the network. We start by using 1.19to obtain $\boldsymbol{\delta}^F$, then we apply Eq.(1.20) to compute $\boldsymbol{\delta}^K$, then we employ again Eq.(1.20) to obtain $\boldsymbol{\delta}^{K-1}$, and so on, all the way back through the network.

- We then define the rate of change in the loss with respect to a given bias in the network $\theta_{j,0}^{(s)}$ as:

$$\frac{\partial \boldsymbol{L}}{\partial \theta_{j,0}^{(s)}} = \boldsymbol{\delta}_j^{(s)}, \tag{1.21}$$

  see proof in Appendix A. We are now able to compute the partial derivative of the loss with respect to any given bias in the network, since we can obtain $\boldsymbol{\delta}_j^{(s)}$ from Eq. 1.20.

- We obtain an equation for the rate of change of the loss with respect to any weight in the network:

$$\frac{\partial \boldsymbol{L}}{\partial \theta_{j,k}^{(s)}} = \boldsymbol{\delta}_j^{(s)} \cdot \boldsymbol{z}_k^{(s-1)}((\boldsymbol{\theta})^{(s-1)}), \tag{1.22}$$

  see proof in Appendix A. We are now aswell able to provide a closed formula to obtain the partial derivative of the loss w.r.t. to any given weight in the network.

The backpropagation equations provide us with a way of computing the gradient of the loss function. The backpropagation algorithm is structured as follows:

1. Define the input $\boldsymbol{x}$ and set $\boldsymbol{z}^0 = \boldsymbol{x}$ as the input layer. Define the set of observations for the response $\boldsymbol{Y}$. Define the maximum number of iterations E (known as epochs) to be performed by the algorithm, hence the algorithm will iterate over for $e = 0, \ldots, E$;

2. Randomly initialize the parameters for the network $\boldsymbol{\theta}(0)$, where the number between brackets represents the number of the iteration i.e. $e = 0$;

3. Feedforward the input $\boldsymbol{x}$ through the network using Eq.(1.14) and (1.15), compute $\boldsymbol{L}(\boldsymbol{\theta}(e))$;

4. Output the error computing $\boldsymbol{\delta}^F$ using Eq.(1.19);

5. Backpropagate the error through all the layers in the network using Eq.(1.19) and Eq.(1.20) and obtain $\boldsymbol{\delta}^{(s)}$ for each $s = K, K-1, \ldots, 1$;

6. Compute the gradient of the loss function with respect to all weights $\partial \boldsymbol{L}/\partial\theta_{j,k}^{(s)}(e)$ and biases $\partial \boldsymbol{L}/\partial\theta_{j,0}^{(s)}(e)$ in the network as in Eq.(1.21) and Eq.(1.22);

7. Update the weights and biases in the network using the gradient obtained in the previous point:

$$\theta_{j,k}^{(s)}(e+1) = \theta_{j,k}^{(s)}(e) - \frac{\gamma_r}{N}\sum_{i=1}^{N}\frac{\partial L_i}{\partial\theta_{j,k}^{(s)}(e)} \tag{1.23}$$

$$\theta_{j,0}^{(s)}(e+1) = \theta_{j,0}^{(s)}(e) - \frac{\gamma_r}{N}\sum_{i=1}^{N}\frac{\partial L_i}{\partial\theta_{j,0}^{(s)}(e)} \tag{1.24}$$

   where $\gamma_r > 0$ is an hyperparameter known as learning rate, that defines the pace at which the network learn a representation of the data. This hyperparameter is chosen sufficiently small so that the update is still in the region of a locally decreasing loss function. Generally a value close to 0.001 is a good default value;

8. Repeat point 3 to 7 until $e = E$ and return the final set of of parameters as the optimal set for the network $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}(e)$.

Examining the algorithm, the name *backpropagation* sounds rather intuitive since we compute the error vectors $\boldsymbol{\delta}^{(s)}$ starting from the last layer backward to the very first layer. The backward movement is a consequence of the fact that the loss is a function of outputs from the network. Hence, to understand how the cost varies with earlier weights and biases, we need to repeatedly apply the chain rule, working backward through the layers to obtain usable expressions.

The backpropagation algorithm described above computes, at each iteration e, the gradient of the loss function for the whole training data $(\boldsymbol{x}, \boldsymbol{Y})$. However, if the number of records in the training data is very large, then the gradient calculation involves high-dimensional matrix multiplications, resulting in long computational times. In practice, to avoid this problem, it is common to employ a learning algorithm known as stochastic gradient descent. The stochastic gradient descent (SGD) method considers for each epoch only a sub-sample (known as *batch*) of the training data. Using the SGD involves a slight change in the algorithm discussed in points 1. through 7., resulting in:

1. Define the maximum number of epochs E for the algorithm, hence the algorithm will iterate over for $e = 0, \ldots, E$;

2. Initialize the parameters for the network $\boldsymbol{\theta}(0)$;

3. Extract a random sample of dimension m $(\boldsymbol{x}^{sub}, \boldsymbol{Y}^{sub})$ from $(\boldsymbol{x}, \boldsymbol{Y})$. Define the input layer as $\boldsymbol{z}^0 = \boldsymbol{x}^{sub}$;

4. Feedforward the input $\boldsymbol{x}^{sub}$ through the network using Eq.(1.14) and (1.15), then compute $\boldsymbol{L}(\boldsymbol{\theta}(e))$;

5. Output the error computing $\boldsymbol{\delta}^F$ using Eq.(1.19);

6. Backpropagate the error through all the layers in the network using Eq.(1.19) and Eq.(1.20), then obtain $\boldsymbol{\delta}^{(s)}$ for each $s = K, K-1, \dots, 1$;

7. Compute the gradient of the loss function with respect to all weights $\partial \boldsymbol{L}/\partial \theta_{j,k}^{(s)}(e)$ and biases $\partial \boldsymbol{L}/\partial \theta_{j,0}^{(s)}(e)$ in the network as in Eq.(1.21) and Eq.(1.22);

8. Update the weights and biases in the network using the gradient obtained in the previous point:

$$\theta_{j,k}^{(s)}(e+1) = \theta_{j,k}^{(s)}(e) - \frac{\gamma_r}{m} \sum_{i=1}^{m} \frac{\partial L_i}{\partial \theta_{j,k}^{(s)}(e)} \tag{1.25}$$

$$\theta_{j,0}^{(s)}(e+1) = \theta_{j,0}^{(s)}(e) - \frac{\gamma_r}{m} \sum_{i=1}^{m} \frac{\partial L_i}{\partial \theta_{j,0}^{(s)}(e)} \tag{1.26}$$

   ;

9. Repeat point 3 to 8 until $e = E$ and return the final set of of parameters as the optimal set for the network $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}(e)$.

1.4.3 Some issue regarding neural networks

Despite their ability to fit more complex data structures, network models have some relevant drawbacks that should be discussed. Indeed training neural networks is quite an art. For this reason, in this subsection, we summarize some critical issues:

- Given the high number of parameters, neural networks are often prone to overfitting, resulting in a model with a good in-sample performance but poor generalization properties to other datasets, i.e., on the out-of-sample data. A handful of techniques to avoid overfitting has been proposed in the literature; for an extensive discussion, see [26]. In this work, we exclusively consider early stopping. In the early stopping procedure, the data is divided between training, testing, and validation set, then the neural network is trained over a large number of epochs on the training set, and its performance in terms of loss function on the validation set is recorded at each step. Once the performance of the model, measured in terms of loss function on the validation set, degrades for a given amount of epochs (defined as *patience*), the training process is stopped. Then the network parameter $\hat{\boldsymbol{\theta}}$ that gives the lowest loss function on the validation set is returned.

- Neural networks have many hyperparameters: learning rate, number of hidden layers, number of neurons in each layer, and the form of the activations ϕ and ψ. Therefore, finding the optimal structure for the network is problematic since a high number of hyper-parameters need to be tuned. Recent advances in data science research propose Population Based Training (PBT) [32], which trains and optimizes a series of networks at the same time, allowing the optimal set-up to be quickly found. Finding the optimal network architecture is beyond the scope of this work as in the following we will consider a standard structure that has already proven to be effective in a similar context, see [53].[7]

- Given a specific network structure, the problem's solution is not unique as the final estimate of $\hat{\boldsymbol{\theta}}$ will depend on the randomly set starting values of $\boldsymbol{\theta}_0$. Hence, different seeds in the algorithm will lead to different parametrization. This is particularly annoying in insurance pricing because taking different seeds for the network algorithm will lead to different results.

- Network models are often criticized for their *black box* nature and their consequent lack of explainability. However, as argued by [8], this is not the main focus of the algorithmic modeling culture (where neural networks belong) which focuses more on training the models to obtain the best possible prediction, while the focal point of the statistical modeling approach is models' explicability. However, as discussed in this work, the literature is progressively filling the gap by providing interpretable tools for Machine Learning models.

1.4.4 The CANN approach

Here we discuss the Combined Actuarial Neural Network Approach (CANN) proposed by [53], which provides a systematic way to improve classical actuarial regressions (such as GLM) using the neural networks toolbox. More specifically, the CANN approach nests a classical actuarial model into a neural network architecture in order to improve the performance given by the first model. Here we combine the two models presented in Eq.(1.3) and Eq.(1.15) i.e. a GLM with a neural network. However, this approach can be extended to any other regression (see Chapter 4). The idea behind this approach is that we start from a basic regression model, say a GLM, and we improve its performance by using a second regression model in order to analyze its residuals, to see whether it finds additional information in those residuals that the first model failed to account for.
The main advantage of the CANN approach is that it combines the high flexibility of the networks with the interpretability of a classical regression, thus providing higher explainability. Besides, according to [64], when the reference regression model is already close to optimal, its maximum likelihood estimator can be used as initialization of the neural network fitting algorithm, then obtaining lower computational

[7] The choice of standard structure for the network models is motivated by a Grid Search analysis, where different network architectures both in terms of network depth, width (number of neurons in each hidden layer) and activation functions have been tested. Such analysis showed no relevant improvement w.r.t. the architecture adopted in [53].

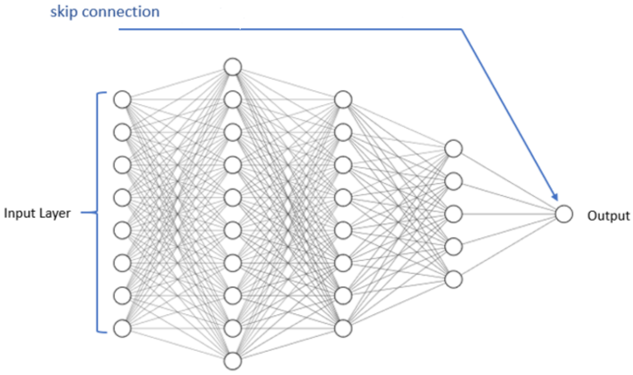


Figure 1.2. CANN architecture

time for the network parameters calibration than a classical neural network model. Formally we define the regression designed by the CANN model as:

$$z^F(\boldsymbol{x}_i)(\boldsymbol{\vartheta}) = \phi(\boldsymbol{x}_i^{'}\boldsymbol{\beta} + \boldsymbol{a}^{(F)}(\boldsymbol{x}_i)), \qquad \text{for } i = 1, 2, \ldots, I, \tag{1.27}$$

where $\boldsymbol{\vartheta}$ is the network parameter and the first term inside the $g^{-1}(.)$ is the GLM regression displayed in Eq.(2.9) while the second term refers to the output layer before activation in Eq.(1.15). Therefore, the CANN regression, combines the models discussed in this section and in Subsection 1.3. Of course, for Eq.(1.27) to make sense, the GLM and the network need to be defined on the same set of observation $(\boldsymbol{Y}, \boldsymbol{x})$.

The embedding produced by the CANN is obtained through a skip connection [8] that links the input given by the GLM estimates to the output layer (see Figure 4.2 for a graphical representation), where the models are merged by summing the two parts. The network parameter $\boldsymbol{\vartheta}$ in for the CANN model consists of:

$$\boldsymbol{\vartheta} = \{\boldsymbol{\beta}, \boldsymbol{\theta}\} = \left\{\boldsymbol{\beta}, \boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(K)}, \boldsymbol{\theta}^{(K+1)}\right\} \tag{1.28}$$

The optimal set for the network parameter $\hat{\boldsymbol{\vartheta}}$ of Eq.(1.27) is obtained training the CANN by minimizing a given loss function L[9]:

$$\underset{\boldsymbol{\vartheta}}{\text{argmin}} \frac{1}{I} \sum_{i=1}^{I} L(Y_i, z^F(\boldsymbol{x}_i)(\boldsymbol{\vartheta})). \tag{1.29}$$

The optimization process to estimate $\boldsymbol{\vartheta}$ works as follows:

1. We first obtain $\hat{\boldsymbol{\beta}}$ parameters via MLE for the GLM in Eq.(2.9);

[8] Skip connections are feed-forward connections that connect one or more elements of a given layer to another element or more elements of a forward but not subsequent layer, thus skipping a given number of hidden layers. In this specific case, the connection skips over all the hidden layers and lands directly into the output layer.

[9] The choice of the loss function has to be coherent with loss minimized by the GLM (or classical model) to boost with the CANN. For instance if the base model is a Poisson GLM, the appropriate choice for the loss is Poisson deviance.

2. then we use such parameters to initialize the network parameter of the Quantile-CANN by considering $\boldsymbol{\vartheta}_0 = \left\{\hat{\boldsymbol{\beta}}, \boldsymbol{\theta}_0\right\}$, where $\boldsymbol{\theta}_0$ is the starting value of the network parameter belonging to the neural network part of model in Eq.(1.27);
3. starting from $\boldsymbol{\vartheta}_0$, we optimize $\boldsymbol{\vartheta}$ minimizing Eq.(1.29) by means of stochasting gradient descent. During the optimization process both the $\boldsymbol{\theta}_0$ parameters and the $\hat{\boldsymbol{\beta}}$ are trained;
4. return $\hat{\boldsymbol{\vartheta}}$.

The gradient descent algorithm investigates the network architecture seeking additional information not captured by the classical regression, returning a lower loss function. If the CANN model does not return any improvement with respect to the GLM, it means that the latter is already able to capture all the relevant information incorporated in the data. In Chapter 4, we extend the CANN approach to the quantile estimation task by combining a Quantile Regression with a neural network.

1.4.5 Preprocessing

In order to feed the training data into a neural network, it is first necessary to pre-process it. Pre-processing consists of a series of operations that are needed to bring the data into a suitable form so that they can enter the network in the input layer $\boldsymbol{z}_0$.

Categorical features in classical regression models are commonly treated via dummy coding, which provides full rank design matrices. For neural network modeling, the full rank property is not essential because we neither have a single (local) minimum of the objective function nor want to calculate the MLE of the network parameter. Typically, in Neural Network modeling one uses one-hot encoding for categorical variables that encodes every level by a unit vector. Let us consider a $I \times p_R$ raw covariate matrix $\boldsymbol{xR}$, assume that the $\boldsymbol{xR}^j$ column vector gathers the observations belonging to a categorical variable j with n different levels $\{a_1, \dots, a_n\}$. One-hot encoding results in:

$$xR_i^j \mapsto \boldsymbol{x}_i^j = \left(\mathbb{I}_{\left\{xR_i^j = a_1\right\}}, \dots, \mathbb{I}_{\left\{xR_i^j = a_n\right\}}\right)^T \epsilon \mathbb{R}^n \tag{1.30}$$

for each $i = 1, \dots, I$. In other terms, One-hot encoding tranforms the $I \times 1$ vector $\boldsymbol{xR}^j$ into an $I \times n$ matrix $\boldsymbol{x}^j$.

However, when the categorical variable has a considerable number of possible values, the dimension of $\boldsymbol{x}^j$ may become difficult to handle since the Neural Network would likely offer a poor representation of the information provided by this variable. In order to avoid this problem, [52] puts forward the idea of using embedding layers[10] to take care of high dimensional categorical variables. This approach has the

[10] Embedding layers are commonly used in Natural Language Processing in order to represent words by numerical coordinates in a low dimensional space.

great advantage of reducing the dimension of the problem compared to one-hot encoding. An embedding layer consists of a layer that we put in front of the network, transforming the n-dimensional inputs from the one-hot encoded variable into an m-dimensional input, where m is a dimension of choice. In other terms, the embedding $\boldsymbol{e}$ is a mapping such that:

$$\boldsymbol{e} : \{a_1, \ldots, a_n\} \mapsto \mathbb{R}^m \tag{1.31}$$

therefore each element $\{a_1, \ldots, a_n\}$ is associated with an m dimensional vector of parameters (or weights) $\{\boldsymbol{e}(a_1), \ldots, \boldsymbol{e}(a_n)\}$, such parameters are normally learned during the model training. Using the embedding introduces $n \times m$ parameters and at same times it lowers the number of parameters by $n \times q_1$.

Continuous covariates do not need pre-processing, but they can directly enter the network, which will take care of representation learning. However, efficient use of gradient descent methods typically requires that all feature components live on a similar scale and that they are roughly uniformly spread across their domains. This makes gradient descent steps more efficient in exploiting the relevant directions. Therefore, in many cases, the so-called MinMaxScaler is used. Let us define with $\boldsymbol{xR}^k$ the column vector gathers the observations belonging to a continuous variable, denote with xR^k_- the minimum value in such vector and with xR^k_+ the maximum. MinMaxScaling consist in the following mapping:

$$xR^j_i \mapsto x^j_i = 2\frac{xR^j_i - xR^k_-}{xR^k_+ - xR^k_-} - 1 \tag{1.32}$$

for $i = 1, \ldots, I$. The resulting feature values x^j_i should roughly be uniformly spread across the interval $[-1, 1]$. If this is not the case, for instance, because we have outliers in feature values, we may first transform them non-linearly to get more uniformly spread values.

1.4.6 Choice of the loss function

As discussed above, Network models require the specification of a loss (or cost) function that is minimized during the training phase of the models. Here we present a general discussion on the importance of choosing the correct loss function, while in the following chapters, we motivate the choice of different loss functions for the different problems at hand.

The standard loss function for regression problems is the sum of square loss already seen in Eq.(1.17) and Eq.(1.29):

$$L(Y_i, f(\boldsymbol{x}_i)) = \sum_{i=1}^{I}(Y_i - f(\boldsymbol{x}_i))^2 \tag{1.33}$$

where Y_i is the observed response and $f(\boldsymbol{x}_i)$ is the prediction produced by a given model for covariates $\boldsymbol{x}_i$. However, when modeling count data for medical claim frequency or right-skewed claim severity data, the sum of squared errors may not be a good choice. In order to clarify this idea, as in [29], we use the concept of deviance, that is defined as $D(Y, f(\boldsymbol{x})) = -2 \cdot ln[\ell(f(\boldsymbol{x}))/\ell(\boldsymbol{Y})]$, consisting in a likelihood ratio where $\ell(f(\boldsymbol{x}))$ is the model likelihood and $\ell(\boldsymbol{Y})$ is the likelihood of the saturated model (i.e., the model in which the number of parameters equals the number of observations). When comparing different models we choose the one returning the lowest deviance on the test data. Following the idea of [60] we consider a loss L function such that $D(\boldsymbol{Y}, f(\boldsymbol{x})) = \sum_i^I L(Y_i, f(\boldsymbol{x}_i)))$.

Let's take for instance the normal (or: Gaussian) deviance, under the assumption of constant variance, we have:

$$D(\boldsymbol{Y}, f(\boldsymbol{x})) = 2\text{ln} \prod_{i=1}^{I} \exp[-\frac{1}{2\sigma^2}(Y_i - f(\boldsymbol{x}_i)))^2] = \frac{1}{\sigma^2}\sum_{i}^{I}(Y_i - f(\boldsymbol{x}_i))^2 \tag{1.34}$$

which consists in a scaled version of the sum of squared errors in Eq. (1.33). Meaning that a loss function based on the squared error is suitable when the normal assumption makes sense.

However, claim frequency and severity data do not follow such distribution. Therefore, in an actuarial context, [66] employs more suitable loss functions inspired by the GLM pricing framework. For instance, claim frequency modeling involves count data, typically assumed to be Poisson distributed in GLM. Therefore, an appropriate loss function is the Poisson deviance, defined as follows:

$$\begin{aligned} D(\boldsymbol{Y}, f(\boldsymbol{x})) = 2\text{ln} \prod_{i=1}^{I} \frac{1}{Y_i\Gamma(\alpha)} \left(\frac{\alpha Y_i}{f(\boldsymbol{x}_i)}\right)^{\alpha} \exp\left(\frac{\alpha Y_i}{f(\boldsymbol{x}_i)}\right) = \\ 2\sum_{i=1}^{I}\left[Y_i\text{ln}\frac{Y_i}{f(\boldsymbol{x}_i)} - (Y_i - f(\boldsymbol{x}_i))\right] \end{aligned} \tag{1.35}$$

As we will discuss in the following chapters, in this work, we will adopt different loss functions that will change w.r.t. to the problem we are trying to solve via neural networks. However, we will always follow a specific guideline: we first define the problem at hand and the corresponding standard model used to solve this problem; we challenge the model using neural networks trained on loss function (or deviance) consistent with the deviance minimized by the reference model. For instance: when challenging a Negative Multinomial GLM for claim frequency modeling, we use a Neural Network trained on a Negative Multinomial deviance see Chapter 2.

Chapter 2

Modeling health insurance claim frequency

In this chapter, we propose a new neural network approach to claim frequency modeling for a set of different health insurance claim types. In particular, we refer to a dataset encompassing three claim types: medical visits, dental care treatments, and diagnostic testing. A typical modeling approach could involve fitting a separate univariate Negative Binomial regression to estimate the frequency for each single claim type. The Negative Binomial is a widespread distributional assumption for health insurance claim counts since it has the great advantage of capturing the overdispersion characterizing such claims, see for instance [31] and [21]. However, using a univariate technique would miss possible correlations between the occurrence of the different claim types, which is often observed in the health insurance realm [18].

For this reason, here, we adopt a multivariate approach. More specifically, we propose Negative Multinomial[1] Neural Networks to estimate the claim frequency for the different claim types. The advantage of this technique is twofold. First, it models the claim frequency of the various claim types via neural network machinery, accounting for non-linearities, covariates interactions, and overdispersion. Second, the multivariate approach provided by the Negative Multinomial distribution allows capturing possible correlations between the different claim types.

The remainder of this chapter is organized as follows: In Section 2.1, we present the health insurance dataset considered in this work, we perform a descriptive analysis on the data, and we motivate our choice of a Negative Multinomial approach. In Section 2.2, we introduce the Negative Multinomial Neural Networks (NM-NN), and we discuss Negative Multinomial GLM (NM-GLM) by [33] that serves as a benchmark to compare the performance of our neural network model. Section 2.3 is devoted to results and discussion, at first we evaluate the in-sample and out-of-sample performance returned by NM-NN w.r.t NM-GLM, then we use a class of model agnostic tools to unveil the data representation learned by the models.

[1] The Negative Multinomial distribution is the multivariate extension of the Negative Binomial distribution, see [54]

2.1 The Data

The dataset considered in this project stems from an Italian insurer and reports the claims collected in a general health insurance plan during 2018. The plan is an employer-based insurance since it provides risk coverage for managers and retired managers belonging to a specific industry in Italy. More specifically, the dataset consists of 132,499 policyholders (employees or former employees). Since each policyholder can also enroll his relatives (parents, spouse, and children below 25 years of age) in the insurance coverage, we totally have 273,950 insured. The dataset covers three classes of claims:

- Medical visits with a wide range of specialist doctors, such as cardiologists, peadiatricians, neurologists and many more.
- Dental care treatments, including, among others, dental braces, implants, and oral surgery.
- Diagnostic tests, e.g. magnetic resonance imaging, blood tests, and electrocardiogram.

For each insured, the dataset reports the following information: number of claims filed during the year (split between *Visits*, *Dentalcare*, and *Diagnostic*), total claim amount per year, a binary variable signaling whether the insured submitted at least a claim during the year, age, gender, regional area, firm dimension, family member type, years of permanence in the coverage, and ID code[2]. Table 2.1 provides a summary for the information available in the dataset. The response variables for the frequency models discussed in Section 2.2 are the claim counts $N_{1,i}, N_{2,i}, N_{3,i}$, while the responses $\mathbb{I}_{N_i}$ and S_i will be employed later on to characterize the models discussed in Chapter 4.
Therefore the dataset is composed of $273,950$ observations and six covariates. Among the insured, we count a total of 205,625 claimants. The monetary volume of the submitted claims is about euro 233 m.
Below, we explore the dataset via summary and descriptive statistics.

2.1.1 Covariates

We start the descriptive analysis by offering a deeper insight into the covariates reported in the dataset. In Figure 2.1 and Table 2.2, we report the histograms and the frequency tables for the covariates in the dataset. The age (**AG**) variable is strongly concentrated at older ages, signaling a fairly senior population. In particular, we notice a 'dip' in the distribution between 25 and 40 years. This 'dip' is due to the specific subscription policies of the Italian insurer: since the policyholders are firm managers, it is unfrequent that they have less than 40 years of age[3]. Moreover,

[2] It is worth mentioning that the dataset displayed above reports no information about the exposure at risk during the year e_i for the insured, hence we assume an exposure equal to one for each insured $e_i = 1$.

[3] The Italian labor market is *seniority driven*, hence it is particularly difficult to become a manager at younger ages.

Variable	**Description**
Claim counts - discrete dependent variable	
$N_{1,i}$ (*Visits* claim counts)	number of medical visits claims submitted by the insured during the year.
$N_{2,i}$ (*Dentalcare* claim counts)	number of dentalcare claims submitted by the insured during the year.
$N_{3,i}$ (*Diagnostic* claim counts)	number of diagnostic claims submitted by the insured during the year.
Covariates	
AG (age)	Age of the insured (in years)
GE (gender)	Gender of the insured (male/female)
PE (permanence)	Years of permanence in the insurance coverage for the insured
RE (region)	Italian region of residence for the insured (categorical variables with 21 classes)
DM (dimension)	Dimension of the company the policyholder is working for (number of employees), for insured different than the policyholder the dataset reports the value of the policyholder
FA (Family member)	Categorical variable reporting the family member type: Policyholder, Spouse, Ex-Spouse, Parents, and Children
Additional information	
Insured ID	Insured identifier used to join this dataset with the data set presented in Chapter 3
Binary dependent variable	
$\mathbb{I}_{N_i}$ (claims binary variable)	Binary variable reporting 1 if the insured filed at least a claim and 0 otherwise.
Positive dependent variable	
S_i (total claim amount)	Total claim amount submitted by the insured during the year (in euros). If the insured submits no claims $S_i = 0$. This variable takes into account the aggregate claim amount for the three claim types.

Table 2.1. Summary of the variables available in the dataset.

managers are not allowed to enroll in the insurance coverage their children above 25 years of age. That is why we observe only a small number of insured between 25 and 40 years. The gender (**GE**) is rather balanced between males and females. The permanence (**PE**) is an integer variable that reports the years of permanence in the insurance plan, its minimum is 1 (for newcomers), and its maximum is 41 (for early adopters). The histogram of this variable shows a decreasing trend. However, there is a strong peak at 38 years, connected to the subscription of the health coverage by a large number of firms whose employees (or pensioners) are still enrolled in the insurance plan. For the region, we observe an intense concentration in 2 of 21 Italian regions: 'Lombardia' and 'Lazio', where most firms have their head office. The dimension variable (**DM**) is a proxy for the firm's dimension the policyholder belongs to. More specifically, the variable reports the number of managers working in the firm. The value of this variable is the same across all the insured belonging to the same family. It ranges from 1 to 1500, with a strong concentration below 100, representing the small to medium-sized firms (that are specific to the Italian economy). The family member type (**FA**) is a categorical variable representing which kind of family member the insured is. From Figure 2.1, we observe that the most relevant classes are: Policyholder, Spouse, and Children. In contrast, Parents and Ex-Spouse are almost negligible since they cover, on aggregate, less than 200 insured in the dataset.

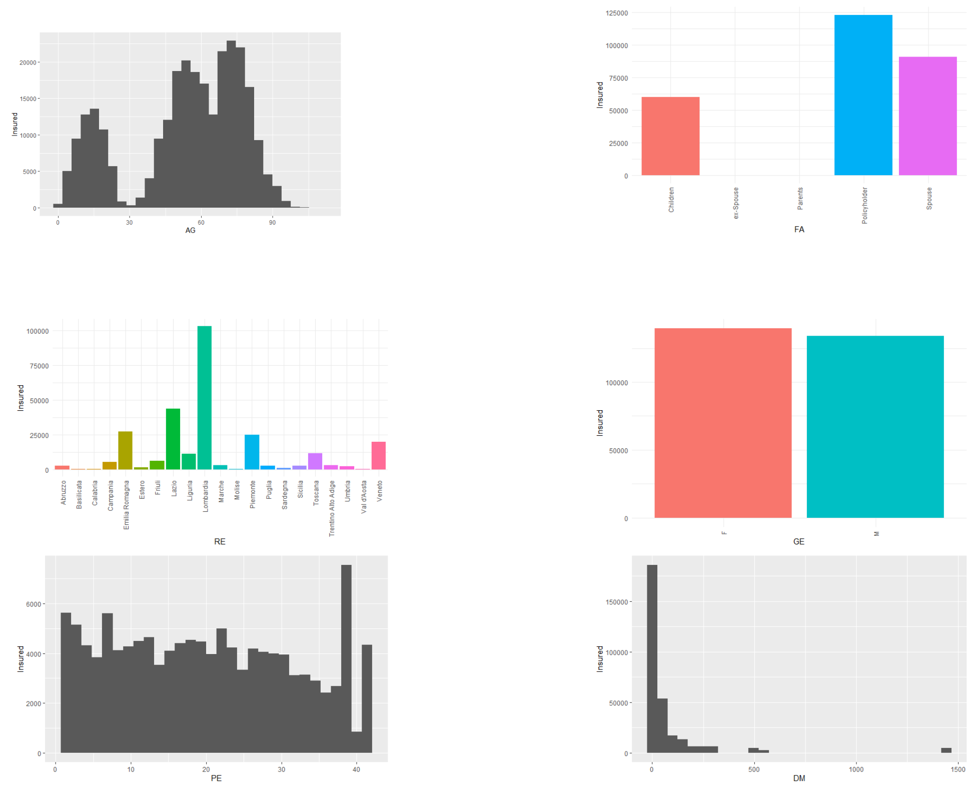


Figure 2.1. Histograms for the covariates in the dataset.

Variable	Absolute frequency	Relative frequency	Variable	Absolute frequency	Relative frequency
AG (age)			**RE** (region)		
$[0, 10)$	15,000	0.054755	Lombardia	103,193	0.376686
$[10, 20)$	32,022	0.11689	Lazio	43,881	0.160179
$[20, 30)$	11,622	0.042424	Emilia Romagna	27,443	0.100175
$[30, 40)$	4,267	0.015576	Piemonte	25,034	0.091382
$[40, 50)$	32,161	0.117397	Veneto	19,931	0.072754
$[50, 60)$	48,274	0.176215	Toscana	11,716	0.042767
$[60, 70)$	45,134	0.164753	Liguria	11,410	0.04165
$[70, 80)$	56,108	0.204811	Others	31,342	0.114408
$[80, 90)$	25,312	0.092396	**DM** (dimension)		
$[90, \infty)$	4,050	0.014784	$[0, 9)$ 116,327	0.424629	
GE (gender)			$[10, 19)$	41,229	0.150498
F	139,664	0.509816	$[20, 49)$	42,267	0.154287
M	134,286	0.490184	$[50, 99)$	25,703	0.093824
PE (permanence)			$[100, \infty)$	48,424	0.176762
$[1, 5)$	49,856	0.181989	**FA** (family member)		
$[5, 10)$	46,074	0.168184	Children	59,886	0.275965
$[10, 15)$	41,499	0.151484	ex-Spouse	43,237	0.199243
$[15, 20)$	34,800	0.12703	Parent	46,224	0.213008
$[20, 25)$	29,501	0.107688	Policyholder	36,067	0.166203
$[25, 30)$	25,619	0.093517	Spouse	24,145	0.111264
$[30, 35)$	19,434	0.07094			
$[35, \infty)$	27,167	0.099168			

Table 2.2. Covariates frequency tables

Before moving on, it is worth discussing some selected associations between covariates reported in Figure 2.2. First, with no surprise, we notice a strong association between the age and family member (top-left pane). For instance, it is evident that Children are younger than Policyholders. The opposite holds for Parents and Policyholders. Therefore the family member type (**FA**) seems a categorical version of the age variable, so dropping the variable would be an option. However, we prefer to keep this variable since, as we will see in Section 2.3, it still provides some valuable information for our models. Age and gender (top-right pane) show almost no association effect since the two curves are almost identical. In the bottom-left sub-plot of Figure 2.2, we report the correlogram for the continuous variables, where we notice a strong positive correlation between age and permanence. A correlation between the two variables is rather obvious since an older insured is more likely to have been in the coverage for a long time. Again, we could consider setting down this variable. However, we prefer to keep also this variable because the information it provides may be helpful to capture the phenomenon of experienced consumers, i.e., insured that have a sound knowledge of the health services provided by the insurance plan and that each year undergoes a set of predefined medical check-ups. The last plot reports the possible associations between age and region for the most frequent regions in our dataset. Some regions show slight differences compared to others, e.g., Liguria seems to have a more aged population. Nevertheless, there is no radical difference between the Italian regions.

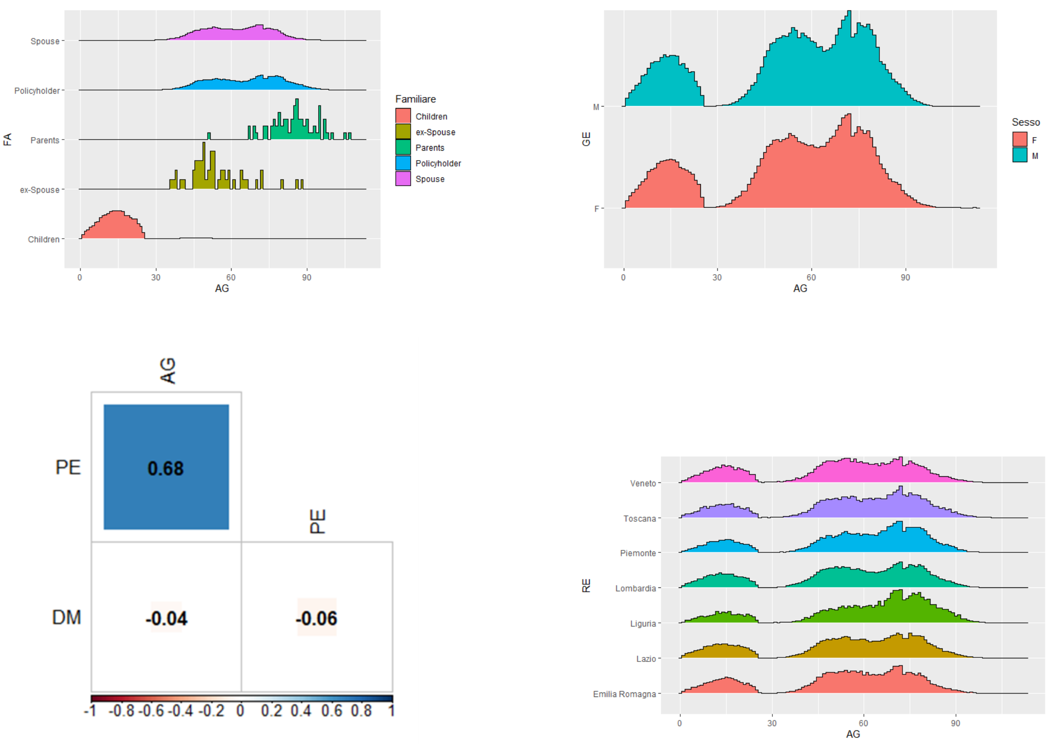


Figure 2.2. Some selected associations between the covariates. We report the **AG** histogram conditional on **FA** (top-left), **GE** (top-right), and some values of **RE** (bottom-right). The bottom-left pane reports the correlogram matrix for the continuous variables in the dataset.

2.1.2 Response variable: claim counts

Here we provide some insight into the response variables employed in the frequency models discussed in the next section. In Figure 2.3, Table 2.3, and Table 2.4 we give a general overview of the different claim counts in the dataset $N_{1,i}$, $N_{2,i}$, and $N_{3,i}$. The figure displays the histograms for the different claim types, Table 2.3 reports their summary statistics, and Table 2.4 displays their frequency tables.

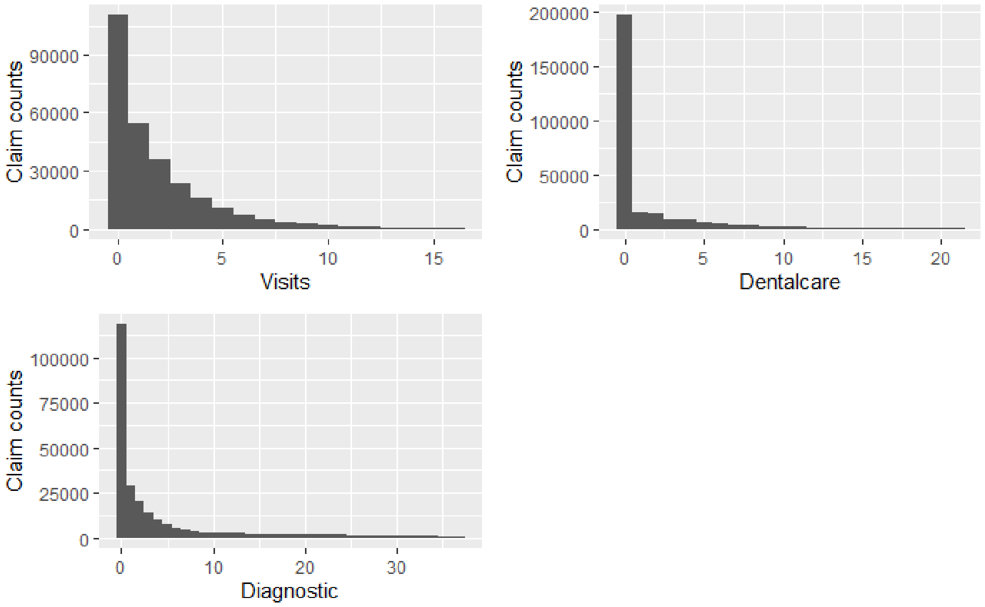


Figure 2.3. Histograms for the different claim counts $N_{1,i}$, $N_{2,i}$, and $N_{3,i}$.

Summary statistics	$N_{1,i}$	$N_{2,i}$	$N_{3,i}$
Median	1	0	1
Mean	1.91	1.59	7.49
95% Quantile	7	9	37
Variance	8.15	19.40	233.08
Claim probability	0.59	0.27	0.56

Table 2.3. Claim counts summary statistics

From Figure 2.3, we observe that the claim counts display a relevant number of zeros. However, taking a closer look at Table 2.3, we notice that a considerable portion of insureds submit a least a claim (last row in the table), which is peculiar to health insurance, where events have a higher frequency w.r.t. other non-life insurance types, i.e., auto insurance or property insurance. More specifically, 50% of the insureds request at least a medical visit or a diagnostic test during the year, while 25% undergo some sort of dental treatment. The average claim frequency for an average insured during the year is 1.91 for medical visits, 1.59 for dental

Variable	Absolute frequency	Relative frequency
$N_{1,i}$ (*Visits* claim counts)		
0	110,397	0.402982
1	54,667	0.199551
$[2, 5)$	85,318	0.311436
$[5, 10)$	18,548	0.067706
$[10, \infty)$	5,020	0.018325
$N_{2,i}$ (*Dentalcare* claim counts)		
0	197,332	0.720321
1	15,851	0.057861
$[2, 5)$	36,075	0.131685
$[5, 10)$	14,051	0.05129
$[10, \infty)$	10,641	0.038843
$N_{3,i}$ (*Diagnostic* claim counts)		
0	118,640	0.433072
1	29,089	0.106184
$[2, 5)$	50,718	0.185136
$[5, 10)$	19,367	0.070695
$[10, \infty)$	56,136	0.204913

Table 2.4. Frequency tables for the response variables

treatments, and 7.49 for diagnostic tests. The frequency for the latter claim type appears to be exceptionally high, and this is also due to the approach used by the insurer to record the claim when an insured undergoes a diagnostic test[4].
The high frequency characterizing such claims generates an overdispersion, as shown in Table 2.3, where the variance is much higher than the mean. To choose the correct discrete distribution for the marginals of the claim counts $N_{1,i}, N_{2,i}, N_{3,i}$, we compare the Poisson distribution with a Negative Binomial distribution by considering two Goodness-of-fit (G.o.f.) measures: the Akaike information criterion (AIC) and the Bayesian information criterion (BIC), see [1] and [43]. The Poisson is a typical distributional assumption when it comes to claim frequency modeling. However, a different distribution may be more appropriate when the data is characterized by overdispersion. The results in Table 2.5 clearly show that the Negative Binomial distribution better describes the claim counts.

Given the nature of the claim types, it is interesting to evaluate the correlation between their occurrence. In Figure 2.4, we report the Spearman correlogram between the claim counts. The results show that diagnostic exams are strongly correlated to visits; this is a mild surprise since the referral given by a medical visit is frequently an essential requirement to undergo an in-depth diagnostic test. Therefore, we choose to jointly model the different claim counts to capture such correlation using a multivariate approach via the Negative Multinomial distribution. Given the low correlation between dental care claims and the other types of claims,

[4] For instance, a blood test claim is split up in each of its components, such as: hemoglobin (Hgb) test, Red cell distribution width (RDW) test, Bilirubin test, Calcium test, and so on. For each single component the insurance company registers a different claim, leading to inflated claim counts.

Type of claim	G.o.f. measure	Poisson	Negative Binomial
Visits $N_{1,i}$	AIC	1,340,309	1,017,746
	BIC	1,340,320	1,017,767
Dentalcare $N_{2,i}$	AIC	1,772,291	729,530
	BIC	1,772,302	729,551
Diagnostic $N_{3,i}$	AIC	5,723,392	1,428,572
	BIC	5,723,402	1,428,593

Table 2.5. G.o.f. statistics for the distribution of the different claim counts. We report the Akaike and Bayesian information criterions. A lower value for the information criterion signals a better fit.

an argument could be made for modeling the dental claims alone while using a multivariate approach for visits and diagnostic tests. However, for simplicity, we still decide to model the three kinds of claims jointly.

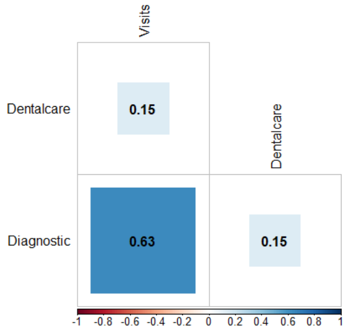


Figure 2.4. Correlogram for the claim counts

In the next section we introduce the Negative Multinomial regression framework that we will use to model the claim frequencies of visits, dental treatments and diagnostic tests.

2.2 Multivariate approach for claim frequency modeling

This section discusses the models employed to model the claim frequencies in a multivariate setting. At first, a theoretical framework for the Negative Multinomial distribution is provided. Then we explore Negative Multinomial GLMs, a standard modeling technique for multivariate counts presented in [33]. At last, we present a novel neural network approach to multivariate claim frequency modeling: Negative Multinomial Neural Networks.

2.2.1 Negative Multinomial distribution

The Negative Multinomial distribution, extensively discussed in [54], provides a model for positively correlated multivariate count data characterized by overdispersion, i.e., where the variance is greater than the mean. Regression models relying on this distributional assumption have already been implemented in different fields, such as genomics [33] and medical statistics [2].

More formally, let us consider an r-dimensional vector of counts $\mathbf{N}_i = (N_{1,i}, \dots, N_{r,i})$, in our case we have $r = 3$ for the claim counts $\mathbf{N}_i = (N_{1,i}, N_{2,i}, N_{3,i})$ presented in Table 2.1. The probability mass for $\mathbf{N}_i$ under a negative multinomial distribution with parameters $\mathbf{p} = (p_1, \dots, p_{r+1})$ and shape parameter α, where $\sum_{j=1}^{r+1} p_j = 1$ and $\alpha > 0$, is:

$$f(\mathbf{N}_i/\mathbf{p}, \alpha) = \frac{\Gamma(\alpha + \sum_{j=1}^{r} N_{j,i})}{\Gamma(\alpha) \prod_{j=1}^{r} N_{j,i}!} \cdot \prod_{j=1}^{r} p_j^{N_{j,i}} p_{r+1}^{\alpha} \tag{2.1}$$

Where the vector of parameters $\mathbf{p} = (p_1, \dots, p_{r+1})$ is composed as:

$$p_j = \frac{\mu_{j,i}}{\alpha + \sum_{k=1}^{r} \mu_{k,i}}, \qquad \text{for } j = 1, \dots, r, \qquad p_{r+1} = \frac{\alpha}{\alpha + \sum_{k=1}^{r} \mu_{k,i}} \tag{2.2}$$

with $\boldsymbol{\mu}_i = (\mu_{1,i}, \dots, \mu_{r,i})$ as the mean parameter vector.

For ease of discussion we set $m_i = \sum_{j=1}^{r} N_{j,i}$ and rearrange the probability mass function in Eq. 2.1 as:

$$f(\mathbf{N}_i/\boldsymbol{\mu}_i, \alpha) = \frac{\Gamma(\alpha+m_i)}{\Gamma(\alpha)\prod_{j=1}^{r} N_{j,i}!} \left(\frac{\alpha}{\alpha+\sum_{j=1}^{r} \mu_{j,i}}\right)^{\alpha} \prod_{j=1}^{r} \left(\frac{\mu_{j,i}}{\alpha+\sum_{k=1}^{r} \mu_{k,i}}\right)^{N_{j,i}} \tag{2.3}$$

As shown in [2], the expectation of the count random variable $\mathbf{N}_i$ characterized by a Negative Multinomial distribution is defined as

$$E(\mathbf{N}_i) = \boldsymbol{\mu}_i = (\mu_{1,i}, \dots, \mu_{r,i}) \tag{2.4}$$

and its covariance matrix is

$$\text{Cov}(\mathbf{N}_i) = \alpha \cdot \left[\text{diag}\left(\frac{\mathbf{p}}{p_{r+1}}\right) + \left(\frac{\mathbf{p}}{p_{r+1}}\right)\left(\frac{\mathbf{p}}{p_{r+1}}\right)^T\right] \tag{2.5}$$

The Negative Binomial can be viewed as a particular case of the Negative Multinomial, where $r = 1$. Resulting in the following probability mass function:

$$f(N_i/\mu_i, \alpha) = \frac{\Gamma(\alpha + N_i)}{\Gamma(\alpha)N_i!} \left(\frac{\alpha}{\alpha + \mu_i}\right)^{\alpha} \left(\frac{\mu_i}{\alpha + \mu_i}\right)^{N_i} \tag{2.6}$$

with

$$E(N_i) = \mu_i, \qquad \text{and} \qquad Var(N_i) = \mu_i + \frac{\mu_i^2}{\alpha} \tag{2.7}$$

It worth noting that, for both Negative Multinomial and Binomial, if the dispersion parameter α is unknown the distribution does not belong to the exponential family. Fitting the distribution involves estimating the parameters $\boldsymbol{\mu}$ and α presented in Eq. 2.3, which are usually obtained via MLE. Thus it is paramount to define the Log-likelihood of the distribution $l(.)$. More formally, given I independent data points $\mathbf{N}_i$ we have:

$$\begin{aligned} l(\mathbf{N}, \boldsymbol{\mu}, \alpha) = & \sum_{i=1}^{I} \log[\Gamma(\alpha + m_i)] - \sum_{i=1}^{I} \log[\Gamma(\alpha)] - \sum_{i=1}^{I} \log(\prod_{j=1}^{r} N_{j,i}!) \\ & -\alpha \sum_{i=1}^{I} \log(\alpha + \sum_{j=1}^{r} \mu_{j,i}) + \sum_{i=1}^{I} \sum_{j=1}^{r} N_{j,i} \log \left(\frac{\mu_{j,i}}{\alpha + \sum_{k=1}^{r} \mu_{k,i}} \right) \end{aligned} \tag{2.8}$$

Below, we briefly discuss the Negative Multinomial GLM that serves as a benchmark for our neural network model.

2.2.2 Negative Multinomial GLM (NM-GLM)

The general framework for GLMs with a Negative Multinomial distribution for the response variable was first introduced in [33]. This approach models the relationship between the multivariate response count variable and a set of covariates capturing the inner correlation structure between counts.

Following the structure presented in Section 1.3, the Negative Multinomial GLM builts upon the following hypothesis:

- Distributional hypothesis: the multivariate 3-dimensional[5]claim count responses $\mathbf{N}_1, \ldots, \mathbf{N}_I$ with $\mathbf{N}_i = (N_{1,i}, N_{2,i}, N_{3,i})$ are i.i.d. following the Negative Multinomial distribution displayed in Eq. 2.3;

- Structural hypothesis: the relationship between the expectation $E(\mathbf{N}_i/\boldsymbol{x}_i)$ and the p-dimensional vector of covariates $\boldsymbol{x}_i$ is represented as:

$$\boldsymbol{\mu}_i = \exp(\boldsymbol{x}_i'\boldsymbol{\beta}), \qquad \text{for } i = 1, 2, \ldots, I, \tag{2.9}$$

where $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3)$ is the $p \times 3$ regression parameters matrix. Then, from Eq. 2.7 and 2.2 we have:

$$E(\mathbf{N}_i/\boldsymbol{x}_i) = \exp(\boldsymbol{x}_i'\boldsymbol{\beta}), \qquad \text{for } i = 1, 2, \ldots, I, \tag{2.10}$$

and the vector $\mathbf{p} = (p_1, p_2, p_3, p_4)$ in Eq. 2.1 is given by:

$$p_j = \frac{\exp(\boldsymbol{x}_i'\boldsymbol{\beta}_j)}{\alpha + \sum_{k=1}^{3} \exp(\boldsymbol{x}_i'\boldsymbol{\beta}_k)}, \qquad \text{for } j = 1, \ldots, 3, \qquad p_4 = \frac{\alpha}{\alpha + \sum_{k=1}^{3} \exp(\boldsymbol{x}_i'\boldsymbol{\beta}_k)} \tag{2.11}$$

[5] Note that from now on we drop the generic r notation for the dimension of the multivariate response and we directly refer to the tridimensional nature of the claim types discussed in Section 2.1

where $\boldsymbol{\beta}$ and α is the set of parameters to be estimated via Maximum Likelihood. Then, the covariance matrix is obtained feeding back Eq. 2.11 in Eq. 2.5.

The set of parameters $\boldsymbol{\beta}$ and α is obtained maximizing the following loglikelihood:

$$\begin{aligned} l(\boldsymbol{\beta},\alpha) = & \sum_{i=1}^{I}\log[\Gamma(\alpha+m_i)] - \sum_{i=1}^{I}\log[\Gamma(\alpha)] - \sum_{i=1}^{I}\log(\prod_{j=1}^{3} N_{j,i}!) \\ & -\sum_{i=1}^{I}(m_i+\alpha)\log(\alpha+\sum_{j=1}^{3}\exp(\boldsymbol{x}_i'\boldsymbol{\beta}_j)) + \sum_{i=1}^{I}\sum_{j=1}^{r} N_{j,i}\cdot \boldsymbol{x}_i'\boldsymbol{\beta}_j \end{aligned} \tag{2.12}$$

where $m_i = \sum_{j=1}^{3} N_{j,i}$.
The authors [33] propose the use of an iteratively reweighted Poisson regression (IRPR) scheme in order to achieve the MLE of the regression model.
Since the GLM, in this context, works as a benchmark model for the neural network, it is worth keeping in mind that maximizing $l(\boldsymbol{\beta},\alpha)$ in Eq. 2.12 is equivalent to minimizing the deviance $D^{\text{NM}}(\boldsymbol{N},\boldsymbol{\beta},\alpha)$:

$$D^{\text{NM}}(\boldsymbol{N},\boldsymbol{\beta},\alpha) = 2\cdot\sum_{i=1}^{I}\left[-(m_i+\alpha^{-1})\log\left(\frac{1+m_i}{1+\sum_{j=1}^{3}\exp(\boldsymbol{x}_i'\boldsymbol{\beta}_j)}\right)+\sum_{j=1}^{3}N_{j,i}\left(\frac{\log(N_{j,i})}{\boldsymbol{x}_i'\boldsymbol{\beta}_j}\right)\right] \tag{2.13}$$

The same deviance is used to train neural network presented in the next subsection. Once we have the estimated matrix of regression parameters $\hat{\boldsymbol{\beta}}$ we obtain the vector of expected claim frequencies as:

$$E^{\text{glm}}(\mathbf{N}_i/\boldsymbol{x}_i) = \Big(E^{\text{glm}}(N_{1,i}/\boldsymbol{x}_i), E^{\text{glm}}(N_{2,i}/\boldsymbol{x}_i), E^{\text{glm}}(N_{3,i}/\boldsymbol{x}_i)\Big) = \exp(\boldsymbol{x}_i'\hat{\boldsymbol{\beta}}), \tag{2.14}$$

for $i = 1, 2, \ldots, I$. Where the superscript 'glm' denotes that the expectation is obtained via a GLM.

2.2.3 Negative Multinomial Neural Networks (NM-NN)

The Neural network approach we propose consists of a feed-forward neural network with three output layers in order to model the multivariate claim count response $\mathbf{N}_i$ w.r.t. to the features set $\boldsymbol{x}_i$. The higher flexibility of Neural Networks provided by their intricate inner structure should account for the dependence that characterizes the different types of claim counts. Given our set of covariates $\boldsymbol{x}_i$ and considering a network on dept K, we can rearrange the expression for the output layer already shown in Eq. 1.15 of Chapter 1 as:

$$\boldsymbol{z}^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta}) = \psi(\boldsymbol{\theta}_0^{K+1}+\textstyle\sum_{l=1}^{q_K}\boldsymbol{\theta}_l^{K+1}z_l^{(K:1)}(\boldsymbol{x}_i)) = \psi(\boldsymbol{a}^{(F)}(\boldsymbol{x}_i)), \qquad \text{for } i = 1,\ldots,I, \tag{2.15}$$

where $\boldsymbol{z}^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta}) = (z_1^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta}), z_2^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta}), z_3^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta}))$ is the tridimensional output produced by network for the $\boldsymbol{x}_i$ string of observations. Note that in this specific formulation $\boldsymbol{\theta}_0^{K+1}$ is a tridimensional vector of output biases and $\boldsymbol{\theta}_l^{K+1}$ is the tridimensional vector of weights connecting the l-th neuron in the last hidden layer to the output layer. While $z_l^{(K:1)}(\boldsymbol{x}_i)$ in Eq. 2.15 can be expressed as in Eq. 1.14. In the expression above, we denote with $\boldsymbol{\theta}$ the full set of weights for the network gathering the matrix of parameters of each layer:

$$\boldsymbol{\theta} = \left\{\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(s)}, \cdots, \boldsymbol{\theta}^{(K)}, \boldsymbol{\theta}^{(K+1)}\right\} \tag{2.16}$$

For a visual representation of the Network with $K = 2$, see Figure 2.5

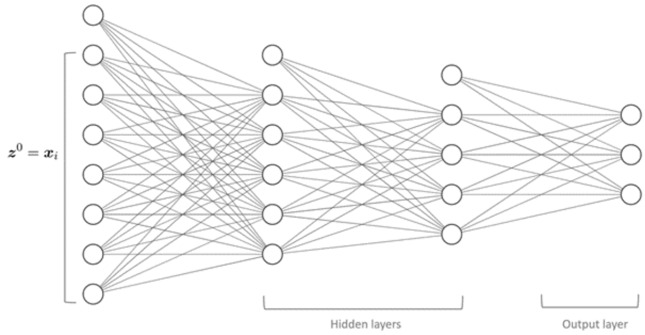


Figure 2.5. Negative Multinomial Neural Network architecture for with $K = 2$, $q_0 = 7, q_1 = 5, q_2 = 4$, and $\boldsymbol{z}^0 = \boldsymbol{x}_i$

As discussed in Section 1.4, the network has to be trained to minimize a given loss function. Following the idea exposed in [60], when training a machine learning model, it is crucial to choose a proper loss function in order to solve the problem at hand. Since our goal is to model multivariate correlated count data, a logical choice for the loss function is the Negative Multinomial Deviance. Rearranging Eq. 2.13 we have:

$$D^{\text{NM}}(\boldsymbol{N}, \boldsymbol{\theta}) = 2 \cdot \sum_{i=1}^{I} \left[-(m_i + \alpha^{-1})\text{log}\left(\frac{1+m_i}{1+\sum_{j=1}^{3} z_j^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta})}\right) + \sum_{j=1}^{3} N_{j,i}\text{log}\left(\frac{N_{j,i}}{z_j^{\text{NM}}(\boldsymbol{x}_i)(\boldsymbol{\theta})}\right)\right], \tag{2.17}$$

the objective when training the network is:

$$\underset{\boldsymbol{\theta}}{\text{argmin}} \qquad D^{\text{NM}}(\boldsymbol{N}, \boldsymbol{\theta}) \tag{2.18}$$

thus the optimal set of parameters $\hat{\boldsymbol{\theta}}$ is obtained minimizing the Negative Multinomial deviance via Stochastic Gradient Descent. Note that in the estimation process the scale parameter α is considered as a given, and is obtained by preemptively performing a Negative Multinomial GLM on the same set of data $(\mathbf{N}, \boldsymbol{x})$.

Given the optimal set of parameters $\hat{\boldsymbol{\theta}}$ we can compute the expected number of claims as:

$$E^{\text{nn}}(\mathbf{N}_i/\boldsymbol{x}_i) = (E^{\text{nn}}(N_{1,i}/\boldsymbol{x}_i), E^{\text{nn}}(N_{2,i}/\boldsymbol{x}_i), E^{\text{nn}}(N_{3,i}/\boldsymbol{x}_i)) = \boldsymbol{z}^{\text{NM}}(\boldsymbol{x}_i)(\hat{\boldsymbol{\theta}}) \tag{2.19}$$

for $i = 1, 2, \ldots, I$. Where the apex 'nn' denotes that the expectation is obtained via a Neural Network.

In the following section we explore the results obtained in the application of the models discussed in this section.

2.3 Results and discussion

To evaluate the general performance of the NM-NN w.r.t. the benchmark NM-GLM, we test the model over the dataset presented in Section 2.1. The results discussed in this section are obtained through five-fold cross-validation, where the dataset is divided into five-folds, and at each iteration, three of the five folds are used as a training set for the models (NM-NN and NM-GLM), one as a validation set, and one as a testing set.

The Network model is trained over $2,000$ epochs using early stopping on the validation set in order to avoid overfitting. The model adopts a five hidden layer structure of dimension $(50, 40, 30, 20, 10)$ with the ReLu (see Eq. 1.9) as the activation function. As for the variables presented in Table 2.1: **AG**, **PE**[6], **DM** are Min-Max scaled, **RE** and **FA** are treated using a $d = 1$ embedding layer, and **GE** is one-hot encoded. As for the GLM model, **AG**, **PE**, and **DM** are treated as continuous variables, while **RE**, **FA**, and **GE** are dummy encoded.

2.3.1 Performance

The performance of each model is measured in terms of Negative Multionomial deviance (see Eq. 2.13 and Eq. 2.17), where a lower deviance signals a better model. For both models, we estimate the claim frequencies for medical visits, dental treatments, and diagnostic tests. Figure 2.6 compares the in-sample (left pane) and the out-of-sample (right pane) performance for the NM-GLM and the NM-NN. In particular, we report the Negative Multinomial deviance over the five data folds. The results are very stable over the different folds, both in-sample and out-of-sample. In fact, the Neural Network model consistently achieves a better performance w.r.t. GLM since it returns a lower deviance.

The results illustrated in Figure 2.6 suggest that using a Neural Network approach can be appealing. Neural networks are often celebrated for their outstanding predictive performance since they easily learn a good representation of the training data that generalize well to new data, of course, if carefully trained. However, such models often face one major criticism: the lack of explainability. Indeed, neural networks have a huge number of parameters and a complex inner structure made up of several hidden layers that make it very difficult for the modeler to understand the results. To overcome such limitations, in recent years, a vast literature covering the topic of model agnostic tools has flourished, see [24], and for an actuarial case study [39] or [29], aiming at providing interpretative tools for Machine Learning models. The corresponding literature has a plethora of well-established model agnostic techniques. In this work, we exploit Permutation Variable Importance [9] to gauge the relevance of each covariate in the model, Individual Conditional Expectation curves and Partial Dependence plots to showcase the marginal effect of a covariate over the prediction produced by the model, and the H-Squared statistic [24] in order to spot

[6] This variable is first log transformed. The log transformation of this variable is used also for the NM-GLM.

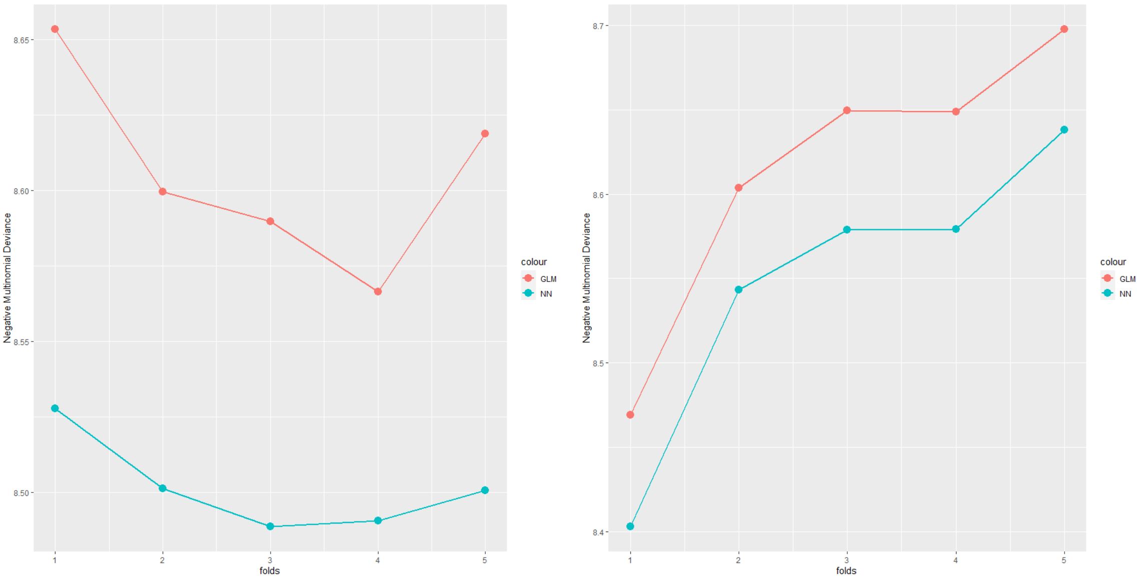


Figure 2.6. Performance for both NM-GLM and NM-NN, in-sample (left pane) and out-of-sample (right pane). On the y-axis we report the Negative Multinomial deviance, while the x-axis stores the reference data fold.

potential interaction effects between the covariates.

2.3.2 Permutation Variable Importance

Permutation Variable Importance by [9] measures the increase in the deviance of a model after permuting the values of a given covariate. The basic idea of this technique is quite simple: the importance of a covariate in a given model is measured by computing the increase in the model's deviance after permuting the covariate values. In other terms, the variable importance is the increase in model deviance when the information provided by an explanatory variable is destroyed. Such variable is deemed important if randomly shuffling its values increases the model deviance because, in this case, the model relied on the variable for the prediction. While a covariate is redundant if shuffling its values produces little to no increase in the model's deviance.

In what follows the general framework for the algorithm.

Let $f(\boldsymbol{x})$ be the generic prediction function given by a model, where $\boldsymbol{x}$ is the covariate matrix, y is the vector of observations and $D(y, f(\boldsymbol{x}))$ is the model's deviance. For instance, for the NM-NN we have $f(\boldsymbol{x}) = \boldsymbol{z}^{\text{NM}}(\boldsymbol{x})(\hat{\boldsymbol{\theta}})$ and $D(y, f(\boldsymbol{x})) = D^{\text{NM}}(\boldsymbol{N}, \hat{\boldsymbol{\theta}})$.

1. We estimate the original model Deviance $D_0 = D(y, f(\boldsymbol{x}))$;

2. For each covariate $j = 1, ..., p$:

    - take the covariate matrix $\boldsymbol{x}_i$, that can also be represented as $\boldsymbol{x}_i = \{\boldsymbol{x}_{.,j}, \boldsymbol{x}_{.,-j}\}$, where $\boldsymbol{x}_{.,j}$ is the column vector belonging to covariate j and

$\boldsymbol{x}_{.,-j}$ is the matrix for the other covariates. Then get the set of covariates $\boldsymbol{x}^{perm}$ by permuting the values in $\boldsymbol{x}_{.,j}$;

- estimate model deviance $D_{perm} = D(y, f(\boldsymbol{x}^{perm}))$;
- compute the Permutation Variable Importance for covariate j as $I_j = D_{perm} - D_0$

3. Sort covariates by descending order I_j for $j = 1, ..., p$.

Permutation Variable Importance can be either performed on the training set or the test set. Performing the analysis on the test set informs on how much the model relies on the covariate for its predictions. In contrast, using the test set would hint at the relevance of the covariate for the model's performance on new and unseen data.

In Figure 2.7, we report the Variable Importance metric to find the most relevant variables in our claim frequency models. The variables are ranked from top to bottom, from the most important to the less relevant. For both models, the two most relevant variables are age (**AG**) and region (**RE**). However, their ranking is different: the age variable is by far the most important variable for NM-NN, while it only achieves second place in NM-GLM. In particular, the increase in deviance is much higher in the Network model (0.12) than in the GLM (0.06). Signaling that probably the GLM is missing some information when modeling the relationship between the age variable and the claim frequencies. As for the regional variable, even though it scores first in the NM-GLM and second in the NM-NN, it has almost the same importance for the two models 0.07. In both models, the other variables are far less relevant, however, some of them seem to have a somewhat higher importance in Network models with respect to the GLM (**GE** and **PE**), with the exception of the family member variable (**FA**) which is more relevant in the GLM than in the neural network. Note that even though the some covariates seem irrelevant in the GLM, their paramaters appear to be significant as shown in Appendix A.

2.3.3 Individual Conditional Expectation and Partial dependence

Individual Conditional Expectations (ICE) profiles are a valuable tool to study the marginal effect of a covariate over the response provided by the model. For a given covariate j, such a profile shows how the prediction provided by the model for an observation $obs_i = \boldsymbol{x}_i$ reacts when the covariate $\boldsymbol{x}_{i,j}$ slides over its range of possible values.

In particular, producing ICE profiles for a set of observations gives a hint on how the response evolves with respect to the different values of the variable. An ICE plot represents the relationship between the prediction and a specific covariate for every single observation separately, producing one profile (or line) per observation. The values for a line of the data matrix are obtained by fixing the values of all other covariates and creating variants of this observation by replacing the covariate's value with values coming from a grid and producing predictions with the model for these new observations. For a specific observation, this procedure results in a set of points

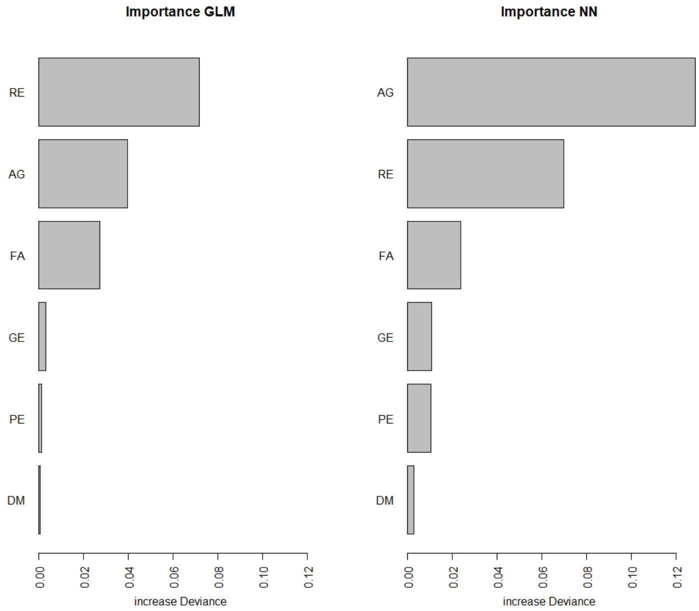


Figure 2.7. Variable Importance for GLM (left) and Negative Multinomial Neural Network (right). The results reported are obtained on the last testing fold of the five-fold crossvalidation.

given by the feature value from the grid and the respective predictions.
From an algorithmic standpoint, we have:

1. take an observation $obs_i = \boldsymbol{x}_i$ and the corresponding prediction $f(\boldsymbol{x}_i)$ given by the model.

2. $\boldsymbol{x}_i$ is a p dimensional vector that can also represented as $\boldsymbol{x}_i = \{\boldsymbol{x}_{i,j}, \boldsymbol{x}_{i,-j}\}$, where j is the covariate we want to study.

3. Consider the grid of V possible values $(\mathrm{v}_1, \dots, \mathrm{v}_V)$ for the selected covariate j. Then for each v_k with $k = 1, ..., V$ we repeat:

   - $\boldsymbol{x}_{i,j} = \mathrm{v}_k$.
   - $obs_i = \boldsymbol{x}_i = \{\mathrm{v}_k, \boldsymbol{x}_{i,-j}\}$.
   - $ICE^j_{i,\mathrm{v}_k} = f(obs_i)$.

   Once the process ends, we obtain a curve $\left\{ICE^j_{i,\mathrm{v}_k}\right\}_{k=1}^{V}$ corresponding to $\{\mathrm{v}_k, \boldsymbol{x}_{i,-j}\}_{k=1}^{V}$.

The process is repeated potentially for each observation in the dataset and each variable.
The ICE profiles are helpful to highlight the presence of interactions. The stronger the interaction effects associated with the variable j, the greater the differences in shape observed across ICE profiles. However, this model agnostic tool does not

reveal with which other variable the interaction arises. Note that, by construction, ICE profiles of a given covariate j are parallel as long as the underlying model does not incorporate interactions (like in a GLM on the log scale transformation).

Friedman's Partial Dependence (PD) profiles [24] are obtained averaging different ICE profiles of a given variable. PD profiles can be viewed as the main effect of covariate j merged over the whole set of observations. In other terms, they represent the average effect of variable j and can display whether the relationship between the response and a covariate is linear, monotonic, or more complex.

Partial dependence marginalizes the model's prediction over the distribution of the covariates in $\boldsymbol{x}_{\cdot,-j}$, so that the profile displays the relationship between variable j and the output produced by the model.

If we consider the ICE profiles obtained for variable j over a set of n observations, ICE^j_{i,v_k} where $i = 1, ..., n$ and $k = 1, ..., V$, the partial dependence profile is obtained as:

$$PD^j = \left(PD^j_{\mathrm{v}_1}, \ldots, PD^j_{\mathrm{v}_V}\right), \qquad \text{with} \qquad \left\{PD^j_{\mathrm{v}_k} = \frac{1}{n}\sum_{i=1}^{n} ICE^j_{i,\mathrm{v}_k}\right\}_{k=1}^{V}, \tag{2.20}$$

where, again, V is the grid of possible values for the selected covariate. The $PD^j_{\mathrm{v}_k}$ function, for a given value v_k of variable j, reveals the average marginal effect on the prediction returned by the model. The computation of partial dependence plots is intuitive: the partial dependence function at a particular covariate value represents the average prediction if we force all data points to assume that feature value. Note that PD profiles are not restricted to a single variable. It is also possible to consider multiple variables at the same time in order to study their joint effect on the response.

We analyze the marginal effect for the different covariates considered in the NM-GLM and the NM-NN in the following lines. In particular, for each dependent variable in the models, we will discuss their marginal effect on the claim frequencies of medical visits, dental treatments, and diagnostic tests.

In particular we observe:

Visits: Figure 2.8 displays the partial dependence plots produced by the neural network and the GLM for the *Visits* claim frequency, where each pane reports the PD plot for a different covariate. We notice a first significant difference when comparing the PD plot for the age variable (**AG**). The marginal effect captured by GLM (in red) is exponential. It starts at a claim frequency of 0 and caps at 5. In contrast, the behaviour captured by the NM-NN (in blue) is more complex. This PD plot starts at 2, and then the curve decreases to its minimum around 15 years of age, reaching a frequency of 1. Then the curve starts slowly increasing then followed by a substantial increase after the age of 50, reaching a maximum of 3.3 at around 80 years. Therefore the marginal effect produced by the Network captures specific features, such as the higher frequency of medical visits associated with younger ages connected to pediatric visits and the steady claim frequency at older ages. The PD plot for the family member covariate (**FA**) shows a similar marginal effect for the Policyholder and Spouse values for both models. However, the PD plot corresponding to the GLM assigns a higher frequency to the children value since the model attempts at capturing the phenomenon of pediatric visits already captured by the Network model with age. The region effect is somewhat different for the two models. The GLM captures a strong claim frequency for 'Lazio', which does not happen for the neural network. In contrast, the gender covariate (**GE**) has the same effect in both models. The effects for the permanence (**PE**) and dimension (**DM**) are pretty similar, and their trend seems to be relatively flat (maybe with a slight increase towards higher values), hence with a negligible main effect over the claim frequency of medical visits.

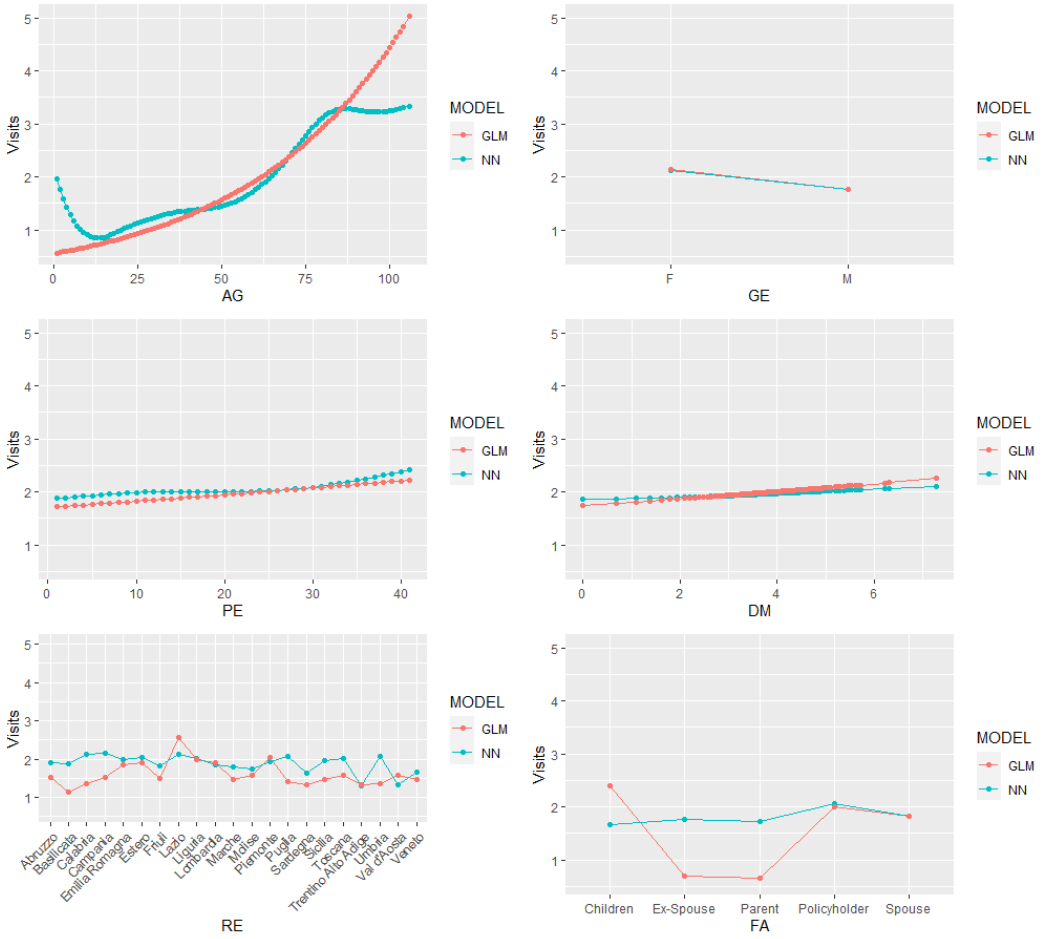


Figure 2.8. Partial dependence plots for Visits

Dentalcare: Figure 2.9 reports the PD plots for *Dentalcare* claim frequency. Also this kind of claim shows some different behaviours for the PD plot age. The GLM's PD plot shows an exponential trend, while the marginal effect produced by the neural network is almost parabolic, capping at 75 years of age, with a slight bump around 15 years of age. The bump is connected to dental braces and teenage oral surgery. Looking at the PD plots for the **FA**, we notice that the effect of this variable for the NM-NN is almost flat, while the GLM effects change across the different types of family members. Similar to *Visits* claim frequency, the GLM tries to capture the effects not registered by the **AG** PD plots. The region marginal effect is rather similar for the two models, even though some regions have different effects. The gender covariate (**GE**) has precisely the same effect in both models. The effects for the permanence (**PE**) and dimension (**DM**) are pretty similar, and their trend seems to be relatively flat, hence with a negligible main effect over the claim frequency of medical visits. For instance, the PD plots for (**DM**) range between 1.63 and 1.68

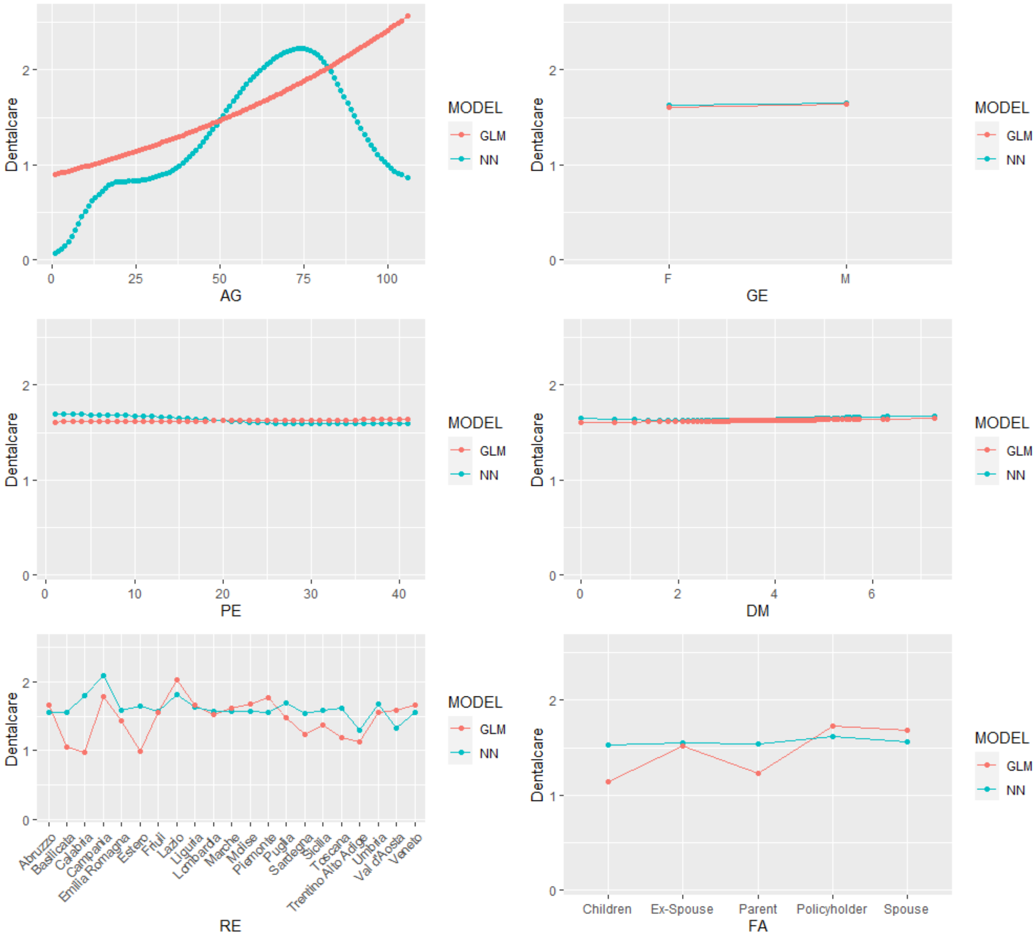


Figure 2.9. Partial dependence plots for Dentalcare

Diagnostic: Figure 2.10 reports the PD plots for the Diagnostic claim frequency. The gender (**GE**) variable has almost the same effect for the GLM and the neural network. The same holds for the region (**RE**) and the dimension (**DM**). The age (**AG**) has a quasi-linear trend in the neural network with two small bumps around the thirties and eighties, while the trend is exponential in the GLM. As for the visits and dental care, the **FA** GLM PD plot displays different values for Children, Parents, and Former Spouse, to capture the effect not captured by the age variable. The neural network PD plot for permanence (**PE**) shows a substantial effect on the claim frequency for a high value of the permanence, while the GLM fails at doing so.

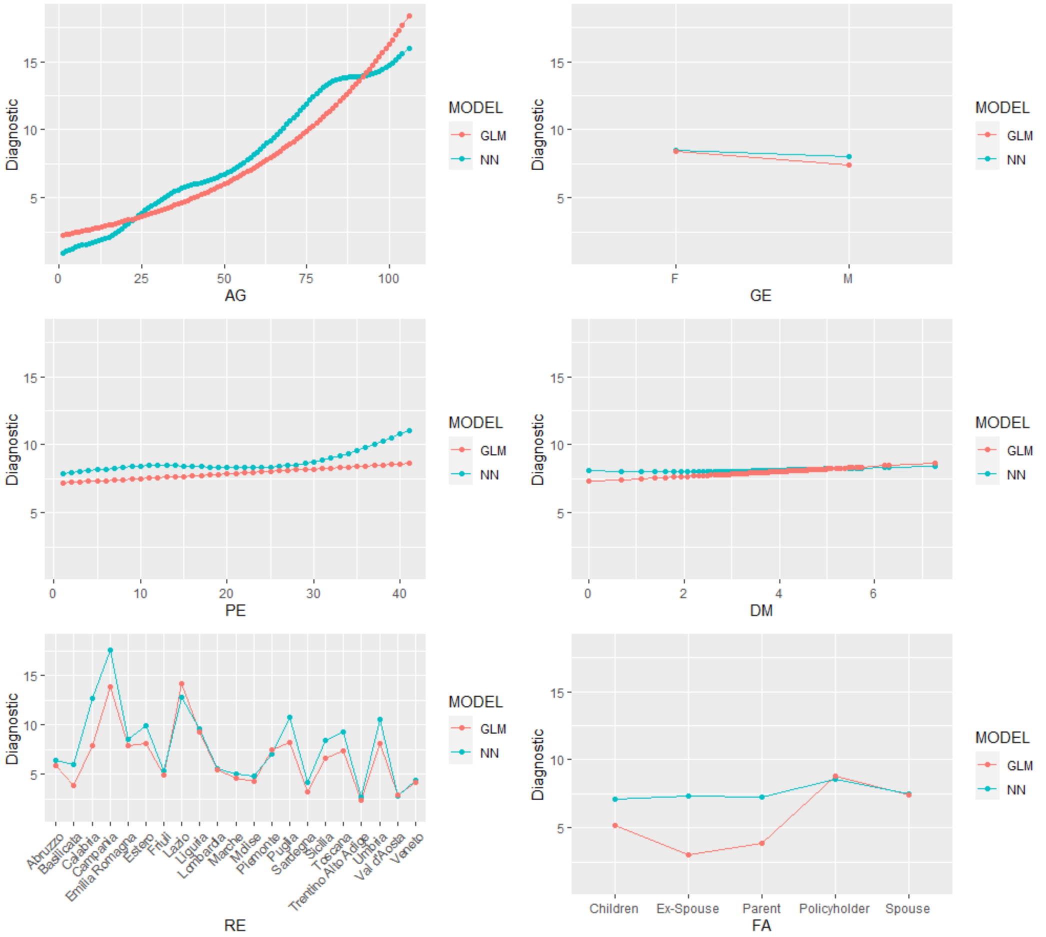


Figure 2.10. Partial dependence plots for Diagnostic

As shown via the different PD plots, the major difference between the two models is related to the age **AG** main effect. This difference is primarily due to the GLM's structural form, which lacks the flexibility to capture the shape of the main effect. In contrast, the NM-NN seems to capture all the information provided by the age variable. The NM-GLM recovers some information via the family member **FA** main effect. In particular, the model gives a higher (*Visits*) or lower (*Dentalcare*) frequency to Children, thus capturing part of the age main that was lacking in the GLM. Therefore the first reason for the performance gap in Figure 2.6 is probably due to a poor modelization of the **AG** main effect from the GLM. This issue could be addressed using a polynomial variate or a spline. However, this is only part of the story since the different performances may also be associated with possible interactions between variables, which the PD plot cannot detect.

2.3.4 Studying interactions

After having a look at main effects, here we closely study possible interaction effects between covariates captured by the Network model. A model agnostic way to measure the interaction between two variables is based on partial dependence profiles introduced by [24]. In particular, when a given model incorporates an interaction effect, its predictions cannot be expressed as the sum of its variables' main effects because the effect of one variable depends on the value of another variable. In particular, considering the PD properties, if two covariates do not interact, the joint Partial Dependence function between $\boldsymbol{x}_{\cdot,j}$ and $\boldsymbol{x}_{\cdot,k}$ can be decomposed as:

$$\bar{PD}^{jk}(\boldsymbol{x}_{i,j}, x_{i,k}) = \bar{PD}^{j}(x_{i,j}) + \bar{PD}^{k}(x_{i,k}) \tag{2.21}$$

where $\bar{PD}^{j}$ and $\bar{PD}^{k}$ are the centered[7] partial dependence function belonging to covariates j and k. Whereas if the two variables interact with each other in the model, the joint PD cannot be expressed as the sum of the two marginal effects as in (2.21).
Therefore, to assess the presence of interaction effects, we adopt the H-statistic introduced by [24], which estimates the interaction strength between two covariates by measuring how much of the prediction variance originates from their interaction. The H-statistic to measure the interaction strength between covariates j and k, is defined as follows:

$$H^2_{jk} = \textstyle\sum_{i=1}^{n} \left[\bar{PD}^{jk}(x_{i,j}, x_{i,k}) - \bar{PD}^{j}(x_{i,j}) - \bar{PD}^{k}(x_{i,k})\right]^2 / \textstyle\sum_{i=1}^{n}(\bar{PD}^{jk}(x_{i,j}, x_{i,k}))^2 \tag{2.22}$$

where the sums run over a subset of n randomly selected observations, and $\bar{PD}^{k}$ is the centered version of the PD profile for variable k, and $\bar{PD}^{jk}$ is the centered Two-way PD for variable j and k. In other words, H^2_{jk} measures the proportion of variability in the joint effect of $\boldsymbol{x}_{\cdot,j}$ and $\boldsymbol{x}_{\cdot,k}$ unexplained by their main effects. A value close to zero indicates almost no pairwise interaction, while a value close to one means that most effects come from the pairwise interaction. The H-statistic can also be larger than 1, this can happen when the variance of the joint interaction is larger than the variance of the 2-dimensional partial dependence plot.

In Figure 2.11, we plot the values of the H-statistic in Eq. 2.22 for the NM-NN and each possible pairwise interaction between covariates. We do not report the plot for the GLM since the model is not designed to capture pairwise interactions between variables. Each pane in Figure 2.11 reports the interactions detected for the different claim types: for *Visits* the age **AG** has a weak interaction with both permanence **PE** and gender **GE**; *Dentalcare* claim frequency shows two strong interactions for the permanence **PE** variable with dimension **DM** and gender **GE**; *Diagnostic* claims have two relevant pairwise interactions for the gender **GE** variable with permanence **PE** and dimension **DM**, moreover we also notice a small interaction between age **AG** and region **RE**.

[7] The centered version for the Partial Dependence profile is obtained via Eq. 2.20 where instead of using the ICE profiles discussed above we consider a set of ICE profiles that are centered around zero.

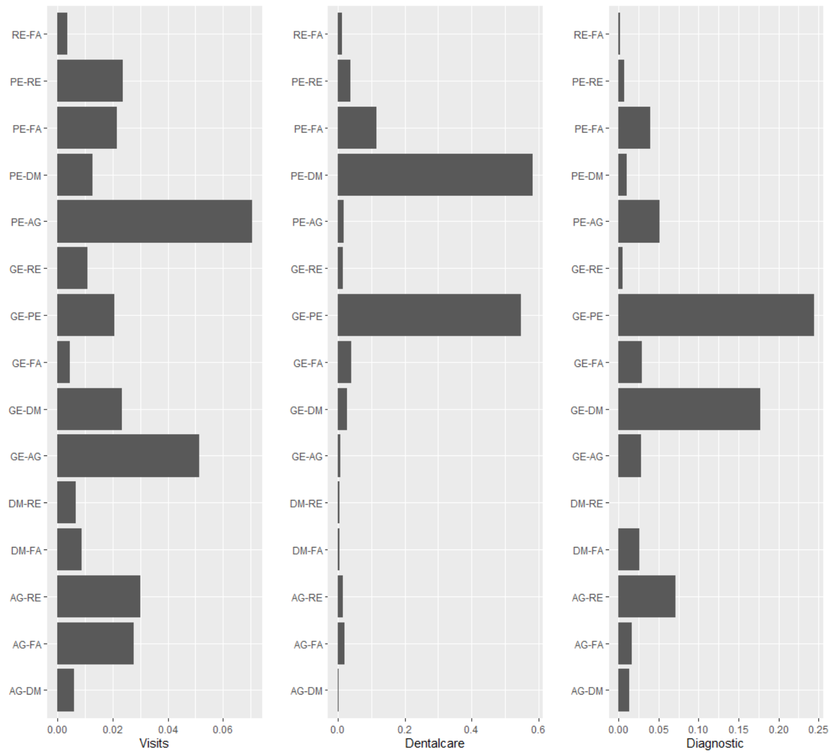


Figure 2.11. H-statistic - for the NM-NN

To gain insight into the behaviour on such interaction effects, we use grouped PD plots to visualize the effect given by the interaction on claim frequencies in the following lines. A grouped PD plot represents the main effect of a variable conditioned on the different values of another variable. Therefore the plot reports v PD curves where v is the number of possible values for the conditioning covariate. The interaction appears to be relevant if the curves display a different behaviour when conditioned to the different values of the conditioning variable. In particular, we expect the different PD curves to be non-parallel.

Visits: The interaction between gender **GE** and age **AG** displayed in Figure 2.12 displays a different behaviour for age PD plot conditioned on females. In particular, we notice a higher claim frequency for female insured between 20 and 60 years of age. This increased frequency is probably associated with gynecologic visits and pregnancy-related visits.

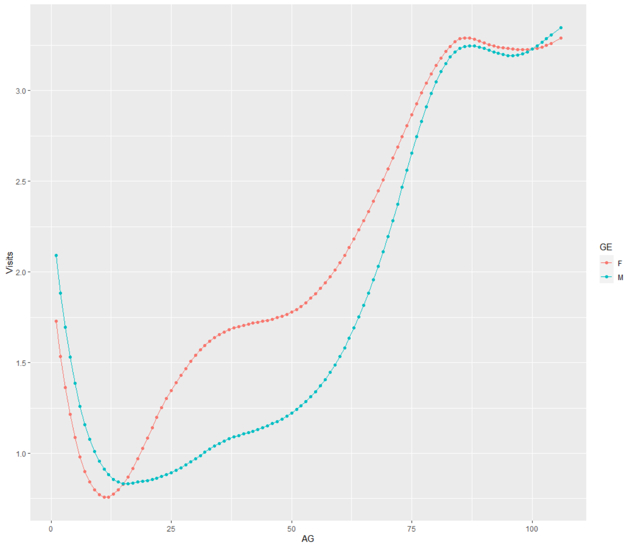


Figure 2.12. *Visits* claim frequency Grouped PD plots for **AG** and **GE** produced by the NM-NN.

In Figure 2.13 we study the interaction between age **AG** and permanence **PE** for the *Visits* claim frequencies. We notice that insured aged between 15 and 75 have an increasing *Visits* claim frequencies w.r.t. permanence **PE**, while insured below 15 and above 75 years of age have higher frequencies when the value for the permanence is low.

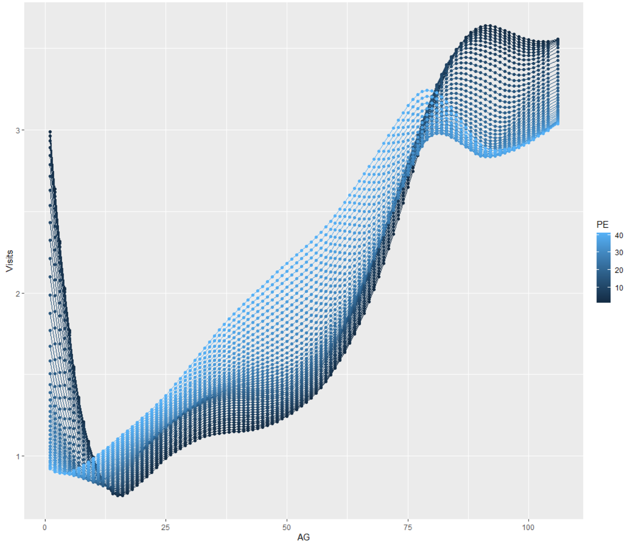


Figure 2.13. *Visits* claim frequency Grouped PD plots for **AG** and **PE** produced by the NM-NN.

Dentalcare: The interaction between the dimension **DM** variable and gender **GE** in Figure 2.14 shows two different effects for males and females. When the insured is a male, the dimension has a negligible effect on his claim frequency. In contrast, when the insured is a female, the dimension has a decreasing effect on claim frequencies. In other terms, for the NM-NN, the dimension variable only has some significant effect for females, while for males, the effect of this variable seems irrelevant. It is also worth noting that the Group PD plot ranges 1.55 and 1.77, which is a somewhat narrow interval.

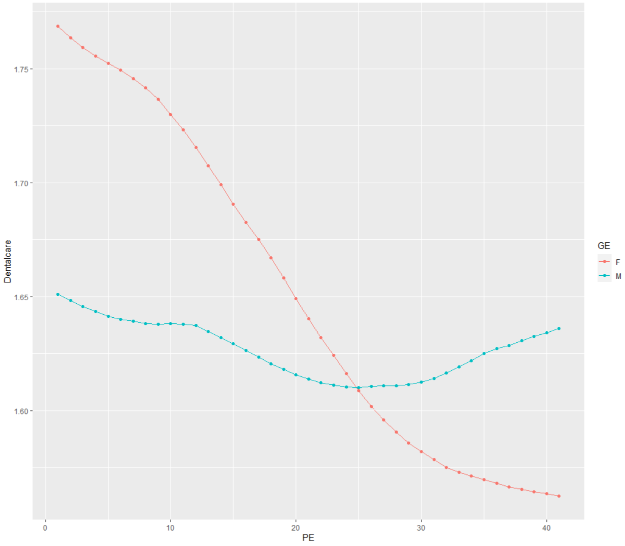


Figure 2.14. *Dentalcare* claim frequency Grouped PD plots for **DM** and **GE** produced by the NM-NN. Here the dimension variable is reported on a log-scale.

Figure 2.15 displays the Grouped PD plots for gender **GE** and dimension **DM**. All the conditional main effects reported in the figure show a positive relationship between the claim frequency and both the company's dimension and the years of permanence within the insurance plan. However, it is possible to observe a different slope for the PD plots conditioned on a permanence below 20 and above 20. The latter group of plots has a timidly growing trend, while a steep increase characterizes the first one. Thus, having a low permanence in a big company seems to have a boosting effect on the number of dental treatment claims submitted by the insured; this may often be the case when big corporates have commercial partnerships with dental clinics.

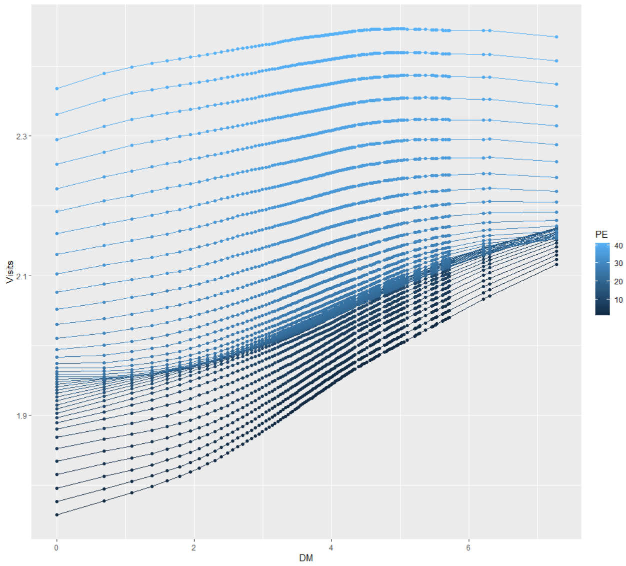


Figure 2.15. *Dentalcare* claim frequency Grouped PD plots for **DM** and **PE** produced by the NM-NN. Here the dimensione variable is reported on a log-scale.

Diagnostic: The interaction between the dimension **DM** and gender **GE** is exposed in Figure 2.16. In this case, gender has an opposite effect on the PD plot produced by the dimension. Conditioning to females induces a definite increasing trend in the **DM**main effect, while having a male insured produces a slightly decreasing trend. Thus, this interaction assigns a higher diagnostic tests claim frequency to females insured from big-sized firms.

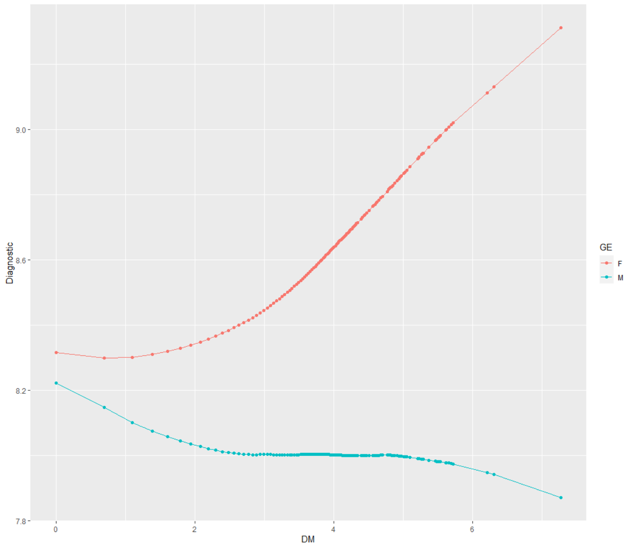


Figure 2.16. *Diagnostic* claim frequency Grouped PD plots for **DM** and **GE** produced by the NM-NN. Here the dimensione variable is reported on a log-scale.

The interaction between the age **AG** variable and region **RE** in Figure 2.17 displays a plethora of different effects w.r.t. the different Italian regions. It is interesting to notice a steep increase of the claim frequency, at older ages, for Lazio, which has an inefficient regional health service with long waiting lists. This lack in the public supply is often supplemented by the private sector, which is partially financed by health insurance. The opposite is true for some virtuous regions, such as Veneto. The effect induced by the 'bad' regions is stronger when the insured is old and needs diagnostic tests. Figure 2.18 displays the Grouped PD plots for gender **GE** and permanence **PE**. In both curves the trend is increasing. However, males have a steeper increase w.r.t to females.

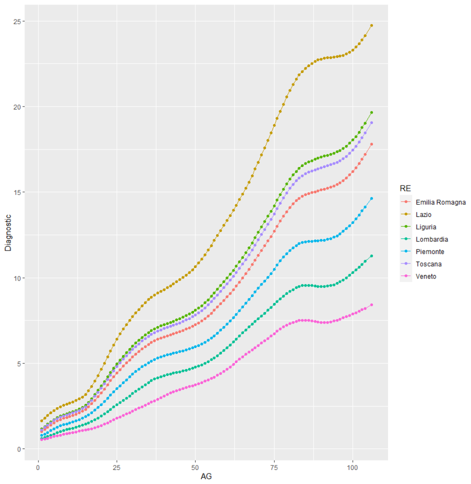


Figure 2.17. *Diagnostic* claim frequency Grouped PD plots for **AG** and **RE** produced by the NM-NN. We only report the PD plots belonging to relevant regions.

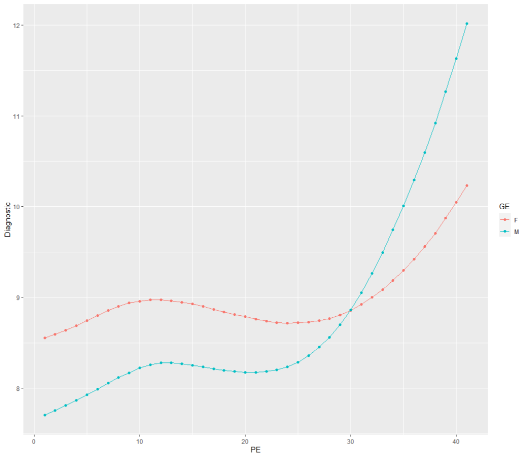


Figure 2.18. *Diagnostic* claim frequency Grouped PD plots for **PE** and **GE** produced by the NM-NN.

2.3.5 Final remarks

From the results discussed in this section, the Negative Multinomial Neural Network appears as a clear winner over the benchmark GLM. The network model performs better in-sample and out-of-sample. These results are achieved thanks to the neural network's flexibility, which is not restricted by a multiplicative structural form like the GLM. The network architecture offers sufficient complexity so that it is capable of reflecting non-linearities in the explanatory variables and interactions between them. As mentioned during the discussion, the GLM's performance could greatly benefit from using polynomial variates or splines to improve the modeling of main effects. Moreover, it would also be possible to improve further the GLM by plugging manually the different pairwise interactions spotted by the NM-NN. In this sense, neural networks can also serve as a complementary tool for GLMs. Where, in a first step, the modeler using a neural network spots the weaknesses of the simpler regression model (GLM, GAM, or others), such as missing interactions or main effects. Then, in a second step, the modeler improves the simpler regression model via brute force modeling of its functional form. To complete the frequency-severity approach discussed in Section 1.2, we will assess the possible merits of Neural Networks for claim cost modeling in the next chapter.

Chapter 3

Modeling claim severity

This chapter addresses the traditional actuarial problem of claim severity estimation by proposing a novel Gamma regression model based on neural networks. This model is employed to compute the claim severities for the health insurance claims presented in Section 2.1. Following the same reasoning discussed in the previous chapter, we compare our model with a Gamma GLM, which is the standard approach when modeling claim severity. The expected claim severities obtained using the two techniques are then used to complete the Frequency-Severity model (Section 1.2) involved in the calculation of the pure premium for each insured in the dataset.

This chapter is organized as follows: In Section 3.1, we briefly report some descriptive statistics for the claim severities corresponding to the dataset presented in Section 2.1; Section 3.2 provides a short discussion of the different models employed in this chapter (Gamma-GLM and Gamma-Neural Networks); In Section 3.3, following the same structure of Section 2.3, at first we evaluate the in-sample and out-of-sample performance returned by Gamma-Neural Network w.r.t Gamma-GLM, then we employ the set of model agnostic tools presented in Section 2.3 to gain insight on the data representation captured by the models; In Section 3.4 we compute the pure premiums, and we assess them using a set of metrics proposed in[15].

3.1 Data description:

Recalling the definition given in Eq. 1.2, the severity for a generic claim j submitted by policyholder i is denoted as $\tilde{S}_{i,j}$. The claim dataset presented below, reports the cost of each single claim submitted by the portfolio of insured discussed in Section 2.1. In particular, the claim dataset, has three different type of claims: *Visits*, *Dentalcare*, and *Diagnostic*. Table 3.1 reports the information available in such dataset.

This chapter is interested in claim severity estimation for the different claim types conditional on the policyholder's characteristics $\boldsymbol{x}$. The matrix $\boldsymbol{x}$ is not immediately available in the dataset presented in Table 3.1. Nonetheless, it is possible to retrieve it by connecting the Insured ID in the table above to the Insured ID presented in Table 2.1. Thus, the set of covariates (or risk factors) considered in the regression models discussed in the next section is the same as the one described in Section 2.1. Before moving to the actuarial modeling of claim severities, in this section, we

Variable	Description
Positive dependent variable	
$\tilde{S}^1_{ij}$ (*Visits* claim severity)	*Visits* claim severity (EUR) for the j-th claim submitted by policyholder i.
$\tilde{S}^2_{ij}$ (*Dentalcare* claim severity)	*Dentalcare* claim severity (EUR) for the j-th claim submitted by policyholder i.
$\tilde{S}^3_{ij}$ (*Diagnostic* claim severity)	*Diagnostic* claim severity (EUR) for the j-th claim submitted by policyholder i.
Additional information	
Insured ID	ID number for the insured that allows to connect the claim dataset to the dataset presented in 2.1 and described in Table 2.1.

Table 3.1. Summary of the variables available in the claim dataset.

provide some descriptive statistics to characterize the claim costs. In Figure 3.1, we report the claim severity histograms for the different claim types, Table 3.2 presents the corresponding frequency tables, and in Table 3.3 we display some summary statistics. The distributions of the different claim severities are right-skewed (see Figure 3.1). This feature is commonly well described through a Gamma distribution. In particular, we notice that *Diagnostic* and *Dentalcare* claims seem far more skewed than medical visits. In particular, dental treatments are characterized by a fat tail between 500 and 1000 euros, probably due to oral surgery claims that have a higher cost.

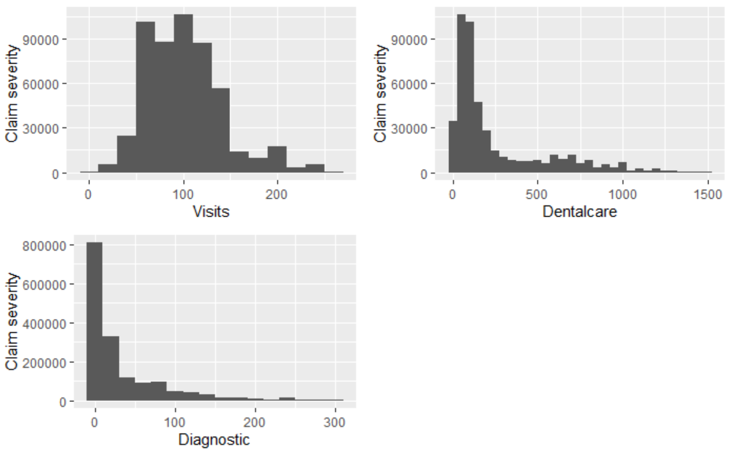


Figure 3.1. The plots report the claim severity trimmed histograms for medical visits, dental treatments and diagnostic tests.

Unlike the claim frequency models discussed in the previous chapter, we will not model the claim severities using a multivariate approach since there is non-correlation between the cost of the different claims (see Figure 3.2). Therefore, we will work with a different Gamma regression model for each type of claim.

Variable	Absolute frequency	Relative frequency
$\tilde{S}_{ij}^{1}$ (*Visits* claim severity)		
(0-50]	29,775	0.056
(50-100]	262,426	0.501
(100-150]	172,013	0.328
(150-200]	39,592	0.075
(200-1000]	18,566	0.035
(1000,∞)	1,285	0.002
$\tilde{S}_{ij}^{2}$ (*Dentalcare* claim severity)		
(0-100]	211,652	0.460
(100-200]	100,241	0.218
(200-500]	60,656	0.131
(500-1000]	71,817	0.156
(1000-1500]	11,528	0.025
(1500,∞)	3,861	0.008
$\tilde{S}_{ij}^{3}$ (*Diagnostic* claim severity)		
(0-30]	1,127,155	0.685
(30-50]	119,646	0.072
(50-100]	220,297	0.134
(100-200]	131,331	0.079
(200-1000]	43,412	0.026
(1000,∞)	1,330	0.000

Table 3.2. Frequency tables for the response variables

Summary stats	Visits	Dentalcare	Diagnostic
Min.	0,01	1	0,01
1st Qu.	70	60	4
Median	100	120	10,5
Mean	105,75	266,53	38,70
3rd Qu.	130	330	49
Max.	7800	10100	18080
95%	198,39	950	150

Table 3.3. Claim cost summary statistics

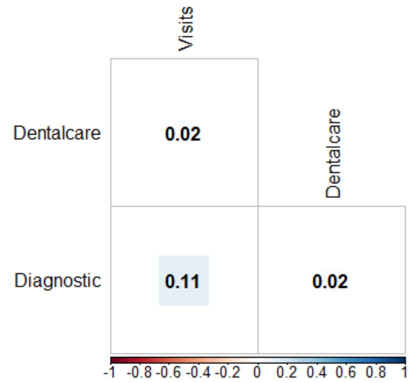


Figure 3.2. Correlogram for the claim severities

Note that we do not distinguish between regular and large claims in this study, and we consider all of them together. However, a case could be made for using two different approaches when modeling small and large claims, as in [4] and [14]. We explore the different gamma regression models employed for claim severity estimation in the following lines.

3.2 Regression models for claim severity estimation:

This section provides a formal description of the two competing regression models employed in the claim severity estimation. We first report a short description of Gamma GLM, which is the go-to technique for claim cost modeling, then we introduce the novel Gamma Neural Networks.

3.2.1 Gamma GLM

We define with $\tilde{S}_{i,j}$ the cost for the j-th generic[1] claim submitted by policyholder i, and we assume the claim severities $\tilde{S}_{i,j}$ with $i = 1, \dots, I_p$ and $j = 1, \dots, N_i$ to be independent. Where I_p is the number of policyholders with positive claims (i.e. $N_i > 0$) and N_i is the number of claims submitted by insured i during the year (that we assume to be a positive integer). Here we change the notation from N_i to n_i to emphasize that the number of claims is treated as a known quantity. The Gamma GLM assumes that the cost of the individual claim $\tilde{S}_{i,j}$ is distributed as:

$$\tilde{S}_{i,j} \sim f(\tilde{S}_{i,j}/\lambda_i, \alpha) = \frac{\lambda_i}{\Gamma(\alpha)}(\lambda_i \tilde{S}_{i,j})^{\alpha-1} e^{-\lambda_i \tilde{S}_{i,j}} \tag{3.1}$$

[1] For ease of discussion, here, we refer to a generic claim $\tilde{S}_{i,j}$, but the concepts exposed in the rest of the section still apply to $\tilde{S}^1_{i,j}$, $\tilde{S}^2_{i,j}$, or $\tilde{S}^3_{i,j}$.

where $\lambda_i > 0$ is the rate parameter and $\alpha > 0$ is the constant shape paramater. The mean and variance for the distribution are defined as:

$$E(\tilde{S}_{i,j}) = \frac{\alpha}{\lambda_i} = \mu, \qquad \text{and} \qquad Var(\tilde{S}_{i,j}) = \frac{\alpha}{\lambda_i^2} = \mu^2\phi \tag{3.2}$$

where $\phi > 0$ is the dispersion parameter. Using the expectation in Eq. 3.2 it is possible to reparametrize the distribution in Eq. 3.1:

$$\tilde{S}_{i,j} \sim f(\tilde{S}_{i,j}/\mu, \alpha) = \frac{\exp^{-\frac{\alpha}{\mu}\tilde{S}_{i,j}}}{\Gamma(\alpha)} \left(\frac{\alpha}{\mu}\tilde{S}_{i,j}\right)^{\alpha} \frac{1}{\tilde{S}_{i,j}} \tag{3.3}$$

Referring to Eq.1.4, in a GLM, the relationship between the conditional expectation $E(\tilde{S}_{i,j}/\boldsymbol{x}_i)$ and the set of covariates $\boldsymbol{x}_i$ is defined as:

$$E(\tilde{S}_{i,j}/\boldsymbol{x}_i) = \mu_i = \exp(\boldsymbol{x}_i'\boldsymbol{\beta}), \quad \text{for, } i = 1, 2, \ldots, I_p, \quad \text{and} \quad j = 1, \ldots, n_i. \tag{3.4}$$

where we set $g(.) = \log(.)$.
Given the vector of observations for the individual claims $\tilde{\boldsymbol{S}} = (\tilde{S}_{1,1}, \ldots, \tilde{S}_{1,n_1}, \ldots, \tilde{S}_{I_p,1}, \ldots, \tilde{S}_{I_p,n_{I_p}})$, the set of regression parameters $\boldsymbol{\beta}$ is obtained maximizing the loglikelihood:

$$l(\boldsymbol{\beta}, \alpha, \boldsymbol{S}) = \textstyle\sum_{i=1}^{I_p} \sum_{j=1}^{n_i} \left\{ \alpha \left[-\frac{\tilde{S}_{i,j}}{\exp(\boldsymbol{x}_i'\boldsymbol{\beta})} - (\boldsymbol{x}_i'\boldsymbol{\beta}) \right] - \ln(\Gamma(\alpha)) + \alpha\ln(\tilde{S}_{i,j}) - \ln(\tilde{S}_{i,j}) \right\} \tag{3.5}$$

which translates in the minimization of the following deviance

$$D^G(\boldsymbol{\beta}, \boldsymbol{S}) = -2\left[l(\boldsymbol{\mu}, \alpha, \boldsymbol{S}) - l(\boldsymbol{S}, \alpha, \boldsymbol{S})\right] = -2\sum_{i=1}^{I_p}\sum_{j=1}^{n_i}\left[\ln\frac{\tilde{S}_{i,j}}{\mu_i} - \frac{\tilde{S}_{i,j} - \mu_i}{\mu_i}\right] \tag{3.6}$$

Note that in Eq. 3.6 the shape parameters α phases out.
Therefore, given the estimated set of parameters $\hat{\boldsymbol{\beta}}$ we can compute the expected claim severity given the information set $\boldsymbol{x}_i$:

$$E^{\text{glm}}(\tilde{S}_{i,j}/\boldsymbol{x}_i) = \exp(\boldsymbol{x}_i'\hat{\boldsymbol{\beta}}), \qquad \text{for } i = 1, 2, \ldots, I_p, \qquad \text{and} \qquad j = 1, \ldots, n_i \tag{3.7}$$

where the superscript 'glm' signals that the estimate is produced by the GLM.

3.2.2 Gamma Neural Network (Gamma-NN)

The Gamma-NN is a feed-forward neural network with a one dimensional output layer. Considering the typical set of covariates $\boldsymbol{x}_i$ and a network on dept K, keeping in mind the notation of Section 1.4, we can define the output layer of the network as:

$$z^G(\boldsymbol{x}_i)(\boldsymbol{\theta}) = \psi(\theta_0^{K+1} + \textstyle\sum_{l=1}^{q_K} \theta_l^{K+1} z_l^{(K:1)}(\boldsymbol{x}_i)) = \psi(a^{(F)}(\boldsymbol{x}_i)), \qquad \text{for } i = 1, 2, \ldots, I_p, \tag{3.8}$$

where $z^G(\boldsymbol{x}_i)(\boldsymbol{\theta})$ is the one-dimensional output produced by network for the vector of covariates $\boldsymbol{x}_i$. Unlike the r-dimensional output Network of Section 2.2, here θ_0^{K+1} is a scalar representing the output bias and θ_l^{K+1} is the weight connecting the l-th neuron in the last hidden layer to the output layer. The $z_l^{(K:1)}(\boldsymbol{x}_i)$ term in Eq. 3.8

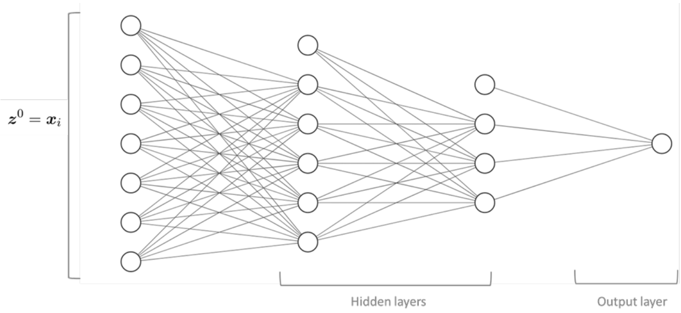


Figure 3.3. Gamma Neural Network architecture for with $K = 2$, $q_0 = 7, q_1 = 5, q_2 = 3$ and $\boldsymbol{z}^0 = \boldsymbol{x}_i$

can be expressed as in Eq. 1.14. The output activation ψ is taken to be exponential, and $\boldsymbol{\theta}$ denotes the entire set of network weights. The network architecture is shown in Figure 3.3.

The expected claim severity is given by the output produced by the network:

$$E(\tilde{S}_{i,j}/\boldsymbol{x}_i) = z^G(\boldsymbol{x}_i)(\boldsymbol{\theta}) \tag{3.9}$$

In order to estimate the set of parameters $\boldsymbol{\theta}$ we have to define an appropriate loss function for the network model. Given that we are in a Gamma setting, we train the model on Gamma deviance, tweaking Eq. 3.6 we have:

$$D^{\mathrm{G}}(\boldsymbol{S}, \boldsymbol{\theta}) = -2 \sum_{i=1}^{I_p} \sum_{j=1}^{n_i} \left[\ln \frac{\tilde{S}_{i,j}}{\boldsymbol{z}^{\mathrm{G}}(\boldsymbol{x}_i)(\boldsymbol{\theta})} - \frac{\tilde{S}_{i,j} - \boldsymbol{z}^{\mathrm{G}}(\boldsymbol{x}_i)(\boldsymbol{\theta})}{\boldsymbol{z}^{\mathrm{G}}(\boldsymbol{x}_i)(\boldsymbol{\theta})} \right], \tag{3.10}$$

the objective when training the network is:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \qquad D^{\mathrm{G}}(\boldsymbol{S}, \boldsymbol{\theta}) \tag{3.11}$$

In particular, we employ Stochastic Gradient Descent in order to obtain the estimate for the set of parameters $\hat{\boldsymbol{\theta}}$. We can compute the expected claim severity for the i-th policyholder as:

$$E^{\mathrm{nn}}(\tilde{S}_{i,j}/\boldsymbol{x}_i) = z^G(\boldsymbol{x}_i)(\hat{\boldsymbol{\theta}}) \tag{3.12}$$

where the superscript 'nn' shows that the expectation is obtained via a neural network.
In this particular case we estimate the expected claim severity for the three claim type, resulting in: $E^{\mathrm{nn}}(\tilde{S}^1_{i,j}/\boldsymbol{x}_i)$, $E^{\mathrm{nn}}(\tilde{S}^2_{i,j}/\boldsymbol{x}_i)$, and $E^{\mathrm{nn}}(\tilde{S}^3_{i,j}/\boldsymbol{x}_i)$.

3.3 Results and discussion

This section assesses the possible merits of the Gamma Neural Network w.r.t. the GLM for claim severity estimation by testing this approach over the dataset presented in Section 3.1. As briefly mentioned above, we fit a different Gamma GLM and a Gamma-NN for each claim type (*Visits*, *Dentalcare*, and *Diagnostic*). In the

GLM, continuous variables are treated via cubic splines, while categorical variables are One-Hot encoded. As concerns Gamma Neural Networks, the models are trained over $1,000$ epochs using early stopping on the validation set in order to prevent overfitting. Each network model adopts a standard three hidden layer structure of dimension $(30, 20, 10)$ ([53]) with a ReLu activation function. Data is preprocessed as follows: **AG**,**PE**, **DM** are Min-Max scaled, **RE** and **FA** are treated using a $d = 1$ embedding layer, and **GE** is dummy encoded. The results discussed below stem from a five-fold cross-validation.

3.3.1 Performance

We evaluate the performance of the different models using the Gamma Deviance. In particular, to assess the estimates produced by each model, we compute the Gamma Deviance (Eq. 3.6 for Gamma GLM and Eq. 3.10 for Gamma-NN), where the lower the deviance, the better the model. Given the set of features $\boldsymbol{x}$, each model returns an estimate for the expected claim severity $E(\tilde{S}_{i,j}/\boldsymbol{x}_i)$ of a medical visit, dental treatment, or diagnostic test. Figure 3.4 compares the in-sample (left panes) and the out-of-sample (right panes) performance for the Gamma-GLM and the Gamma-NN. In each plot, we report the Gamma Deviance over the five data folds to evaluate our results' stability. The outcomes seem relatively robust for each claim severity model across the five-folds, both in-sample and out-of-sample. In fact, neural network models outperform GLMs consistently since they always return a lower deviance.

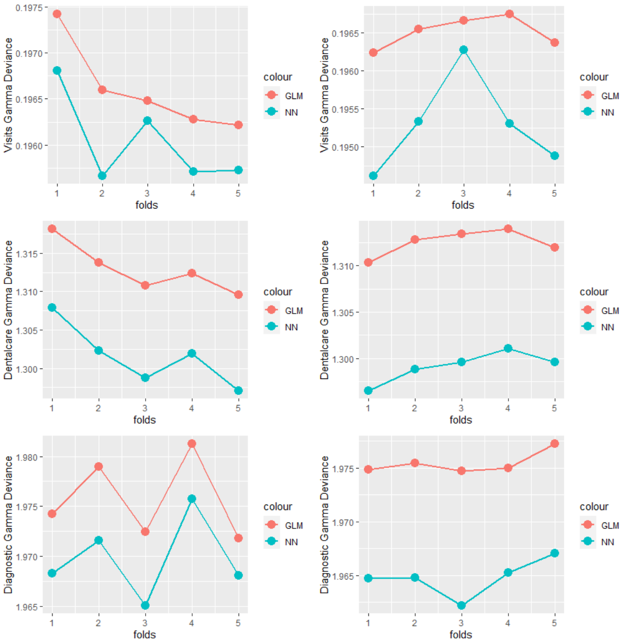


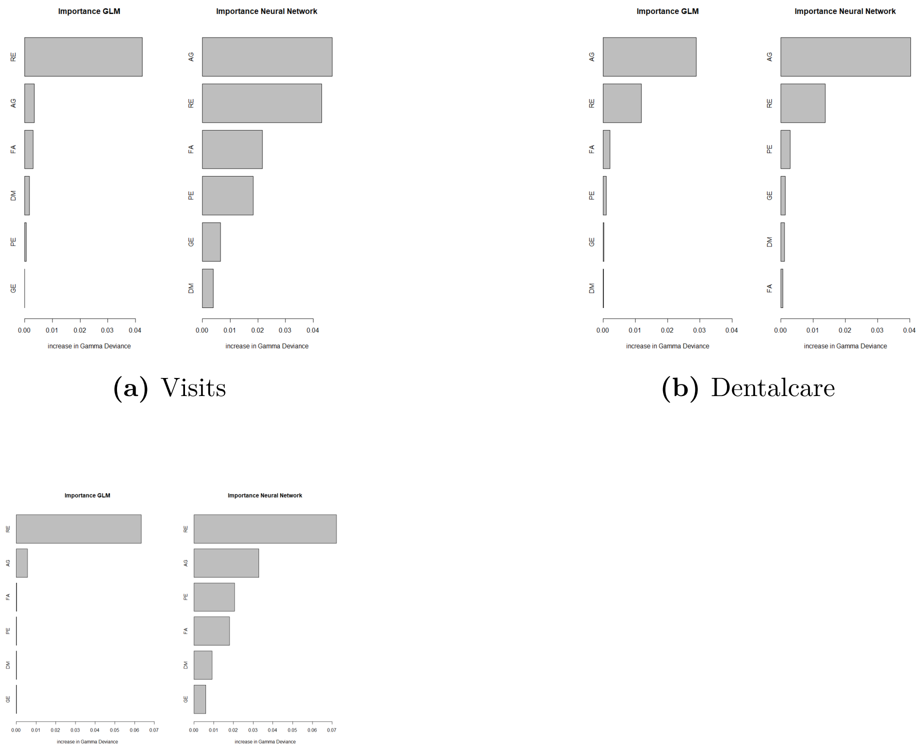
Figure 3.4. Performance for both GLM and NN, in-sample (left pane) and out-of-sample (right pane). On the y-axis we report the Gamma deviance, while the x-axis stores the reference data fold.

In the remainder of this section, we motivate this performance gap by analyzing the data representation learned by the different models using the set of model agnostic tools presented in Section 2.3: Variable Importance, PD plots, and interaction strength.

3.3.2 Importance

To learn which variables are more relevant in claim severity prediction, in Figure 3.5, we compare the variable importance for the different regression techniques. The covariates are ranked from top to bottom, starting with the most important one. The left plots report the variable importance for GLMs, while the right plots display the variable importance for neural networks. Claim severity models for *Visits* (Figure 3.5a) show some radical differences when it comes to variable importance. The only variable that seems to be relevant for the GLM is the region (**RE**), while the NN has several important covariates, age (**AG**) and region (**RE**) above all. In particular, the region's importance is comparable between the two models, while the age is where the real difference comes up since the variable is the most important in the NN, whilst it is almost irrelevant for the GLM. Other differences between GLM and NN are given by the **FA** and **PE** variables, which are relevant only in the latter model. In contrast, *Dentalcare* models (Figure 3.5b) show a similar variable importance plot. Both models strongly rely on the age variable for their predictions, with minor but still relevant importance for the region. Even though the plots are roughly the same, we notice a slightly higher importance for variables in the NN model. In a similar fashion to *Visits*, the claim severity models for *Diagnostic* display two different variable importance plots. Indeed, except for region (**RE**), the variables entering the GLM are deemed irrelevant. While the neural network extracts important information also from the age (**AG**), the permanence (**PE**) and the family member type (**FA**).
The results reported below suggest that probably GLM models are unable to exploit the entire informative set provided by the covariates. Nevertheless, even if some of the variables characterizing GLMs display a low level of importance, their parameters appear to be often significant, as shown in Tables B.2, B.3, and B.4.

In the following, we further investigate our models looking at main effects.



(a) Visits

(b) Dentalcare

(c) Diagnostic

Figure 3.5. Variable Importance for GLM (left) and neural network (right). The results reported are obtained on the last testing fold of the five-fold crossvalidation. Each plot reports the results for a different claim type.

3.3.3 Individual Conditional Expectation and Partial dependence

In order to study the covariates' main effects, in this subsection, we present the partial dependence plots for the claim severity models (Gamma GLM and Gamma NN) and the different claim types.

Visits: Comparing the PD plots for the GLM and the neural network in Figure 3.6, we notice some significant differences for the age (**AG**) and permanence (**PE**) variables, followed by a minor difference for the family member's main effect. Such divergences reflect the differences reported in the importance plots in Figure 3.5a. The age (**AG**) displays by far the wider distance between the curves. Its main effect in the neural network follows an almost concave up parabolic trend that associates higher claim severities to younger ages, probably connected to pediatric visits. In contrast, the GLM age main effect is mostly flat. The right-bottom pane shows the region's (**RE**) main effect for the two models. Such effect is roughly the same for the two models, and it displays a relevant variability w.r.t. the region in which the claim occurs. In this context, 'Lombardia' appears as the priciest region when it comes to medical visits. We also notice some slight differences for the permanence (**PE**) and the family member's (**FA**) main effects. While **GE** and **DM** show roughly the same PD plot.

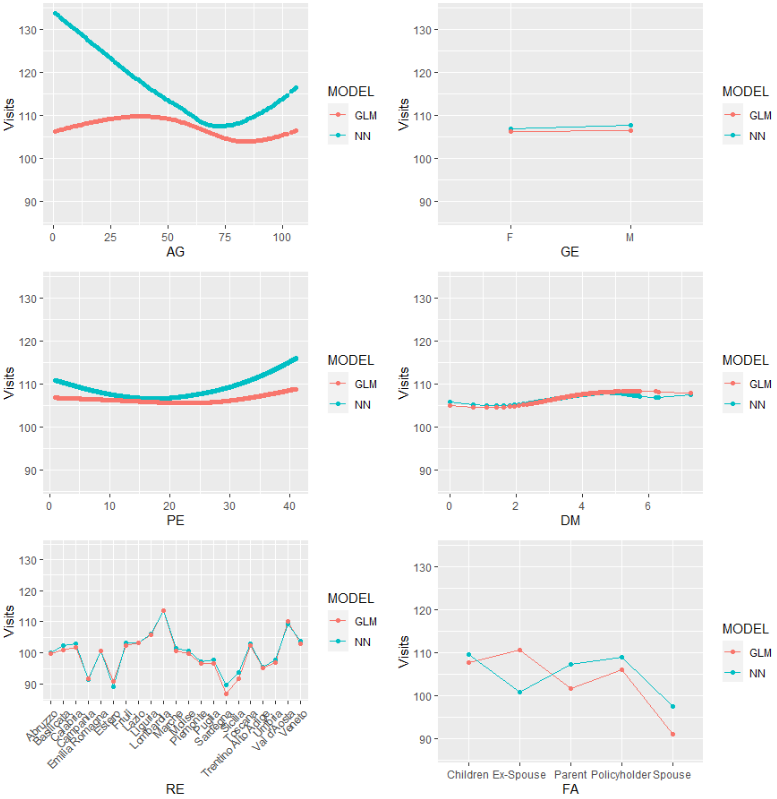


Figure 3.6. Partial dependence plots for *Visits* - NN

Dentalcare: Consistently with the importance plot in Figure 3.5b, except for the age (**AG**), all the PD plots in Figure 3.7 for the two models look almost alike. Moreover, the main effects for gender (**GE**), permanence (**PE**), dimension (**DM**), and family member (**FA**) are almost horizontal. In contrast, the region effect shows different claim severities for the Italian regions. 'Lombardia' still presents a considerable unitary claim cost. However 'Trentino Alto Adige' is clearly the region with the highest cost associated with dental treatments.
As mentioned above, the only variable with a substantial divergence in the PD plot is the age (top-left). Such difference is connected to younger ages, where the GLM shows a decreasing unitary claim cost between 0 and 35 years of age; while the NN main effect starts at a really low value and it starts increasing reaching a peak at 15 years, capturing the high cost associated to dental braces that characterize teenagers. In this case, the neural network seems to fit this specific effect better.

Diagnostic: Figure 3.8 reports the PD plots for diagnostic test claim severity models. Major distances in the PD plots are observed for the age (**AG**), permanence (**PE**), and a specific value of the family member variable (**FA**). At the same time, the other variables show approximately the same pattern for the main effect plot, almost flat for gender (**GE**) and dimension (**DM**), and with a fairly relevant effect for the region (**RE**). The great difference in the age main effect (top-left plot) is mainly connected to younger ages, where the GLM has an almost flat effect, and the neural

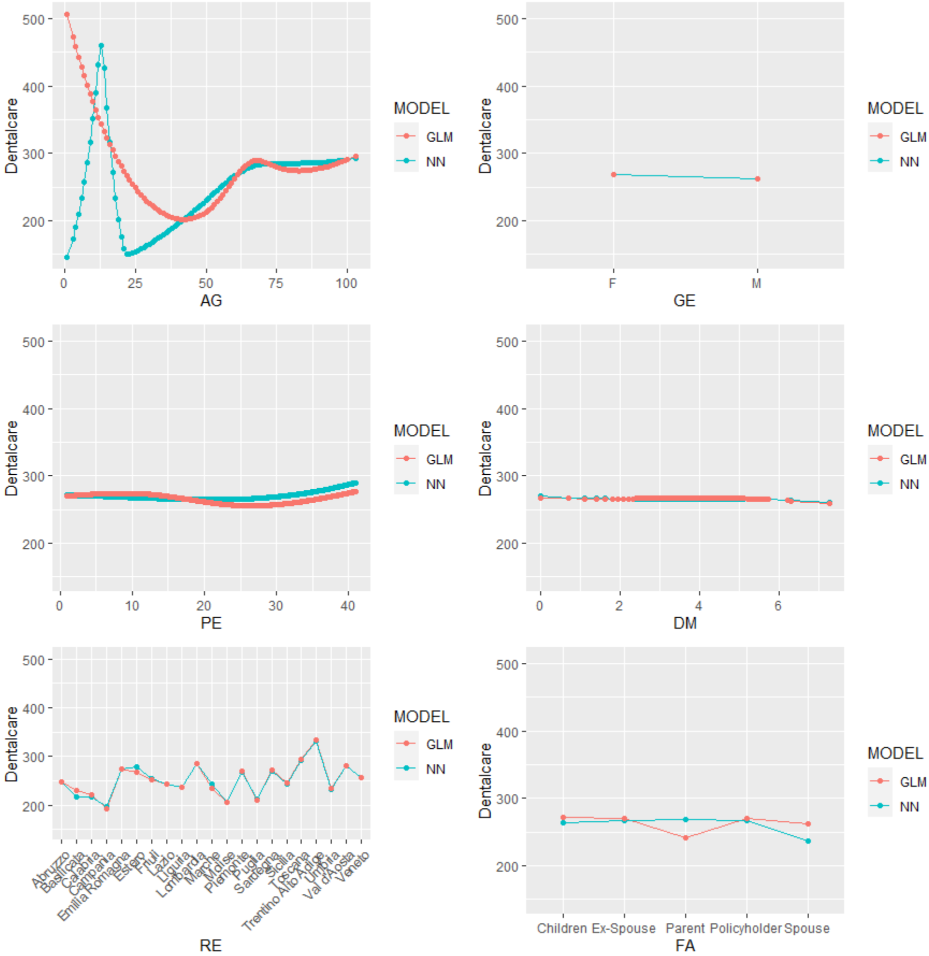


Figure 3.7. Partial dependence plots for *Dentalcare* - NN

network reports a higher, though decreasing, claim severity. The difference in the family member effect is related to the Ex-Spouse policyholder value. There is no real reason for that; hence it may be a potential signal for overfitting the learning data.

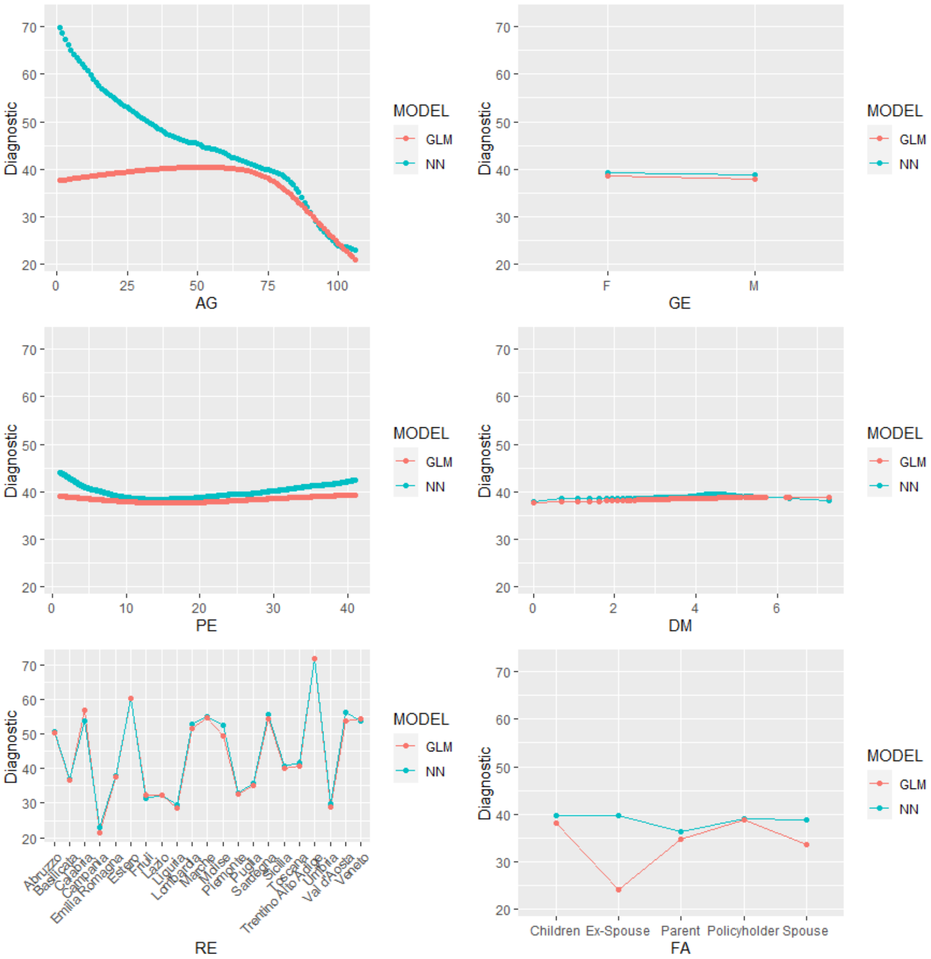


Figure 3.8. Partial dependence plots for *Diagnostic* - NN

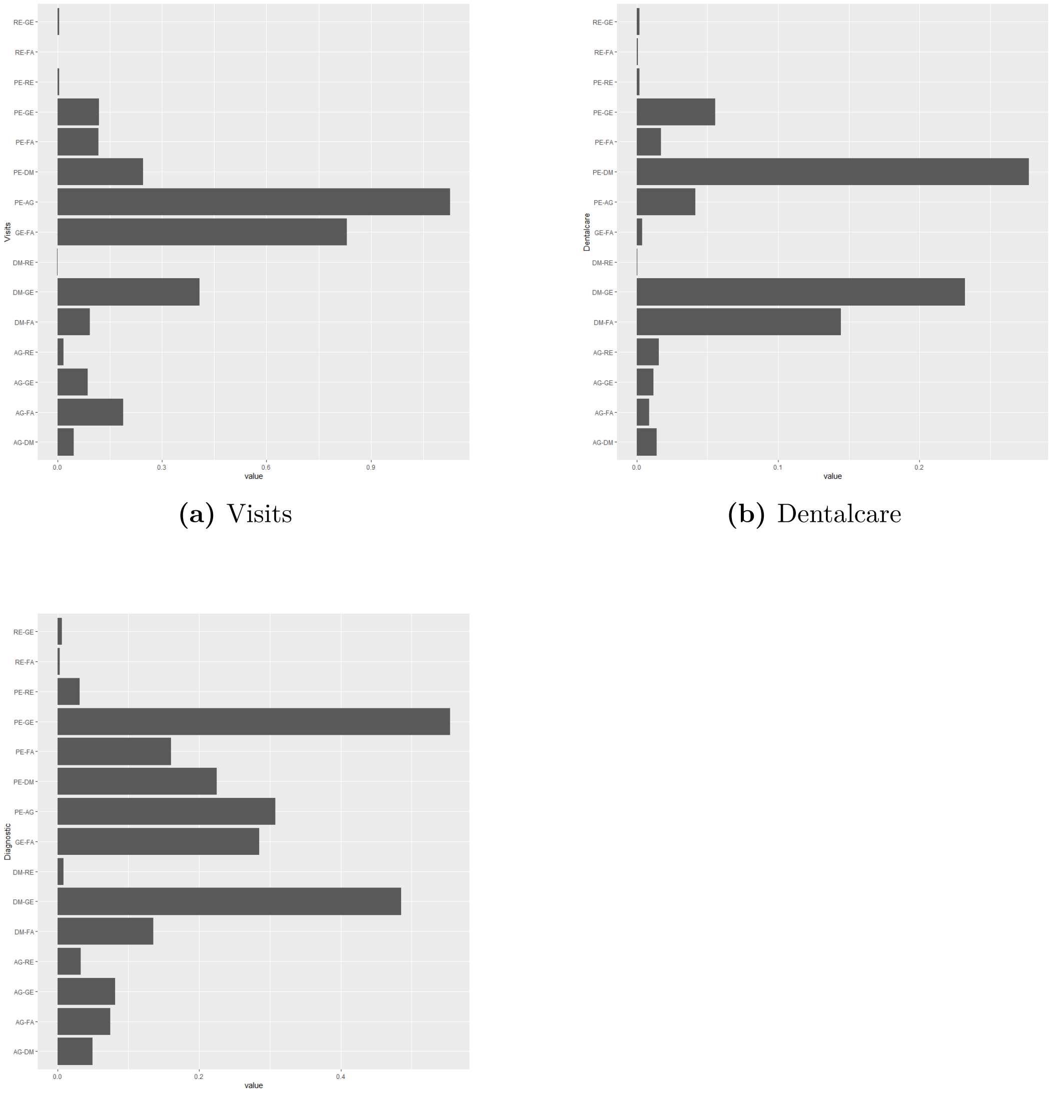
3.3.4 Studying interactions

This subsection looks for two-way interaction effects using the H-statistic [24] presented in Eq. 2.22. In Figure 3.9, we report the H-statistic for each possible pairwise interaction between variables entering the Gamma Neural Networks. Note that we compute the H-statistic only for neural network models since GLM are not designed to capture variable interactions.

Namely, interactions captured by the Gamma-NN for medical visits are presented in Figure 3.9a. The most relevant interaction is the one between permanence **PE** and age **AG**. Such interaction is quite peculiar since its H-static exceeds 1; this happens when the variance of joint interaction between the variables is greater than that of their 2-dimensional PD plot. This anomaly could also be due to the strong and positive correlation between the two covariates observed in Figure 2.2. Other relevant interactions are observed between gender and family member type, firms dimension and gender, permanence and dimension.

From Figure 3.9b, we notice that all the relevant interactions for dental treatment Gamma-NN model lean on the company dimension variable (**DM**). In particular, this variable interacts with permanence (**PE**), gender (**GE**), and family member type (**FA**). It is curious to observe such relevance for the dimensional variable since this covariate has a low importance (Figure 3.5b) and an almost flat PD plot (Figure 3.7). The grouped PD plots that will be discussed in the second part of this subsection will allow understanding if such interactions are really relevant.

Figure 3.9c reports variable interactions for the *Diagnostic* Gamma-NN model. The plot reports a plethora of relevant interactions, among the most relevant ones we have those between: the permanence and the gender, the dimension and the gender, the permanence and the age, the gender, and the family member.



(a) Visits

(b) Dentalcare

(c) Diagnostic

Figure 3.9. H-statistic - for the NN.

The H-statistic informs us on the strength of the interaction between two variables relative to their joint main effect. However, this statistic does not explain how the effect behaves on the prediction produced by the model. Grouped partial dependence plots, presented in Section 2.3, help shed light on the effect produced by the interaction. Below we present and discuss the grouped PD plots to investigate the interactions mentioned above.

Visits: In the top-left pane of Figure 3.10, we present the grouped PD plot for the age variable w.r.t. the family member type (for ease of discussion, we drop Ex-Spouse and Parents). The plots display a different slope at younger ages (below 50 years). Signaling that Children at younger ages present higher claim severities than policyholders. In contrast, the claim cost at older ages is roughly the same for each insured type. This grouped PD plot is a decomposition of the main effect represented in the top-left pane of Figure 3.6. The PD plot at younger ages only makes sense for Children since it is implausible to observe a Policyholder or a Spouse below 25 years of age (see top-left plot in Figure 2.2). While the opposite is true for the PD plots at higher ages, only the Policyholders or Spouses have such a level of seniority.

The interaction between **DM** and **GE** (in the top-right plot) presents an increased claim severity for females insured belonging to large-sized firms. However, this interaction only has a limited effect on the claim cost. The Grouped PD plots for the family member type conditional to the insured gender (bottom-right pane) are almost parallel for males and females insured, with a lower claim severity associated with females. However, when the policyholder is a female, the cost for an individual claim is roughly at the same level (if not higher) as a male. Such interaction may signal that when the insured is the policyholder (i.e., the owner of the insurance coverage), the effect produced by the gender is ruled out. For the interaction between age **AG** and permanence **PE** (bottom-right plot in Figure 3.10), that displayed an anomalous value in Figure 3.9, we notice a strong decreasing slope for younger insured (below 50 years of age) associated with a mid to high permanence (from 20 to 40 years)[2]. However, this does not make much sense since it is impossible to have a young insured with many years of permanence, i.e., a teenager with 20, 30, or 40 years of permanence. Therefore, the grouped plots for this interaction are only marginally informative. For instance, the PD plot conditional on a permanence of 40 years in the bottom-right pane of Figure 3.10 (in light blue) only makes sense for insured with more than 40 years of age. It is possible to conclude that the anomaly detected by this interaction does not seem to be highly informative, and the value of its H-statistic is probably due to the strong correlation between age and permanence.

[2] In order to have an easily readable plot we present Grouped PD plots conditional only on a representative set of permanence values (i.e. 1, 10, 20, 30, 40)

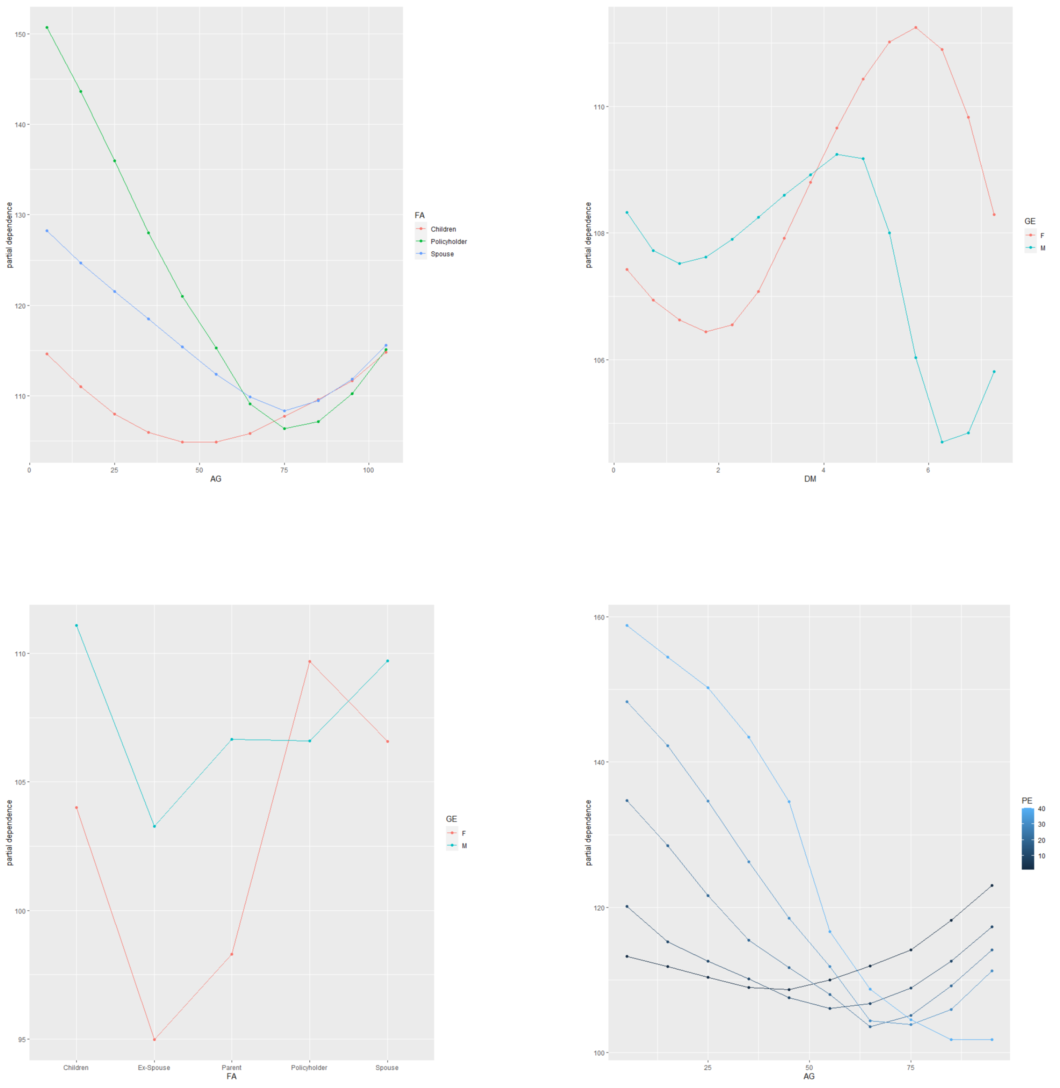


Figure 3.10. Grouped PD plots to study the interactions presented in Figure 3.9a.

Dentalcare: In Figure 3.11, we present the Grouped PD plots of the interactions captured by the Gamma-NN for dental treatment claims. The top-left pane displays the Grouped PD plots of the dimension (**DM**) w.r.t. to the family member type (for ease of discussion, we remove the Ex-Spouse from the plot). The interaction between **DM** and **FA** seems to spot an effect for Parents of policyholders working in big-sized firms since the behaviour of this PD profile (in green) is radically different from those of other types of insured (Policyholder, Spouse, and Children). In particular, for Parents, we notice an increasing claim severity when associated with more prominent firms, while for other family member types, we observe a slightly decreasing trend. The interaction between **DM** and **GE** (top-right pane) produces two utterly different PD plots. The dimension (**DM**) PD plot conditional on females is concave down, while the plot conditional on males is concave up. However, even though the two curves have radically different behaviour, their range of claim severity is relatively small (between 260 and 275 euros). Hence the relevance of this interaction is fairly limited.

The bottom-left pane displays the Grouped PD plot to study the interaction between dimension (**DM**) and (**PE**). Considering all possible values for permanence would result in a messy plot with over 40 PD profiles. Looking at the graph, it is clear that more prominent firms (with a high **DM** value) have an increasing effect on the cost of dental treatment claims for insured with high permanence. In contrast, they have a decreasing effect on the claim severity for insured with a mid to low permanence.

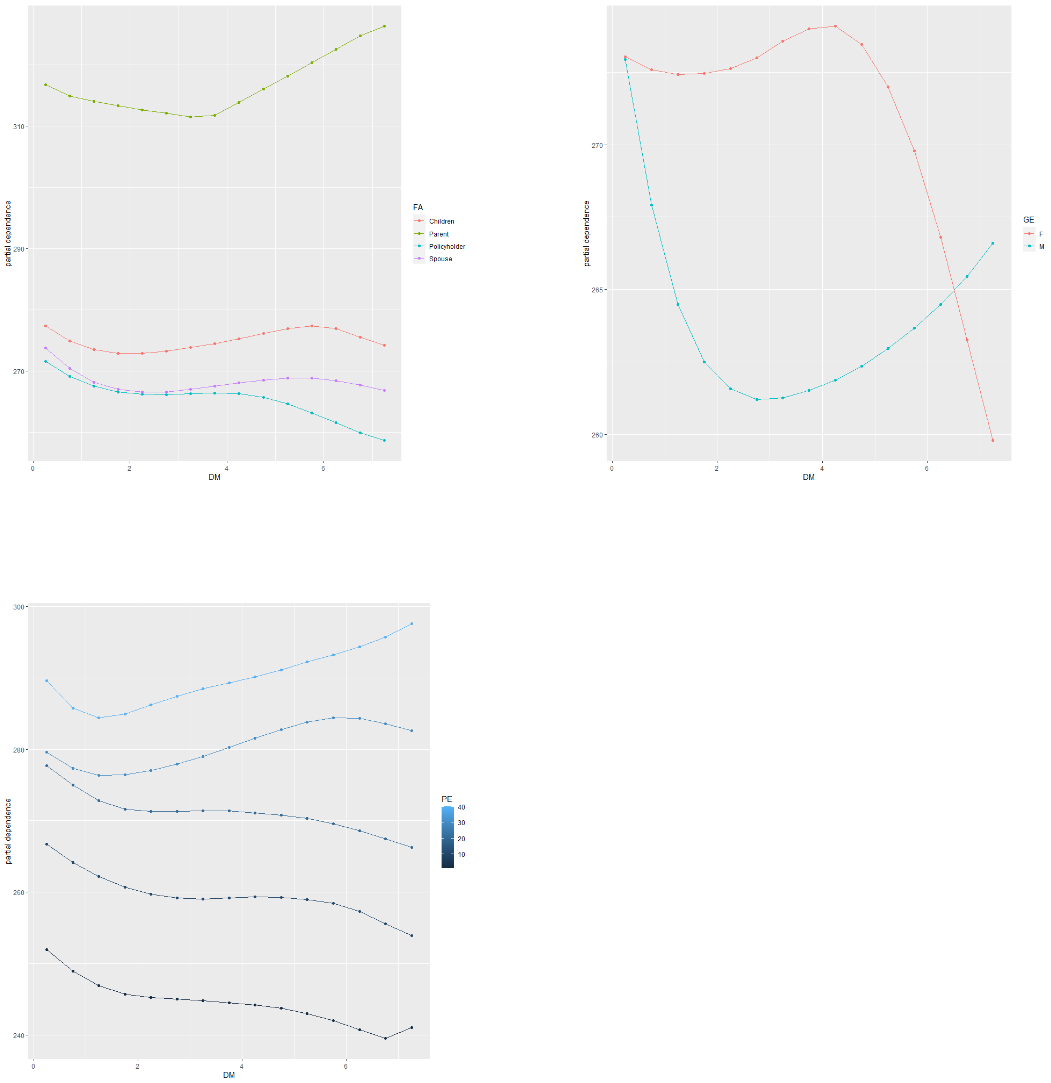


Figure 3.11. Grouped PD plots to study the interactions presented in Figure 3.9b.

Diagnostic: The interaction between permanence and gender, depicted in the top-left plot of Figure 3.12, seems to increase the cost of diagnostic claims filed by male insured with many years of permanence. Again, gender has another relevant interaction with the dimension (top-right plot in Figure 3.12). In particular, the

effect is more relevant in more prominent companies, where females tend to submit increasingly larger claims, while males present cheap *Diagnostic* claims. The bottom-right plot apparently spots a strong interaction between the type of family member and gender. In particular, the interaction signals huge claims for male ’Parents’. It is worth noting that ’Parents’ are usually older insured, as witnessed in Figure 2.2, who often need expensive diagnostic tests that result in bigger claim severities. However, the strong peak observed in the PD plot appears to be excessive since it is unlikely for male ’Parents’ to have a higher claim cost than female ’Parents’. This result is possibly due to overfitting produced by the neural network model on the training data. Namely, the overfitting may be caused by the dataset’s low number of ’Parents’ (see Figure 2.1)
We neglect the bottom-right Grouped PD plot between age and permanence since it has the same issues as the bottom-left plot reported in Figure 3.10 discussed above.

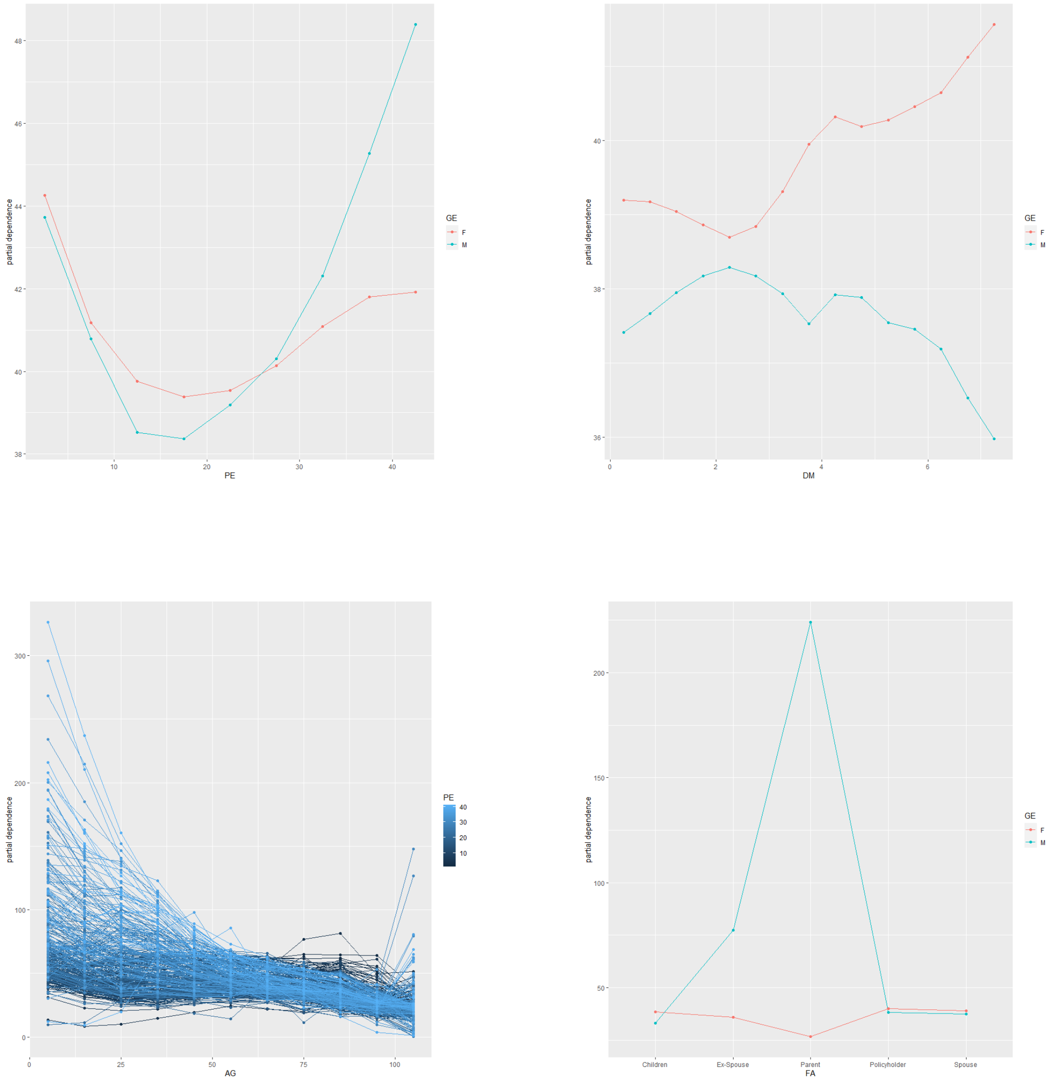


Figure 3.12. Grouped PD plots to study the interactions presented in Figure 3.9c.

Similarly to Section 2.3, we observe that the proposed neural network models outperform GLMs by better representing the main effects and incorporating possible interactions (even though, as argued above, some of those interactions have proven to be irrelevant). The comparison we have drawn up until now between neural networks and GLMs only focuses on the statistical performance of our models. However, this is not enough to determine whether it is worth choosing neural networks over GLMs. In fact, when determining the price (or the potential cost) of a risk coverage, it is also vital to consider business-related metrics. Therefore, in the next section, we combine the frequency models discussed in Chapter 2 and the claim severity model in a pricing model and compare the different tariff structures using practical economic metrics relevant for an insurance company.

3.4 Evalute Premiums - Model Lift

Now that we have extensively discussed claim frequency models (Section 2.2) and claim severity models (Section 3.2) we can combine them to complete the Frequency-Severity approach displayed in 1.2 devoted to pure premium evaluation. For instance, it is possible to compute the pure premium for a set of insureds according to their characteristics by combining the NM-NN and the Gamma-NN. In our specific case, in order to obtain the pure premium for the health insurance plan presented in this work, we must tweak Eq. 1.2 in order to account for the different claim types. In particular, the pure premium obtained via neural network models is defined as:

$$\hat{\pi}^{\text{nn}}(\boldsymbol{x}_i) = \sum_{k=1}^{3} E^{\text{nn}}(S_i^k/\boldsymbol{x}_i) = \sum_{k=1}^{3} E^{\text{nn}}(N_{k,i}/\boldsymbol{x}_i) \cdot E^{\text{nn}}(\tilde{S}_{i,j}^k/\boldsymbol{x}_i), \tag{3.13}$$

for $i = 1, \dots, I$, where $E^{\text{nn}}(N_{k,i}/\boldsymbol{x}_i)$ is defined as in Eq. 2.19 and $E^{\text{nn}}(\tilde{S}_{i,j}^k/\boldsymbol{x}_i)$ is obtained as in Eq. 3.12.
While, using the GLM models we have:

$$\hat{\pi}^{\text{glm}}(\boldsymbol{x}_i) = \sum_{k=1}^{3} E^{\text{glm}}(S_i^k/\boldsymbol{x}_i) = \sum_{k=1}^{3} E^{\text{glm}}(N_{k,i}/\boldsymbol{x}_i) \cdot E^{\text{glm}}(\tilde{S}_{i,j}^k/\boldsymbol{x}_i), \tag{3.14}$$

for $i = 1, \dots, I$, where $E^{\text{glm}}(N_{k,i}/\boldsymbol{x}_i)$ is defined as in Eq. 2.14 and $E^{\text{glm}}(\tilde{S}_{i,j}^k/\boldsymbol{x}_i)$ is obtained as in Eq. 3.7.
As it is common in actuarial pricing, the claim frequency and claim severity models discussed in this work are calibrated to optimize a goodness-of-fit measure. Thus, until now, when comparing neural networks to GLMs, we have mainly focused on the statistical performance of the models. Therefore, even if neural networks have outperformed GLMs from a statistical standpoint both in Section 2.3 and Section 3.3, it is crucial to understand if the premium produced by such models provides added value to the business in which the premium is to be implemented. Thus, it is also important to consider an economic criterion when deciding whether it is worth implementing a given model in insurance applications. Therefore it is crucial to go beyond the classical deviance metric. That is where model lift metrics come in handy. In brief, the model's lift refers to the pricing model's ability to prevent adverse selection. Precisely, the lift quantifies the model's ability to charge each insured an actuarially fair rate, thereby minimizing the potential risk of losing insured attracted by competitors using finer price lists.
In this section, we discuss two model lift methods proposed by [15] in order to evaluate the performance of a set of candidate premiums. The metrics proposed by the authors aim at assessing the two following aspects of a given premium: the variability of the resulting premium amounts, as larger premium differentiation induces greater lift, and the ability of the premium amount to match the actual total claim amount S for increasing risk profiles. The first objective is tackled using Lorenz curves that evoke the concept of convex orders, which are often used in applied probability to compare the variability of probability distributions beyond variance. The second point is assessed considering concentration curves. Given an insurance portfolio, if we consider the subset of insured gathering a certain percentage of policies associated

with smaller premiums (i.e., those insured that are likely to be lost to a potential competitor because of their low-risk profile), the concentration curve compares the premium amounts belonging to such group to their aggregate losses. The respective positions of the graphs of the Lorenz and concentration curves allow the actuary to assess the premium's performance under consideration accurately.
Before moving on, let us first define a generic working premium $\pi(\boldsymbol{x})$. This working premium $\pi(\boldsymbol{x})$ is an a approximation of the so-called true premium $\mu(\boldsymbol{x})$, which is the unknown regression function representing the insured riskiness w.r.t. to the information set $\boldsymbol{x}$. Then it is possible to assess the quality of a pricing model comparing $\pi(\boldsymbol{x})$ to $\mu(\boldsymbol{x})$ since it is crucial to evaluate the ability of a given model of predicting the true premium $\mu(\boldsymbol{x})$.
As mentioned above, the performance lift metrics proposed by the authors for the predictor are based upon concentration and Lorenz curves, whose definitions are recalled next.

Concentration curve - CC

In order to properly discuss the concentration curve it is crucial to define the cumulative distribution function for the working premium $\pi(\boldsymbol{x})$, as:

$$F_\pi(t) = P[\pi(\boldsymbol{x}) \leq 1], \qquad t \geq 0, \tag{3.15}$$

and F_π^{-1} is the associated quantile loss function.
Now, the concentration curve of the true premium $\mu(\boldsymbol{x})$ with respect to the working premium π conditional on the information set $\boldsymbol{x}$ is defined as:

$$\alpha \mapsto CC\left[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}); \alpha\right] = \frac{E\left[\mu(\boldsymbol{x}) \cdot \mathbb{I}\left[\pi(\boldsymbol{x}) \leq F_\pi^{-1}(\alpha)\right]\right]}{E\left[\mu(\boldsymbol{x})\right]} \tag{3.16}$$

the curve represents the proportion of the total true premium income corresponding to the sub-portfolio $\pi(\boldsymbol{x}) \leq F_\pi^{-1}(\alpha)$, i.e. to the $100\alpha\%$ of contracts with the smallest premium $\pi(\boldsymbol{x})$. This set of policies is characterized by a low risk profile, thus it is crucial to provide them with a proportionally low premium. Otherwise the insurance company is likely to lose such insured to a competitor offering lower premiums. In other terms, the concentration curve is a tool to assess the appropriateness of the premium $\pi(\boldsymbol{x})$ under consideration w.r.t. the insured risk profile.

Lorenz curve - LC

However, a concentration curve alone is not enough to assess performance of $\pi(\boldsymbol{x})$. For this reason, [15] propose to consider the Lorenz curve of the predictor, in addition to the concentration curve of the response with respect to the predictor.

$$\alpha \mapsto LC\left[\pi(\boldsymbol{x}); \alpha\right] = CC\left[\pi(\boldsymbol{x}), \pi(\boldsymbol{x}); \alpha\right] = \frac{E\left[\pi(\boldsymbol{x}) \cdot I\left[\pi(\boldsymbol{x}) \leq F_\pi^{-1}(\alpha)\right]\right]}{E\left[\pi(\boldsymbol{x})\right]} \tag{3.17}$$

The Lorenz curve represents the share of the total income produced by the working premium belonging to the sub-portfolio $\pi(\boldsymbol{x}) \leq F_\pi^{-1}(\alpha)$. Note that for Eq, 3.17 and 3.16 we have:

$$\alpha \mapsto CC\left[\pi(\boldsymbol{x}), \pi(\boldsymbol{x}); \alpha\right] = LC\left[\pi(\boldsymbol{x}); \alpha\right] \tag{3.18}$$

In other words, if $\mu(\boldsymbol{x}) = \pi(\boldsymbol{x})$, the two performance curves reduce to the Lorenz curve of $\mu(\boldsymbol{x})$. Meaning that the sub-portfolio corresponding to $\pi(\boldsymbol{x}) \leq F_\pi^{-1}(\alpha)$ is in equilibrium, as the premium matches the conditional expectation on average. Hence, a large difference between the CC and the LC suggests that the predictor $\pi(\boldsymbol{x})$ provides a poor approximation for the true premium.

Estimating curves and premium ranks

When it comes to curves estimation, it is important to note that the true premium $\mu(\boldsymbol{x})$ is not observed in reality. Which of course is a big problem when estimating $CC\left[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}); \alpha\right]$. However, as discussed in [15], it is possible to replace $\mu(\boldsymbol{x})$ with the total claim amount $\boldsymbol{S}$[3], thus we have:

$$CC\left[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}); \alpha\right] = CC\left[\boldsymbol{S}, \pi(\boldsymbol{x}); \alpha\right] = \frac{E\left[\boldsymbol{S} \cdot \mathbb{I}\left[\pi(\boldsymbol{x}) \leq F_\pi^{-1}(\alpha)\right]\right]}{E\left[\mu(\boldsymbol{x})\right]} \tag{3.19}$$

Therefore, in this case, the concentration curve describes the proportion of the total losses $\boldsymbol{S}$ belonging to the sub-portfolio gathering a given proportion α of policies with the lowest predictions. The concentration curve of the pure premium can be estimated as follows:

$$\widehat{\mathrm{CC}}\left[\boldsymbol{S}, \hat{\pi}(\boldsymbol{x}); \alpha\right] = \frac{\sum_{i|\hat{\pi}(\boldsymbol{x_i}) \leq \hat{F}_\pi^{-1}(\alpha)} S_i}{\sum_{i=1}^{I} S_i} \tag{3.20}$$

Where, $\hat{\pi}(\boldsymbol{x_i})$ is the estimated premium, produced by the frequency and severity models with their estimated parameters, and $\hat{F}_\pi^{-1}(\alpha)$ denotes the empirical distribution function of the estimated premium:

$$\hat{F}_\pi^{-1}(\alpha) = \frac{1}{I} \sum_{i=1}^{I} \mathbb{I}\left[\hat{\pi}(\boldsymbol{x}_i) \leq \alpha\right] \tag{3.21}$$

The empirical $\widehat{\mathrm{CC}}$ can be interpreted as the ratio of the total loss produced by those policies with estimated predictor $\hat{\pi}$ below its empirical quantile at level α and the total loss of the whole portfolio. It means that $\widehat{\mathrm{CC}}$ represents the sub-portfolio losses in relative terms, as a percentage of the aggregate loss at the entire portfolio level. The empirical version of the Lorenz curve is defined as:

$$\widehat{\mathrm{LC}}\left[\hat{\pi}(\boldsymbol{x}); \alpha\right] = \frac{\sum_{i|\hat{\pi}(\boldsymbol{x_i}) \leq \hat{F}_\pi^{-1}(\alpha)} \hat{\pi}(\boldsymbol{x_i})}{\sum_{i=1}^{I} \hat{\pi}(\boldsymbol{x_i})} \tag{3.22}$$

In other terms, $\widehat{\mathrm{LC}}$ is the percentage of the total premium income corresponding to the $100\alpha\%$ smaller premiums when the latter are computed using a predictor π.

The estimated premium $\hat{\pi}(\boldsymbol{x})$ allows to define a specific rank for the different insured. In particular, we notice that

$$\hat{\pi}(\boldsymbol{x}) \leq F_{\hat{\pi}}^{-1}(\alpha) \Leftrightarrow F_{\hat{\pi}}\left(\hat{\pi}(\boldsymbol{x})\right) \leq \alpha \tag{3.23}$$

[3] Where $\boldsymbol{S} = (S_1, \ldots, S_i, \ldots, S_I)$ is the vector of total claim amounts briefly mentioned in Table 2.1

it follows that we can represent the rank induced by the premium as:

$$\boldsymbol{\Pi} = F_{\hat{\pi}}\left(\hat{\pi}(\boldsymbol{x})\right) \tag{3.24}$$

In other terms, Π_i is the rank of insured i once all contracts have been ordered in ascending order according to their corresponding premiums.

Below we present the metrics proposed by [15] to assess the predictive performance of a pricing model. Such indicators, known as ICC (Integrated Concentration) and ABC (Area Between Curves), are based on the concentration and Lorenz curves discussed above. These metrics allow to measure the level of lift achieved by the pricing (or costing) model in consideration.

ICC - Integrated Concentration

To evaluate model lift a first method is to consider the area below the CC, i.e. the integral of the CC. The integrated curve is defined as:

$$\begin{aligned} ICC\left[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}), \alpha\right] &= \int_0^\alpha CC\left[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}), \varepsilon\right] d\varepsilon = \\ &= \frac{\operatorname{Cov}\left[\mu(\boldsymbol{x}), (\alpha - \Pi)_+\right]}{E(\boldsymbol{S})} + E\left[(\alpha - \Pi)_+\right] \end{aligned} \tag{3.25}$$

where $E\left[(\alpha - \Pi)_+\right]$ is a constant and Π is the ordering (Eq. 3.24) induced by the technical premium $\mu(\boldsymbol{x})$.
By ICC, we mean the integral of the concentration curve over the whole interval $[0, 1]$, i.e.

$$\begin{aligned} ICC = ICC\left[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}), 1\right] &= \frac{\operatorname{Cov}\left[\mu(\boldsymbol{x}), 1 - \boldsymbol{\Pi}\right]}{E(\boldsymbol{S})} + \frac{1}{2} = \\ &= \frac{1}{2} - \frac{\operatorname{Cov}\left(\mu(\boldsymbol{x}), \boldsymbol{\Pi}\right)}{E(\boldsymbol{S})} \end{aligned} \tag{3.26}$$

Let us now provide an intuitive interpretation for ICC. The ICC is based on the covariance between the real premium $\mu(\boldsymbol{x})$ and the rank $\boldsymbol{\Pi}$. The idea is that, the smaller (grater) $\boldsymbol{\Pi}$ the lower (larger) the true premium should be. Hence, a positive relationship between $\pi(\boldsymbol{x})$ and $\mu(\boldsymbol{x})$ translates into a positive covariance between $\mu(\boldsymbol{x})$ and $\boldsymbol{\Pi}$. Hence a more positive covariance term in the ICC, the better the corresponding candidate premium, resulting in a lower ICC value. In other terms, ICC measures the strenght of the association between the insured ordering Π given by our working premium and their true riskiness $\mu(\boldsymbol{x})$.
In order to compute the ICC we can replace $\mu(\boldsymbol{x})$ with $\boldsymbol{S}$ in Eq. 3.26.

ABC - Area Below Curve

Another interesting solution to assess model lift is to compare the Lorenz curve of the working premium to the concentration curve of the response with respect to the working premium. As stated in Eq. 3.17, if $\pi(\boldsymbol{x}) = \mu(\boldsymbol{x})$, $LC[\pi(\boldsymbol{x}); \alpha]$ and $CC[\mu(\boldsymbol{x}), \pi(\boldsymbol{x}); \alpha]$ coincide. Hence, considering both curves, and computing the area between them provides a good indicator of the performance of a given predictor.

Since the closer $\pi(\boldsymbol{x})$ to $\mu(\boldsymbol{x})$ the smaller the area between the CC and the LC. The area between the curves, ABC in short, is given by

$$\begin{aligned} \mathrm{ABC}\left[\pi(\boldsymbol{x})\right] &= \int_0^1 \left(\mathrm{CC}[\boldsymbol{S}, \pi(\boldsymbol{x}); \alpha] - \mathrm{LC}[\pi(\boldsymbol{x}); \alpha]\right) \mathrm{d}\alpha = \\ &= \frac{1}{\mathrm{E}[\pi(\boldsymbol{x})]} \left(\mathrm{Cov}[\pi(\boldsymbol{x}), \boldsymbol{\Pi}] - \mathrm{Cov}[\boldsymbol{S}, \boldsymbol{\Pi}]\right) = \\ &= \frac{1}{\mathrm{E}[\pi(\boldsymbol{x})]} \mathrm{Cov}[\pi(\boldsymbol{x}) - \boldsymbol{S}, \boldsymbol{\Pi}] \end{aligned} \tag{3.27}$$

If we define $\pi(\boldsymbol{x}) - \boldsymbol{S}$ as the profit associated with a given working premium, then ABC is proportional to the covariance between profits and the rank of premiums $\boldsymbol{\Pi}$. Clearly, a lower ABC signals a better model.

Results

We now employ the two lift metrics discussed above (ABC and ICC) to compare the of premiums obtained via GLMs $\pi^{\mathrm{glm}}(\boldsymbol{x})$ (Eq. 3.14) and neural networks $\pi^{\mathrm{nn}}(\boldsymbol{x})$ (Eq. 3.13).

In Table 3.4 we report the ABC and ICC for the two set of premiums. Both premiums are computed on the out-of-sample data, more specifically on the first fold of the 5-fold cross-validation. In particular, we notice that premiums issued from neural network models return both a lower ABC and ICC, signaling that neural network models produce a better lift if compared to GLMs. In other terms, the lower ABC registered by NN signals that the premium produced by this model is closer to the actual risk presented in the insurance portfolio, while the lower ICC means that such premiums cover the expected share of true premiums in the portfolio.

Model	ABC	ICC
GLM	0.00827	0.37760
NN	0.00562	0.37706

Table 3.4. ABC and ICC measure of lift for the two set of premiums

Thus, even from a business metric standpoint, neural network models have proven to have some added value if compared to GLMs, since their greater precision translates into better premiums.

In order to furtherly improve the discussed premiums, it would be necessary to complement the informative set presented in Section 2.1 including, for instance, additional covariates such as policyholders yearly income and level of education, that are generally good drivers for health expenditure.

In the next chapter, we revisit the quantile approach presented in [30] using neural network machinery in order to build a framework to gauge the potential riskiness for the insured enrolled in the health insurance coverage analyzed in this work.

Chapter 4

Modeling policyholders potential riskiness using a quantile approach

The Frequency-severity approach discussed in the previous chapters is purely devoted to estimating the expected value for the total claim amount $E(S_i)$ since it is designed to produce the pure premium for a health insurance coverage. Such a pure premium represents the average loss for a given policyholder. However, an insurance company (or an IHF) may also be interested in going beyond the pure premium, measuring, for instance, the potential loss associated with a given policyholder. A good metric to evaluate the potential riskiness of a policyholder is given by the quantile of the aggregate claim amount, which is equivalent to the Value-at-Risk defined on S_i. This risk measure is well established in the insurance industry, it works as a cornerstone for the definition of the Solvency II regulatory capital requirements, and it can be employed to compute a premium risk margin (or safety loading) to face possible adverse deviations from the expected value of S_i.

Therefore, this chapter discusses the estimation of conditional quantiles of the aggregate claim amount for health insurance, embedding the problem in a quantile regression framework using a neural network approach. More specifically, as the first step, we consider the Quantile Regression Neural Network (QRNN) procedure to compute conditional quantiles for the aggregate claim severity. We then propose a new model combining the traditional quantile regression approach with a Quantile Regression Neural Network that we call Quantile Regression Combined Actuarial Neural Network (Quantile-CANN). In both models, we adopt a two-part model scheme, see [30], where we fit a logistic regression to estimate the probability of positive claims and the QRNN model or the Quantile-CANN for the positive outcomes. We exploit the health insurance dataset presented in Section 2.1 to highlight the overall better performances of the proposed models with respect to the classical quantile regression one proposed in [30], [36] and [5]. We then use the estimated quantiles for S_i to calculate a loaded premium following the Quantile Premium Principle, showing that the proposed models provide a better risk differentiation w.r.t. the classical quantile regression.

The remainder of the chapter is structured as follows: Section 4.1 is devoted to data description, in particular, we deepen the discussion on the binary variable $\mathbb{I}_{N_i}$ and on the aggregate claim amount S_i, that we have briefly mentioned in Section 2.1; Section 4.2 explores the two-part model considered in this work; Section 4.3 introduces the use of QRNN and Quantile-CANN for the estimation of the quantile of the total claim amount; in Section 4.4 we debate the performance of our models also with the help of agnostic tools; Section 4.5 presents the actual quantile-based ratemaking for the health insurance coverage discussed in the previous chapters.

4.1 Data description

As previously mentioned, two-part Quantile models rely on two variables for their definition: a binary variable $\mathbb{I}_{N_i}$[1] signaling whether the insured files at least a claim during the year; and a positive continuous variable representing the aggregate claim severity $\tilde{S}_i$ which is defined as the sum of the monetary cost of all claims submitted by a given insured within the year. The dataset presented in Section 2.1 reports the claim binary variable $\mathbb{I}_{N_i}$ and the aggregate claim amount S_i representing the aggregate loss of a given insured during the year, this variable can either be null (when the insured submits no claim, i.e., $\mathbb{I}_{N_i} = 0$) or positive (when the insured submits at least a claim, i.e., $\mathbb{I}_{N_i} = 1$). From S_i it is possible to retrieve the aggregate claim severity $\tilde{S}_i$ by considering the positive part of the aggregate claim amount.
Table 4.1 displays a summary for the variables in the dataset involved in the quantile modeling. Figure 4.1 plots the histogram of the aggregate claim amount S_i for the 273,500 insured in our dataset. The histogram shows the semicontinuos [44] nature of this variable, with a consistent number of zeros due to insured submitting no claims, with the right-skewed positive part of the distribution. From the plot we observe the existence of some large claims, however the tail doesn't seem to be particularly fat. It is worth mentioning that S_i aggregates all the three claim types discussed in the previous chapters (*Visits*, *Dentalcare*, and *Diagnostic*) in one single variable. Therefore, we won't follow a multivariate approach for the response variable as in Section 2.2, but we will build our models on the aggregate loss encompassing the risk arising from all the different type of claims in the dataset. [2]

[1] Where N_i is the number of claims submitted by the policyholder. Here we consider all the claim types discussed in the previous chapter all together (*Visits*, *Dentalcare*, and *Diagnostic*). In other terms, using the notation introduced in Chapter 2, $N_i = N_{1,i} + N_{2,i} + N_{3,i}$.

[2] The adoption of a multivariate approach could be an interesting topic for further research.

Variable	Description
Binary dependent variable	
$\mathbb{I}_{N_i}$ (claims binary variable)	Binary variable reporting 1 if the insured filed at least a claim and 0 otherwise.
Positive dependent variable	
S_i (total claim amount)	Total claim amount submitted by the insured during the year (in euros). If the insured submits no claims $S_i = 0$. This variable takes into account the aggregate claim amount for the three claim types (*Visits*, *Dentalcare*, and *Diagnostic*).
Additional information	
Insured ID	Insured identifier used to join this dataset with the data set presented in Section 2.1

Table 4.1. Summary of the variables available in the dataset.

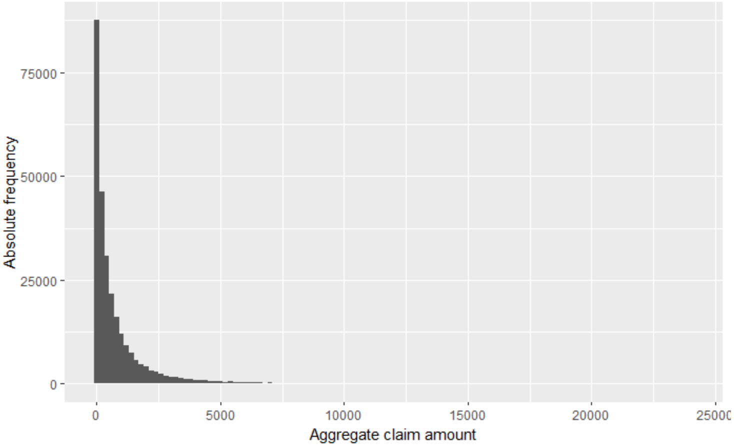


Figure 4.1. The plots report the histogram for the aggregate claim amount S_i.

To provide further insight into the data at hand in Table 4.2, we report the frequency table for the response variables characterizing the two-part model discussed in the next section. Observing the frequency table for the binary variable we observe 205,625 claimants (i.e. $\mathbb{I}_{N_i} = 1$) and 68,325 insured with no claims (i.e. $\mathbb{I}_{N_i} = 0$). Again, as mentioned in the previous chapters, the high proportion of claimants is peculiar to the health insurance context. For the aggregate claim amount S_i we observe that most claimants submit an yearly aggregate loss between 100 and 3,000 euros, while the top 5% insured has an aggregate claim amount grater than 3,000 euros.

Variable	Absolute frequency	Relative frequency	Variable	Absolute frequency	Relative frequency
S^i (*Visits* claim severity)			$\mathbb{I}_{N_i}$		
0	68325	0.249	0	68325	0.249
(0-100]	16675	0.060	1	205625	0.751
(100-500]	79311	0.289			
(500-1000]	44376	0.161			
(1000-3000]	48097	0.175			
(3000-5000]	9978	0.036			
(5000-∞]	7188	0.026			

Table 4.2. Frequency tables for the response variables

Note that the covariates characterizing the quantile models are those presented in Chapter 2. As we will discuss in the following, the goal of the quantile modeling approach is to study the cost associated to those insured that submit larger claim amounts to provide the insurance company (or the IHF) with a valuable risk management tool when evaluating the potential riskiness of a given insured.

4.2 The two part quantile model

In this section, we discuss the two-part quantile regression framework, first considered by [30], devoted to conditional quantile estimation of the aggregate claim amount S_i at level τ and for a specific insured i. As briefly mentioned above, two-part models involve a mixture distribution consisting in mixing a discrete point mass, with all mass at zero, and a continuous random variable. In particular, they are described by two equations: a binary choice model is fitted for the probability of observing a positive-versus-zero outcome. Then, conditional on a positive outcome, an appropriate regression model is fitted for the continuous outcome. The common structure of such models assumes that the effect of the covariates influence the mean of the conditional distribution of the response. In many real applications like the actuarial ones, however, the effect of the covariates can be different on different parts of such distribution. For this reason, a quantile regression approach for the continuous part of the model may be more appropriate when studying the potential riskiness of an insured. In particular, here, we focus our attention on the effect of covariates on the quantile of the aggregate claim amount S_i. In order to build the two-part quantile model (see [16], [22], and [42]) we consider the indicator random variable $\mathbb{I}_{N_i}$. If $N_i > 0$ then a positive aggregate claim severity $\tilde{S}_i$ is observed.
Consistently with this approach, given a set covariates $\boldsymbol{x}_i$, we model the τ-th conditional quantile of the total claim amount $Q_{S_i}(\tau|\boldsymbol{x}_i)$ in two stages:

- the first stage allows to estimate the no claim probability $p_i = Pr(\mathbb{I}_{N_i} = 0) = Pr(N_i = 0)$ as function of covariates . To achieve this goal we use the logistic regression:
$$log\left(\frac{1-p_i}{p_i}\right) = \boldsymbol{x}_i'\boldsymbol{\beta}, \tag{4.1}$$
where the no claim probability is obtained as $p_i = \frac{1}{1-exp(\boldsymbol{x}_i'\boldsymbol{\beta})}$;
- The second stage uses p_i to obtain the $\tau_i^\star$ conditional quantile level of $\tilde{S}_i$, $Q_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$, corresponding to the τ quantile level defined on the total claim

amount S_i, $Q_{S_i}(\tau|\boldsymbol{x}_i)$. Following [30] the $\tau_i^\star$ level can be calculated as:

$$\tau_i^\star = \frac{\tau - p_i}{1 - p_i} \tag{4.2}$$

for which

$$Q_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i) = Q_{S_i}(\tau|\boldsymbol{x}_i). \tag{4.3}$$

In the literature $Q_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$ is generally calculated using the well known quantile regression approach of [35]. In the next section we will generalized this approach by introducing two alternative methods, the Quantile Regression Neural Network (QRNN), a particular specification of neural networks introduced by [57], and the Quantile Combined Actuarial Neural Network (Quantile-CANN) which is a new method to calculate quantiles embedding the CANN approach of [53] in a quantile regression framework. These approaches allow performing quantile estimation without imposing any predetermined structure for the relations between the claim severity and the related covariates. In the following section we briefly introduce the classical quantile regression and then we explore the new techniques proposed in this work.

4.3 Quantile claim severity models

The standard approach to tackle the problem of quantile claim severity estimation refers to traditional QR models, see for example [36], [30] and [5]. In this paper, we generalize this approach by proposing neural network models to estimate the conditional quantile of the aggregate claim severity. This approach allows performing these calculations without imposing any predetermined structure for the relations between the aggregate claim severity and the related covariates. In this way, we can capture non-linear and complex patterns in the data and possible interactions between predictors.

The first model we introduce is Quantile Regression Neural Network (QRNN), which is basically a feed-forward neural network minimizing the same quantile loss function minimized by a QR model. The second one is a new model that generalizes the CANN approach introduced by [53] by combining the QR model with a QRNN to improve the model's performance. We call this model Quantile-CANN.

Since all the models mentioned above are based on Quantile Regression, we give a light insight into QR models in the following subsection.

4.3.1 Quantile regression

Quantile Regression, originally introduced by [35], is a distribution free method providing a way to model the conditional quantiles of a response variable with respect to a set of covariates in order to have a more robust and complete picture of the entire conditional distribution with respect to the classical mean regression. Quantile regression approach is quite suitable method used in all the situations where specific features, like skewness, fat-tails, outliers, truncation, censoring and heteroscedasticity arise. In this section we show how to calculate $Q_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$ using QR standard tools, in particular we analyze how to calculate the quantile of the

$\log(\tilde{S}_i)$. The use of the logarithmic function derives from the need to transform the dependent variable from $\mathbb{R}^+$ to $\mathbb{R}$ to apply the QR approach. Moreover, the use of the logarithmic transformation is coherent within the insurance pricing context, since it allows to consider multiplicative tariffs. In addition, by considering the equivariance under monotone transformation property of the quantile it is possible to retrive the quantity of interest, i.e. $Q_{\log(\tilde{S}_i)}(\tau_i^\star|\boldsymbol{x}_i) = \log(Q_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i))$. For notational simplicity hereafter we will use $\tau^\star$ instead of $\tau_i^\star$ in the formulas below.
The quantile regression model can be stated as follows:

$$\log(\tilde{S}_i) = \boldsymbol{x}_i'\boldsymbol{\beta}(\tau^\star) + \varepsilon_i, \qquad \text{for all } i = 1, 2, \dots, I_p, \tag{4.4}$$

where $\boldsymbol{\beta}(\tau^\star) = (\beta_1(\tau^\star), \beta_2(\tau^\star), \dots, \beta_{q_0}(\tau^\star)) \in \mathbb{R}^{q_0}$ is the vector of unknown regression parameters , I_p is the number of claimants, and ε_i has the $\tau^\star$-th conditional equal to zero for all $i = 1, 2, \dots, I_p$. The estimation of the regression parameters $\boldsymbol{\beta}(\tau^\star)$ can be obtained by solving the following minimization problem:

$$\hat{\boldsymbol{\beta}}(\tau^\star) = \underset{\boldsymbol{\beta}(\boldsymbol{\tau}^\star)}{\operatorname{argmin}} \frac{1}{I_p} \sum_{i=1}^{I_p} \rho_{\tau^\star}(\log(\tilde{S}_i) - \boldsymbol{x}_i'\boldsymbol{\beta}(\tau^\star)), \tag{4.5}$$

where $\rho_{\tau^\star}$ is the quantile loss function defined as:

$$\rho_{\tau^\star}(u) = u\,(\tau^\star - \mathbb{I}_{(u<0)}) \tag{4.6}$$

with $\mathbb{I}_{(u)}$ being the indicator function. The conditional quantile of $\tilde{S}_i$ is then estimated as $\hat{Q}_{\tilde{S}_i}(\tau^\star|\boldsymbol{x}_i) = \exp(\boldsymbol{x}_i'\hat{\boldsymbol{\beta}}(\tau^\star))$.
In what follows we explore the model proposed based on the neural network approach.

4.3.2 Quantile Regression Neural Networks

Quantile Regression Neural Network (QRNN) is a modeling technique introduced by [57] based on neural networks that enables to estimate the conditional probability distribution of multiperiod financial return within a quantile regression framework. With this approach, it is possible to estimate potential non-linear quantile relations without imposing any distributional assumption or functional relations between dependent and independent variables. Several applications of the methodology have already been implemented in different fields, see for instance [67] and [10]. Up to our knowledge, the QRNN methodology has never been considered in an insurance pricing context.
At a deeper insight QRNN model, for a fixed depth $K \in \mathbb{N}$, and fixed $\tau^\star$, follows a typical feed-forward structure:

$$Q_{\log(\tilde{S}_i)}(\tau^\star) = \exp\left\langle \boldsymbol{\theta}^{(K+1)}, \left(\boldsymbol{z}^{(K)}(\boldsymbol{\theta}^{(K)}) \circ \cdots \circ \boldsymbol{z}^{(s)}(\boldsymbol{\theta}^{(s)}) \circ \cdots \circ \boldsymbol{z}^{(1)}(\boldsymbol{\theta}^{(1)})\right)(\boldsymbol{x}_i)\right\rangle, \tag{4.7}$$

for $i = 1, 2, \dots, I_p$. Where the output of the network is given by the exponential activation function applied to the scalar product between the readout parameter vector $\boldsymbol{\theta}^{(K+1)}$ returning one neuron in the output layer and the composition of the different K hidden layers $\boldsymbol{z}^{(1)}(\boldsymbol{\theta}^{(1)}), \cdots, \boldsymbol{z}^{(K)}(\boldsymbol{\theta}^{(K)})$, where $\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(K)}$ are the

parameters belonging to each layer.
The generic s-th hidden layer $\boldsymbol{s}^{(s)}(\boldsymbol{\theta}^{(s)})$ of dimension $q_s \in \mathbb{N}$ is defined as:

$$\boldsymbol{z}^{(s)}(\boldsymbol{\theta}^{(s)}) : \mathbb{R}^{q_{s-1}} \to \mathbb{R}^{q_s}, \qquad \boldsymbol{z}^{(s)}(\boldsymbol{\theta}^{(s)}) = \left(z_1^{(s)}(\boldsymbol{\theta}_1^{(s)}), \cdots, z_j^{(s)}(\boldsymbol{\theta}_j^{(s)}), \cdots, z_{q_s}^{(s)}(\boldsymbol{\theta}_{q_s}^{(s)})\right)', \tag{4.8}$$

where the j-th neuron in the s-th hidden layer is given by:

$$z_j^{(s)}(\boldsymbol{\theta}_j^{(s)}) = \phi\left(\theta_{j,0}^{(s)} + \sum_{l=1}^{q_{s-1}} \theta_{j,l}^{(s)} \cdot z_l^{(s-1)}(\boldsymbol{\theta}_l^{(s-1)})\right) = \phi\left\langle \boldsymbol{\theta}_j^{(s)}, \boldsymbol{z}^{(s-1)}(\boldsymbol{\theta}^{(s-1)})\right\rangle, \tag{4.9}$$

where ϕ is the activation function and $\boldsymbol{\theta}_j^{(s)} = (\theta_{j,0}^{(s)}, \theta_{j,1}^{(s)}, \dots, \theta_{j,q_{s-1}}^{(s)})'$ is the vector of parameters belonging to j-th neuron in the s-th hidden layer. Considering the vector of parameters $\boldsymbol{\theta}_1^{(s)}, \dots, \boldsymbol{\theta}_{q_s}^{(s)}$ for each neuron in (4.8), we can define the matrix of parameters for the s-th hidden layer as $\boldsymbol{\theta}^{(s)} = (\boldsymbol{\theta}_1^{(s)}, \dots, \boldsymbol{\theta}_j^{(s)}, \dots, \boldsymbol{\theta}_{q_s}^{(s)})'$ of dimension $q_s \times (1 + q_{s-1})$.
Since the network in (4.7) has K hidden layers, it is possible to denote with $\boldsymbol{\theta}$ the full set of parameters for the network gathering the matrix of parameters of each layer:

$$\boldsymbol{\theta} = \left\{\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(s)}, \cdots, \boldsymbol{\theta}^{(K)}, \boldsymbol{\theta}^{(K+1)}\right\} \tag{4.10}$$

with dimension r, where $r = \sum_{s=1}^{K+1} q_s(1 + q_{s-1})$.
To obtain the optimal set of parameters $\hat{\boldsymbol{\theta}}$ for (4.10) we train the network (4.7) minimizing the quantile loss function:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{I_p} \sum_{i=1}^{I_p} \rho_{\tau^\star}(\log(\tilde{S}_i) - Q_{\log(\tilde{S}_i)}(\tau^\star)) \tag{4.11}$$

where we fix the starting value of the network parameter $\boldsymbol{\theta}_0$ at the beginning of the training.
From (4.7) and (4.11) we estimate $\hat{Q}_{\log(\tilde{S}_i)}(\tau^\star)$, then given the equivariance to monotone transformation of the quantile function we retrive $\hat{Q}_{\tilde{S}_i}(\tau^\star) = \exp(\hat{Q}_{\log(\tilde{S}_i)}(\tau^\star))$.
It is worth noting that if we replace the exp and the ϕ activation functions in (4.7) and in (4.9) with the linear activation function and we consider no hidden layers, then the QRNN boils down to the classical QR model in (4.5)

4.3.3 Quantile-CANN

The innovative model we propose in this section to estimate the quantiles for the distribution of interest is an extension of the Combined Actuarial Neural Network (CANN) approach proposed by [53] and launched in the editorial of [64]. In particular, the CANN framework nests a generic regression model into the neural network architecture to enhance the estimates given by the generic regression model. Following the CANN approach, but in a quantile framework, we boost the QR model with the neural network features proposing the so-called Quantile-CANN model. Our approach allows for exploiting the quantile neural networks to improve the conditional quantile estimates given by a QR model. In such a way, the neural network directly improves the classical QR estimates, preserving the information contained therein.

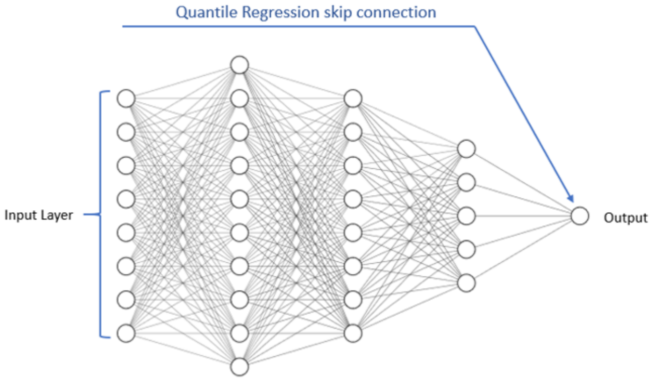


Figure 4.2. Quantile-CANN architecture

The main advantage of the Quantile-CANN approach compared to the QRNN is that it combines the flexibility of the networks with the interpretability of a QR approach, thus providing higher accountability. According to [64], when the reference regression model is already close to optimal, its maximum likelihood estimator can be used as initialization of the neural network fitting algorithm. We use the QR estimator to initialize the network parameter of the Quantile-CANN, then obtaining lower computational time for the network parameters calibration than the QRNN model.

Formally, we define the Quantile-CANN as:

$$Q^{CANN}_{\log(\tilde{S}_i)}(\tau^\star) = \langle \boldsymbol{\beta}(\tau^\star), \boldsymbol{x}_i \rangle + \left\langle \boldsymbol{\theta}^{(K+1)}, \left(\boldsymbol{z}^{(K)}(\boldsymbol{\theta}^{(K)}) \circ \cdots \circ \boldsymbol{z}^{(1)}(\boldsymbol{\theta}^{(1)}) \right) (\boldsymbol{x}_i) \right\rangle, \qquad \text{for } i = 1, 2, \ldots, I_p, \tag{4.12}$$

where the first term of the right hand side of (4.12) refers to the QR model in (4.4) with vector of parameters $\boldsymbol{\beta}(\tau^\star)$, while the second term is the QRNN model displayed in (4.7) (except for the missing exponential activation in the output layer). Therefore, the Quantile-CANN model, combines the models discussed in the two previous sub-sections (4.3.1 and 4.3.2) by embedding the QR into the network architecture using a skip connection that links the input given by the QR estimates to the output layer (see Figure 4.2 for a graphical representation), where the models are merged by summing the two parts as in (4.12). The network parameter of the Quantile-CANN model is denoted by $\boldsymbol{\vartheta}$ and consists of:

$$\boldsymbol{\vartheta} = \{\boldsymbol{\beta}(\tau^\star), \boldsymbol{\theta}\} = \left\{\boldsymbol{\beta}(\tau^\star), \boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(K)}, \boldsymbol{\theta}^{(K+1)}\right\} \tag{4.13}$$

The optimal set for the network parameter $\hat{\boldsymbol{\vartheta}}$ of (4.13) is obtained training the Quantile-CANN (4.12) minimizing the quantile loss function:

$$\underset{\boldsymbol{\vartheta}}{\operatorname{argmin}} \frac{1}{I_p} \sum_{i=1}^{I_p} \rho_{\tau^\star}(\log(\tilde{S}_i) - Q^{CANN}_{\log(\tilde{S}_i)}(\tau^\star)). \tag{4.14}$$

The optimization process to estimate $\boldsymbol{\vartheta}$ in (4.13) works as follows: we first obtain $\hat{\boldsymbol{\beta}}(\tau^\star)$ parameters minimizing the quantile loss function for the quantile in (4.5). Then we use such parameters to initialize the network parameter of the Quantile-CANN

by considering $\boldsymbol{\vartheta}_0 = \left\{\hat{\boldsymbol{\beta}}(\tau^\star), \boldsymbol{\theta}_0\right\}$, where $\boldsymbol{\theta}_0$ is the starting value of the network parameter belonging to the QRNN part of model (4.12). Therefore, starting from $\boldsymbol{\vartheta}_0$, we optimize $\boldsymbol{\vartheta}$ minimizing (4.14) by means of the gradient descent algorithm. During the optimization process both $\boldsymbol{\theta}_0$ and $\hat{\boldsymbol{\beta}}(\tau^\star)$ parameters are trained.
Given the optimal set of parameters $\hat{\boldsymbol{\vartheta}}$, from (4.12) we estimate:

$$\hat{Q}^{CANN}_{\log(\tilde{S}_i)}(\tau^\star) = \left\langle \hat{\boldsymbol{\beta}}(\tau^\star), \boldsymbol{x}_i \right\rangle + \left\langle \hat{\boldsymbol{\theta}}^{(K+1)}, \left(\boldsymbol{z}^{(K)}(\hat{\boldsymbol{\theta}}^{(K)}) \circ \cdots \circ \boldsymbol{z}^{(1)}(\hat{\boldsymbol{\theta}}^{(1)})\right)(\boldsymbol{x}_i) \right\rangle \quad (4.15)$$

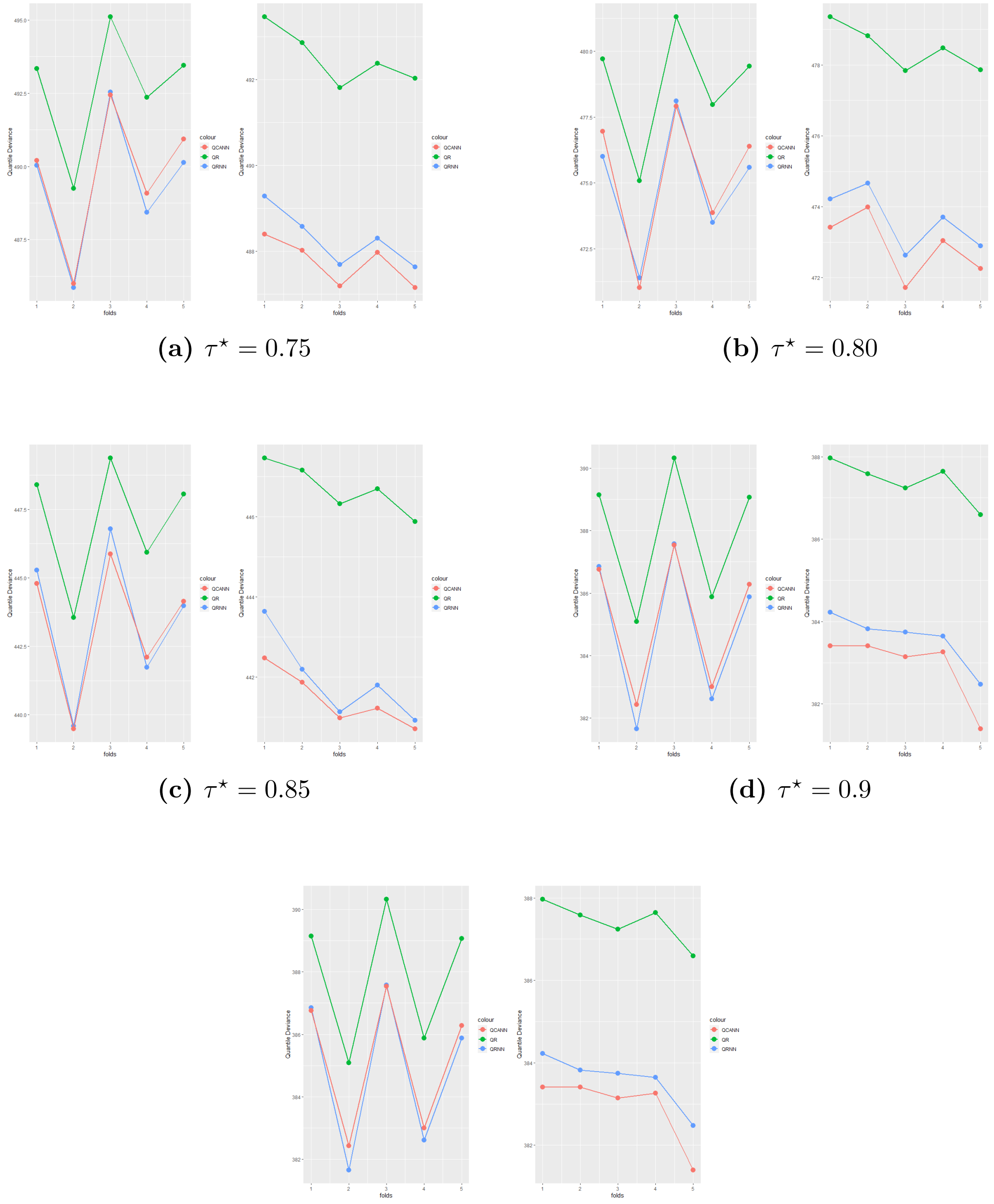
then, the quantile of the claim severity is obtained as $\hat{Q}^{CANN}_{\tilde{S}_i}(\tau^\star) = \exp(\hat{Q}^{CANN}_{\log(\tilde{S}_i)}(\tau^\star))$. Note that if the Quantile-CANN model does not return any improvement with respect to the QR model it means that the latter is already able to capture all the relevant information incorporated in the data.
In the following of this chapter we analyze the results obtained using the models presented in this section to the health insurance portfolio presented in Sections 2.1 and 4.1.

4.4 Results and discussion

In order to have a general overview of the performance of the QR, QRNN, and Quantile-CANN models, in this section, we estimate the conditional quantile of the aggregate claim severity ($\tilde{S}_i$) at different $\tau^\star$ levels for each insured in our portfolio. Precisely, the performance of each model is measured in terms of the quantile loss function (see Eq. 4.6), where the lower the loss, the better the model.
The network models are trained over $2,000$ epochs using early stopping on the validation set in order to prevent overfitting. Both models adopt a three hidden layer structure of dimension $(20, 15, 10)$, we consider the hyperbolic tangent Eq. 1.9 activation function for QRNN, while we use the ReLu Eq. 1.9 activation function for Quantile-CANN. As for the variables presented in Table 2.1: **AG**, **PE**, **DM** are Min-Max scaled, **GE** is dummy encoded, while **RE** and **FA** are treated using a $d = 1$ embedding layer. As for the QR model we consider a splines function to model the **AG** and the **PE** effects. Also in this case here we adopt a five-fold crossvalidation.

4.4.1 Performance

For each model and each fold in the cross-validation process we estimate the conditional quantile of the total claim severity $Q_{\tilde{S}_i}(\tau^\star)$ at levels $\tau^\star = (0.7, 0.75, 0.80, 0.85, 0.9)$ and compute the respective in sample and out of sample quantile loss function to evaluate their performance. From the results reported in Figure 4.3 we clearly observe that QRNN and Quantile-CANN exhibit an overall better performance in terms of the quantile loss function compared to the classical QR, for each quantile level $\tau^\star$ and each fold. It is also worth mentioning that the Quantile-CANN always yields a lower quantile loss function than the QRNN on the out-of-sample set, while the two network models display approximately the same score on the in-sample. This result suggests that building the network around the QR has not only improved the performance given by the QR model but also beats the QRNN since it provides the model with a greater ability to generalize to new and unseen data.



(a) $\tau^{\star} = 0.75$

(b) $\tau^{\star} = 0.80$

(c) $\tau^{\star} = 0.85$

(d) $\tau^{\star} = 0.9$

(e) $\tau^{\star} = 0.9$

Figure 4.3. Quantile deviance for the different models at different $\tau^{\star}$ levels for the five folds in the cross-validation process.

As discussed in the previous chapters, assessing the model's performance is not enough to decide which model is better. It is also necessary to consider the model's explainability. Therefore, in the remainder of this section, we present the results issued by the different well-known model agnostic tools. More specifically, for the sake of brevity, we will carry out our analysis on the quantile models defined at the

$\tau^\star = 0.80$ level and fitted on the first fold of the cross-validation process.

4.4.2 Importance

In Figure 4.4, we report the Variable Importance metric to find the most relevant variables in our dataset for the quantile models. The variables are ranked from top to bottom, starting with the most important one as measured by the Variable Importance. For all models, the most important variable is the age (**AG**) followed by the regional variable (**RE**) and the family member type (**FA**). The latter (**FA**) scores second in the Quantile-CANN, while it reaches third place in the QR and the QRNN. With regards to the regional variable (**RE**), we have a second place in the QR and the QRNN models and a third place in the Quantile-CANN. The other variables are less relevant, with the sole exception for the permanence (**PE**) that registers higher importance in the Quantile-CANN. It is also worth mentioning that the age variable (**AG**) seems to have greater importance in QRNN models w.r.t to Quantile-CANN and QR. Moreover, it is possible to note that, in general, all the variables are more relevant in network models rather than in the QR. These results may be a potential sign that network models capture more information from the data w.r.t. the classical quantile regression. Note that notwithstanding the low importance measured for the variables entering the QR, the parameters corresponding to such variables appear to be significant as shown in Table B.5.

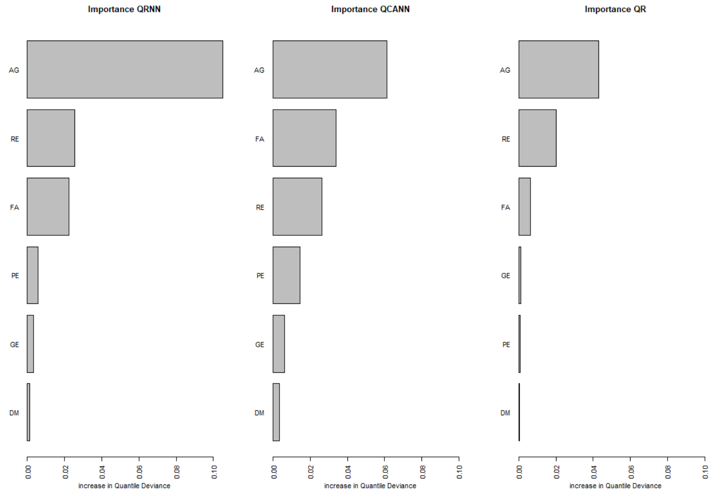


Figure 4.4. Variable Importance

4.4.3 Individual Conditional Expectation and Partial dependence

In Figures 4.6 and 4.5, we consider PD plots and individual conditional expectations to gain an understanding of the main effects of the variables over the conditional quantile of the total claim severity for the different models. The top-left plot in Figure 4.6 compares the PD plots for the **AG** variable produced by the different models. At first glance, the curves look quite similar, however taking a closer look, we notice an important difference in the leftmost part of the plot for the values below 30 years of age. More specifically, for the Quantile-CANN, we observe an upward trend in the riskiness of the insured, starting from $1,000$ euros and peaking at almost $2,000$ euros at around 15 years of age, then the curve falls off down to approximately $1,000$ euros at 30 years. In contrast, the QRNN's age main effect displays a clear upward trend starting at 400 euros for newborns and reaching $1,000$ at 30 years of age. The QR shows an almost flat trend kicking off at around $1,000$ euro and slowly increasing after 25 years of age. The behaviour displayed by the Quantile-CANN seems to be more reasonable since it captures the cost for dental treatments at younger ages, as it is usually low for children below ten years, while it raises significantly for teenagers often associated with the use of dental braces. The ICE curves associated with the PD plot displayed in the top-left plot of Figure 4.5 seem to reveal some interactions within the Quantile-CANN model since the plot shows some very wild profiles. Also, the QRNN seems to have an interaction. Of course, the ICE profiles for the QR are always parallel (on a log scale) since the variables are always modeled additively, and the model does not incorporate interactions.

The permanence (**PE**) PD plots display a similar evolution but at different levels (top-right pane in Figure 4.6). Furthermore, for this variable, we observe some messy ICE profiles (top-right pane in Figure 4.5) that may signal the presence of interactions with other variables. The gender (**GE**) variable does not seem to have a particular behaviour. In fact, the range for the y-axis is pretty narrow, even though the range of values appears to be larger for the quantile regression model. As for the regional variable (**RE**), the PD plots show almost identical profiles, with particular riskiness associated with insured living in Lazio and Lombardia. The main effect for the dimension variable (**DM**), reported in the bottom-left plot, shows almost the same increasing behaviour for the three models. Moreover, the y-axis range is rather limited, swinging between $1,520$ and $1,620$ euros. The PD plots for family member type (**FA**) reported in the bottom-right pane of Figure 4.6 shows an almost flat effect for the QRNN, while it has a diversified effect in the QR and the Quantile-CANN. In particular, such models associate lower riskiness to 'Children' and 'Ex-Spouse' and slightly higher riskiness to 'Policyholders'.

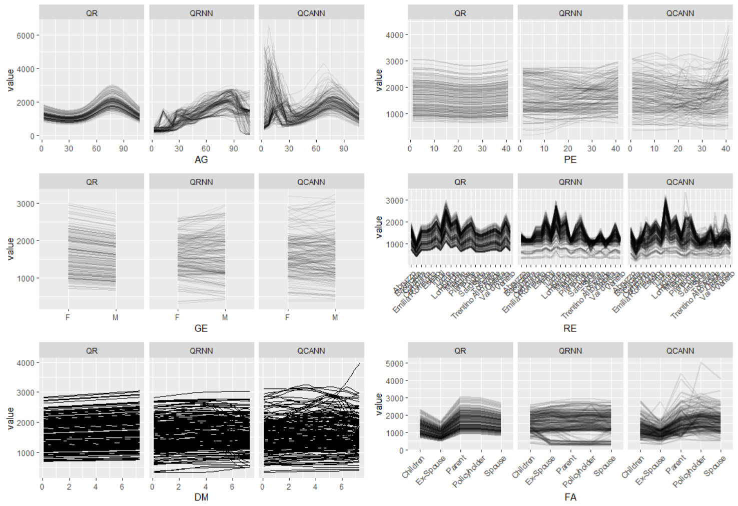


Figure 4.5. Individual conditional expectation

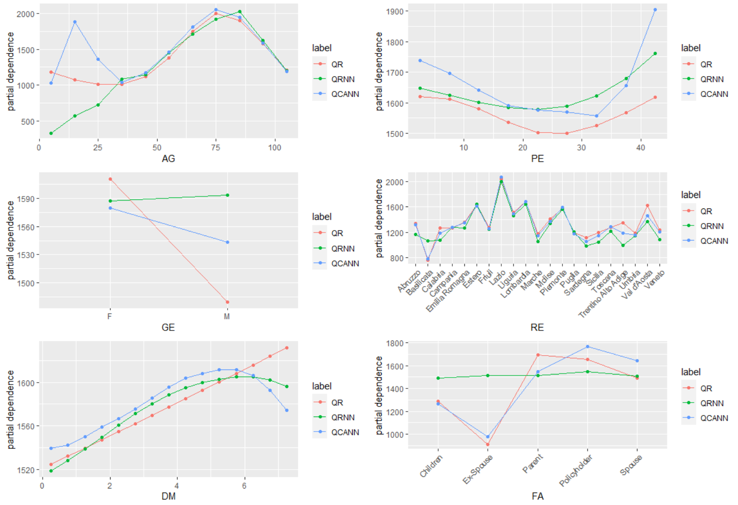


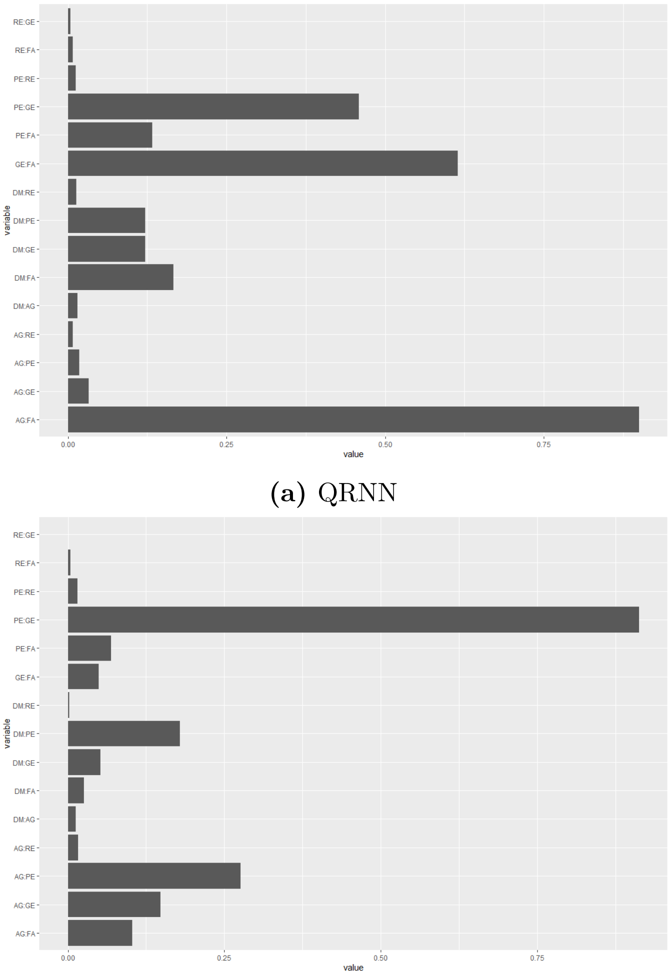
Figure 4.6. Partial dependance

In the following we deepen the analysis studying possible interactions between variables.

4.4.4 Studying interactions

In order to study possible interactions, in Figure 4.7 we plot the values of the H-statistic, introduced in 2.22, for each model and each possible pairwise interaction. Both QRNN and Quantile-CANN display some specific interactions. The strongest interaction for the QRNN are those between: age (**AG**) and family member type

(**FA**), gender (**GE**) and family member type (**FA**), permanence (**PE**) and gender (**GE**), dimension (**DM**) and family member type (**FA**). While for the Quantile-CANN we observe the following relevant interactions: permanence (**PE**) and gender (**GE**), age (**AG**) and permanence (**PE**), firms dimension (**DM**) and permanence (**PE**), age (**AG**) and gender (**GE**).



(a) QRNN

(b) Quantile-CANN

Figure 4.7. H-statistic - for the possible interactions characterizing network QRNN and Quantile-CANN.

Following the same reasoning of the previous chapters, looking at the H-statistic can be misleading since the value of this statistic does not necessarily prove the interaction to be relevant. In particular, this may happen when the covariates interacting are correlated. Therefore, in order to properly evaluate the soundness of such interactions, we try to gain knowledge on their behaviour using the Grouped PD plots reported in Figures 4.8 and 4.9.

Figure 4.8 displays the different grouped PD plots for the top four interactions captured by the QRNN. The top-left pane of Figure 3.10 presents the grouped PD plot for the age variable w.r.t. the family member type, where for ease of discussion, Ex-Spouse and Parents are dropped. The curves show almost the same shape. However, we observe a strong peak at younger ages for 'Children' insured that motivates the interaction. The effect displayed by the top-right plot of Figure 4.8 is pretty straightforward. The interaction returns a higher risk for non-Policyholder male insured w.r.t to their female peer. While male and female 'Policyholders' display almost the same riskiness. The interaction between permanence (**PE**) and gender (**GE**) is represented in the Grouped PD plots in the bottom-left pane. The graphs look almost parallel for low permanence values (below 20 years), while we observe increasingly high riskiness for male insured at higher permanence values. Meaning that enfranchised insured are more prone to have a high expenditure if they are male. As shown by the bottom-left plot of Figure 4.8, the interaction between **DM** and **FA** does not seem to be significant since the different PD plots are nearly parallel.

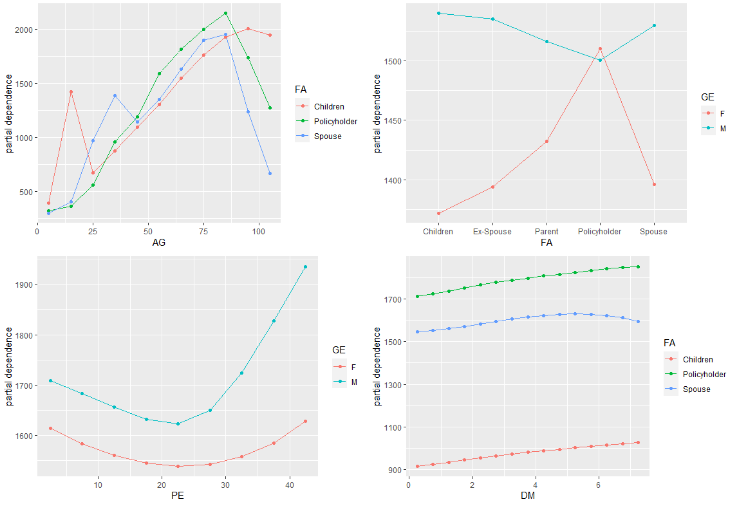


Figure 4.8. Grouped Partial dependance

In Figure 4.9 we report the grouped PD plots for the four interactions in terms of H-statistic in the Quantile-CANN. The interaction between **PE** and **GE** presented in the top-left plot of Figure 4.9 displays an higher riskiness for males with a permanence above 20 years. While the grouped PD plots below 20 years of permanence present nearly the same behaviour. The interaction between age (**AG**) and permanence (**PE**) (top-right plot) should be taken with a grain of salt since, as already discussed in the previous chapter for Figure 3.10, some associations between the permanence and age values are incompatible. For instance, it is impossible to have a 25 years old insured with 40 or 30 years of permanence. In fact, the PD plot conditional on a permanence of 40 years in the top-right pane of Figure 4.9 (in light blue) only makes sense for an insured with more than 40 years of age. Moreover, we observe that most PD plots look roughly alike. The only big difference we notice is connected to the PD plot conditional on a permanence of 40 years, which, as discussed above, is only marginally informative. Hence, the relevance of this interaction is rather dubious, and its high H-statistic (0.27) may also be due to the correlation between **AG** and

PE.
The bottom-left pane in Figure 4.9 presents the grouped PD plots for the interaction between **DM** and **PE**. From the pane, we observe that some PD curves look almost parallel with an increasing trend w.r.t. **DM** (i.e., those conditional to a low permanence), while others have a heterogeneous behaviour or a decreasing trend. Therefore, the permanence seems to have some diversified effect on the main effect of the dimension.
The interaction between age **AG** and gender **GE**, presented in the last pane, shows a strongly relevant effect for insured below 25 years of age. In particular, the interaction associates higher riskiness for young male insured w.r.t. female teenagers.

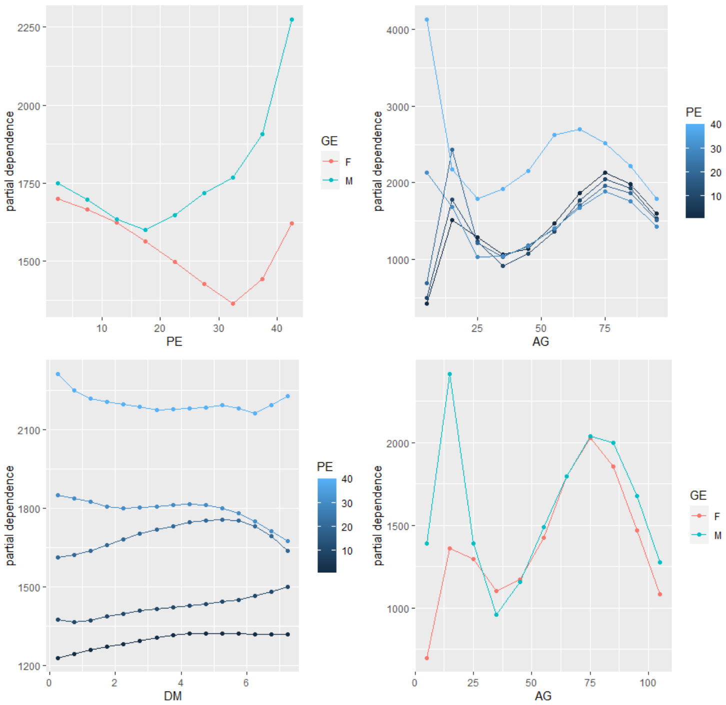


Figure 4.9. Grouped Partial dependance

As shown in this section, the network models (QRNN and Quantile-CANN) seem to have a hedge over the classical QR model. In particular, we have seen that our models offer strong flexibility that results in a better in-sample and out-of-sample performance. This versatility is partially motivated by their ability to obtain a better fit for the covariates' main effects and their ability to automatically detect covariates interactions (even though some have proven to be scarcely informative). However, as an analogy to the third chapter, to assess the possible merits of a given pricing (or costing) model, it is also essential to consider its ability to produce reasonable premiums. For this reason, in the next section, we compare the premium produced by the different quantile models using a widespread insurance-based metric discussed in [23].

4.5 Ratemaking and Premium evaluation

Above we have investigated models' behaviours evaluated at different quantile levels. We are now interested in evaluating models' performances when the focus is estimating the Quantile Premium Principle introduced by [30], where the premium paid by the insured is loaded according to its potential riskiness. In particular, we consider the convex combination of the quantile claim severity $Q_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$ and the conditional expected value of the total claim amount S_i:

$$\pi_i^L = \gamma \cdot \hat{Q}_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i) + (1-\gamma) \cdot E(S_i|\boldsymbol{x}_i), \tag{4.16}$$

where $0 < \gamma < 1$ is the loading factor and $E(S_i|\boldsymbol{x}_i)$ the conditional expected value of the total claim amount computed in Eq. 3.13 using the network models proposed in Chapter 2 and 3. Thus, in order to compute Eq. 4.16 we need to obtain $\hat{Q}_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$ using the two-part model discussed in Section 4.2.

The two-part model is estimated taking the first fold in the 5 fold cross-validation, where we consider a 60-20-20 split between learning, validation, and testing set. The first step of the two-part model consists in estimating the no claim probability p_i for each insured using logistic regression. Then as discussed in Section 4.2, the estimated p_i is employed to compute the $\tau_i^\star$ level as in Equation 4.2 for each insured setting $\tau = 0.95$. As a result of this first step we obtain 47,120 unique values for $\tau_i^\star$ ranging between 0.799 and 0.896. Following the two part approach designed by [30] would involve fitting a regression model for each different quantile level $\tau_i^\star$. Doing this would be rather time-consuming. Thus, to avoid that, we approximate the $\tau_i^\star$ values up to the second digit, where the second digit is rounded to the closest even digit.[3] Lastly, we perform the second step of the two-part model by computing the conditional quantile of the claim severity $\hat{Q}_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$ using QR, QRNN, and Quantile-CANN.

In order to test the ability of the different models to accurately estimate the desired quantile level of $\tilde{S}_i$, we use the backtest criteria approach. More specifically, the models are backtested using the unconditional coverage (UC) test proposed by [37] (see Appendix C), which is a widespread testing technique generally employed to validate VaR models in the financial literature. This technique consists in a binomial test checking if the proportion of insured with a claim severity above the conditional quantile is consistent with the predefined quantile level $\tau^\star$. The UC test performs a likelihood ratio test, where the null hypothesis states that the unconditional probability of a violation[4] is equal to $(1-\tau^\star)$.

In Table 4.3, we report the backtesting results for QR, QRNN, and Quantile-CANN. In the left part of the table, we report the unconditional coverage test statistic LR_{uc}, while on the right side of the table, we display the corresponding p-values. QRNN and Quantile-CANN pass the backtest at all quantile levels, with the sole exception of the $\tau^\star = 0.86$ and $\tau^\star = 0.88$ levels for the QRNN and the Quantile-CANN respectively, while QR fails the backtest at two quantile levels: 0.86 and 0.88. Therefore, QRNN and Quantile-CANN seem more able to accurately estimate the quantile of

[3] For instance: $\tau_i^\star = 0.815$ is rounded up to 0.82, while $\tau_i^\star = 0.805$ is rounded down to 0.80. This approximation results in six different quantile levels $\tau_i^\star$ for the testing set.

[4] We have a violation when we observe an insured submitting a larger total claim severity w.r.t. the estimated conditional quantile at level $\tau_\star$

the total claim severity $\hat{Q}_{\tilde{S}_i}(\tau_i^\star|\boldsymbol{x}_i)$ accurately.

$\tau^\star$	LR_{uc}			P-values		
	QR	QRNN	Q-CANN	QR	QRNN	Q-CANN
0.8	0.892*	0.892*	0.892*	0.344	0.344	0.344
0.82	0.277*	0.000*	0.000*	0.598	0.984	0.984
0.84	0.145*	0.026*	0.002*	0.702	0.869	0.958
0.86	6.436	6.436	1.072*	0.011	0.011	0.300
0.88	13.722	0.366*	5.769	0.000	0.544	0.016
0.9	1.666*	0.092*	2.342*	0.196	0.761	0.125

Table 4.3. For the different models we report the values for the LR_{uc} statistic and its corresponding p-values. The critical values of the LR_{uc} statistic is 3.84, denoting that the null hypothesis is rejected at the 5% significance level. When an asterisk is reported the model passes the test.

Once the models are estimated, we can compute tariffs π_i as in Eq. (4.16) and evaluate them using the ordered Lorenz curve introduced by [23]. Below we do not consider the goodness of lift metrics presented in Section 3.4 since such metrics are not designed to gauge the suitability of a loaded premium, such as the π_i of Eq. 4.16. The ordered Lorenz curve is a twist on the classical Lorenz curve usually employed in welfare economics to represent social inequality via the Gini index, see [23]. In insurance literature, the ordered Lorenz curve is employed to compare different tariff structures issued by a set of competing models. Given a base tariff structure π_i^{base} and a competing tariff π_i^{comp} the Lorenz curve proposed by [23] is ordered according to the relativity r_i:

$$r_i = \frac{\pi_i^{comp}}{\pi_i^{base}} \tag{4.17}$$

A relativity r_i consistently below 1 reveals a largely profitable policy for the company, that is likely to be lost to a competing insurance company proposing a cheaper premium. Instead a relativity r_i greater than 1 signals an underpriced policy. Of course these statement hold true only if we assume P_i^{comp} to give a sharper representation of the real risk compared to π_i^{base}.

Given r_i, the ordered Lorenz curve can be defined as follows:

$$\left(\frac{\sum_{i=1}^n S_i \mathbb{I}\left\{F_n(r_i) \leq s\right\}}{\sum_{i=1}^n S_i}, \frac{\sum_{i=1}^n \pi_i^{base} \mathbb{I}\left\{F_n(r_i) \leq s\right\}}{\sum_{i=1}^n \pi_i^{base}} \right). \tag{4.18}$$

for $s \in [0,1]$ where $F_n(r_i)$ is the empirical cumulative distribution function of the relativities r_i. The idea behind the ordered Lorenz curve is that a model producing tariffs with a greater Gini index produces a more robust separation among premiums paid by the insured, signaling that such a model can distinguish good risks from bad risks. Hence, a tariff structure π_i^{comp} that yields a larger Gini index is likely to result in a more profitable portfolio because of a better risk differentiation.

Table 4.4 displays the two-way comparison of Gini indices for the Network models and the QR. The rows report the model generating the base tariff structure π_i^{base} whereas the column stores the model from which the competing tariff structure π_i^{comp} is generated. The approach we use for selecting a tariff based on the Gini index is the 'mini-max' strategy designed by [23], which consists of selecting the model that provides the minimum Gini index among the maximal Gini indices taken over the competing models. The strategy is relatively intuitive: if we have to choose a base premium, we choose the one with the minimal maximum improvement compared to other models, meaning that the selected base premium is the least vulnerable to alternatives. In practice, we look for the base model with the lowest value in bold in Table 4.4. The Quantile-CANN appears as a clear winner since the other models cannot achieve a high Gini index when considered an alternative (see the last row), signaling that this model leads to a tariff structure that is the least likely to incur in adverse selection. The QRNN tariff structure achieves second place, followed by QR.

	Competing model		
Base model	QR	QRNN	Q-CANN
QR	-	2.07	**3.44**
QRNN	1.32	-	**2.87**
Q-CANN	**0.47**	0.047	-

Table 4.4. Two-Way Comparison of Gini Indices for the models

Hence, the proposed quantile network models not only outperformed the quantile regression from a mere statistical standpoint, as shown in Section 4.4, but they also produce more competitive loaded premiums that seem more able to differentiate between good and bad risks.

Chapter 5

Final remarks, limitations and further developments

In the domain of insurance ratemaking, neural networks may be considered as a tool to perform high-dimensional non-linear regressions. The actuarial literature on neural networks has mainly focused on car and property insurance, while little work has been done towards the application of such models in health insurance. Following this line, in this work, we have explored three applications for such techniques within the context of health insurance pricing. At first, we have proposed a neural network with multivariate response to model possibly correlated health claim counts. In particular, we have implemented a deep neural network with a three output layer structure minimizing a Negative Multinomial deviance to account for different health claim types, namely medical visits, dental care treatments, and diagnostic exams. In order to define the pure premium of the considered health insurance coverage, we have coupled the Negative Multinomial model with Gamma Neural Networks, a novel approach designed to compute the expected cost of a given claim. Such network models have been tested against a traditional GLM approach. More specifically, we have considered a Negative Multinomial GLM and Gamma GLMs for claim frequency and claim severity estimation, respectively. The comparison highlighted that the proposed neural network approach holds some value over the traditional GLM. In fact, not only do our models outperform the classical GLMs in terms of the loss function, as shown in Figures 2.6 and 3.4, but they also provide the modeler with additional information on the data structure learned by the model. In particular, this kind of information is retrieved using a suite of model agnostic tools presented in Section 2.3. Our analysis shows that the neural networks, which as known present a high degree of flexibility, allow us to find a good fit and to capture eventual non-multiplicative interactions among the variables involved in the model. However, these model agnostic tools display somewhat simplified trends that only hint at the effect captured by the model, e.g., Partial Dependence plots and Grouped Partial Dependence plots. This insight is valuable and increases the interpretability of such models but is often insufficient in the insurance pricing industry, where the regulator demands strong model transparency [1]. Bearing in mind these limitations,

[1] Note that the issue related to the model's transparency does not hold in the context of Integrated Health Funds (IFHs) characterizing the Italian market of employer-based collective health insurance

we still believe that neural networks, if properly combined with the model agnostic tools discussed in this work and put forward by [39], do offer a relevant added value to the analyst that can be used to enhance a simpler model such as a GLM. Moreover, as shown in Section 3.4 via model lift metrics, the merits of the proposed network models go beyond the mere statistical performance and translate into a set of more efficient pure premiums that result in a better risk diversification for the insurance coverage considered in this work.

Moreover, we have proposed a new quantile neural network approach to compute the quantile of the aggregate claim amount, which serves as a risk measure for the insured. In particular, we have proposed two different models: Quantile Regression Neural Networks (QRNN) and the Quantile-CANN. The performance of such models was tested the traditional quantile regression model over the health insurance dataset presented in Section 2.1. The results have shown that, not only our models outperform quantile regression in terms of quantile loss function, but they also exhibit a better tariff structure w.r.t QR since they provide a better separation among the risks in the portfolio. This feature is paramount for the insurer since a better differentiation between good and bad risks is likely to produce higher profits for the company.

However, even if neural networks have proven to perform better than a classical regression approach, they should not be regarded as a straightforward substitute for more established methods such as GLMs and GAMs (or even Quantile regression). Machine learning models shall be regarded as complementary tools to improve simple regression models. In fact, nowadays, AI and Machine Learning are used by pricing teams across Europe as a tool to perform in-depth exploratory analysis, while they still struggle to find use in production due to their complexity and lack of transparency. For instance, networks can be used to inform the analyst on how to improve a given basic model by changing the shape of the marginal effect of a variable (using a polynomial variate or a spline) or by factoring a specific interaction effect between variables spotted via the network model. Another clear example for the complementarity of network models is the CANN approach discussed in Chapter 4, where it is possible to build a neural network on top of a simple regression model. Using the CANN framework, the performance of the starting model is enhanced via the network structure, which allows incorporating additional information in the model. For these reasons, the pricing industry is slowly moving towards transparent use of automated machine learning solutions used in conjunction with GLMs and GAMs to provide more accurate and competitive insurance tariffs.

A possible development of this work could consider using Recurrent Neural Networks (RNN) to model the health expenditure for the policyholders according to their past health consumption. A recurrent neural network (RNN) is a class of neural networks that model sequential data or time-series data. State of the art in this field is represented RNNs with Long short-term memory (LSTM) cells that have already found application in mortality forecasting [46]. In particular, we could implement a

since this kind of institution dramatically lacks a clear regulatory framework and a vigilance authority.

Negative Binomial LSTM network that recursively predicts the claim counts for each policyholder based on its past claim counts. This approach may allow the model to incorporate past policyholder behaviour to predict its future claim frequency. This feature would be particularly interesting in health insurance, where the policyholder medical expenditure is often correlated over time. For instance, think of policyholders that each year undergo a sample of predefined medical check-ups or policyholders that have an increasingly high number of claims due to a worsening of their health condition, or even policyholders that make great use of preventive healthcare at younger ages that may result in future lower claim counts.In order to explore this approach, more details on the claims reported in Section 2.1 would be needed since the partition provided by the three groups of claims (*Visits*, *Dentalcare* and *Diagnostic*) is far too broad. In particular, each group of claims should be disaggregated into the single health care services belonging to a said group; this is key to providing more insight into the policyholder's health expenditure.

With reference to the models discussed in Chapters 2 and 3 it would be interesting to extend the CANN approach adopted in Chapter 4. Given the promising results obtained via the Quantile-CANN, we could expect to earn a good performance for a future Negative Multinomial CANN and a Gamma CANN, which could also exceed the results achieved via the Negative Multinomial NN and the Gamma NN.

Another possible development of this work, concerning Chapter 4, might consider a multivariate QRNN or Quantile-CANN approach to jointly model the conditional quantile of the total claim severity for different and possibly correlated claim types. It would also be worth exploring quantile models designed on the claim frequency rather than the aggregate claim amount. This idea makes sense in the health insurance domain since most riskiness stems from the claim frequency rather than the claim size.

Despite the above-mentioned limitations, this thesis delivers an in-depth discussion on the application of neural networks and model agnostic tools to health insurance pricing. The implementation of such techniques to health insurance is a hopefully valuable innovation, since this specific insurance branch is often disregarded by actuarial literature compared to car and P&C insurance. Hence, we believe that the original contribution provided by this work could foster the discussion around the use of neural networks in health insurance.

Appendices

Appendix A

Proofs for Back propagation

Proof Eq.(1.19) Let us start with Eq.(1.19), which given an expression of the error in the output layer $\boldsymbol{\delta}^F$:

$$\boldsymbol{\delta}^F = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}^F} \tag{A.1}$$

Applying the chain rule, we can re-write the partial derivative above in terms of partial derivatives with respect to the output activation

$$\boldsymbol{\delta}^F = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{z}^F} \frac{\partial \boldsymbol{z}^F}{\partial \boldsymbol{a}^F} \tag{A.2}$$

recalling, from Eq.(1.15), that $\boldsymbol{z}^F = \psi(\boldsymbol{a}^F)$, we have

$$\boldsymbol{\delta}^F = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{z}^F} \psi'(\boldsymbol{a}^F) \tag{A.3}$$

Furthermore from Eq.(1.17) $\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{z}^F} = 2 \cdot [\boldsymbol{Y} - \boldsymbol{z}^F(\boldsymbol{\theta})]$. Hence we have:

$$\boldsymbol{\delta}^F = 2 \cdot [\boldsymbol{Y} - \boldsymbol{z}^F(\boldsymbol{\theta})] \cdot \psi'(\boldsymbol{a}^F) \tag{A.4}$$

Proof Eq.(1.20) Here the goal is to express $\boldsymbol{\delta}_j^{(s)}$ in terms of the errors in the next layer $\boldsymbol{\delta}_j^{(s+1)}$. Hence, starting from

$$\boldsymbol{\delta}_j^{(s)} = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}_j^{(s)}} \tag{A.5}$$

we rephrase it as:

$$\boldsymbol{\delta}_j^{(s)} = \sum_{l=1}^{q_s} \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}_l^{(s+1)}} \frac{\partial \boldsymbol{a}_l^{(s+1)}}{\partial \boldsymbol{a}_j^{(s)}} = \sum_{l=1}^{q_s} \boldsymbol{\delta}_j^{(s+1)} \frac{\partial \boldsymbol{a}_l^{(s+1)}}{\partial \boldsymbol{a}_j^{(s)}} \tag{A.6}$$

To evaluate the second term on the right hand side, note that:

$$\boldsymbol{a}_l^{(s+1)} = \sum_{k=1}^{q_s} \theta_{lk}^{(s+1)} \phi(\boldsymbol{a}_k^{(s)}) + \theta_{l0}^{(s+1)} \tag{A.7}$$

Differentiating, we have

$$\frac{\partial \boldsymbol{a}_l^{(s+1)}}{\partial \boldsymbol{a}_j^{(s)}} = \theta_{lj}^{(s+1)} \phi'(\boldsymbol{a}_j^{(s)}) \tag{A.8}$$

Feeding back into Eq.(A.6):

$$\boldsymbol{\delta}_j^{(s)} = \sum_{l=1}^{q_s} \boldsymbol{\delta}_l^{(s+1)} \theta_{lj}^{(s+1)} \phi'(\boldsymbol{a}_j^{(s)}) \tag{A.9}$$

Proof Eq.(1.21) To prove Eq.(1.21) we simply apply the chain rule to $\frac{\partial \boldsymbol{L}}{\partial \theta_{0j}^{(s)}}$:

$$\frac{\partial \boldsymbol{L}}{\partial \theta_{0j}^{(s)}} = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}_j^{(s)}} \frac{\partial \boldsymbol{a}_j^{(s)}}{\partial \theta_{0j}^{(s)}} \tag{A.10}$$

note that the first term in the right hand side is equivalent to $\boldsymbol{\delta}_j^{(s)}$ while the second derivative is equal to one, thus we have:

$$\frac{\partial \boldsymbol{L}}{\partial \theta_{0j}^{(s)}} = \boldsymbol{\delta}_j^{(s)} \cdot 1 = \boldsymbol{\delta}_j^{(s)} \tag{A.11}$$

Proof Eq.(1.22) Also to prove Eq.(1.22) we apply the chain rule to $\frac{\partial \boldsymbol{L}}{\partial \theta_{jk}^{(s)}}$:

$$\frac{\partial \boldsymbol{L}}{\partial \theta_{jk}^{(s)}} = \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}_j^{(s)}} \frac{\partial \boldsymbol{a}_j^{(s)}}{\partial \theta_{jk}^{(s)}} \tag{A.12}$$

for the first term in the right hand side we have $\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{a}_j^{(s)}} = \boldsymbol{\delta}_j^{(s)}$. While, to evaluate the second term, we note that:

$$\boldsymbol{a}_j^{(s)} = \theta_{j,0}^{(s)} + \sum_{l=1}^{q_{s-1}} \theta_{j,l}^{(s)} \cdot \boldsymbol{z}_l^{(s-1)}(\boldsymbol{\theta}_l^{(s-1)}) \tag{A.13}$$

Differentiating w.r.t. $\theta_{jk}^{(s)}$ we have

$$\frac{\partial \boldsymbol{a}_j^{(s)}}{\partial \theta_{jk}^{(s)}} = \boldsymbol{z}_k^{(s-1)}(\boldsymbol{\theta}^{(s-1)}) \tag{A.14}$$

Substituing back into Eq.(A.12):

$$\frac{\partial \boldsymbol{L}}{\partial \theta_{jk}^{(s)}} = \boldsymbol{\delta}_j^{(s)} \cdot \boldsymbol{z}_k^{(s-1)}(\boldsymbol{\theta}^{(s-1)}) \tag{A.15}$$

Appendix B

Covariates Wald Test for NM-GLM

Unlike regression for univariate responses, the NM-GLM has a matrix of parameters $\boldsymbol{\beta} = (\boldsymbol{\beta}_j, \ldots, \boldsymbol{\beta}_p) \in \mathbb{R}\ r \times p$, with each column corresponding to the effect of one predictor. As discussed by the authors [33], the significance of the parameters belonging to a given covariate is assessed via a Wald test on $\boldsymbol{\beta}$. Below we report the Wald test for the different covariates, where the categorical variables are dummy encoded. As it is possible to see in Table B.1 almost every covariate turns out to be significant.

Covariate	Wald statistic	P-value
AG	3617.68*	-
DM	522.73*	0.00
PE	406.56*	0.00
RE - Abruzzo	130.10*	0.00
RE - Basilicata	90.79*	0.00
RE - Calabria	193.23*	0.00
RE - Campania	305.52*	0.00
RE - Emilia Romagna	153.46*	0.00
RE - Estero	167.96*	0.00
RE - Friuli	112.47*	0.00
RE - Lazio	218.46*	0.00
RE - Liguria	172.12*	0.00
RE - Lombardia	91.90*	0.00
RE - Marche	103.86*	0.00
RE - Molise	68.80*	0.00
RE - Piemonte	126.64*	0.00
RE - Puglia	199.00*	0.00
RE - Sardegna	73.60*	0.00
RE - Sicilia	154.60*	0.00
RE - Toscana	168.85*	0.00
RE - Trentino Alto Adige	60.10*	0.00
RE - Umbria	199.83*	0.00
RE - Val d'Aosta	60.81*	0.00
RE - Veneto	99.31*	0.00
FA - Coniuge	21.35*	0.00
FA - ex-Coniuge	4.77	0.19
FA - Figlio	55.85*	0.00
FA - Titolare	29.46*	0.00
GE - M	1007.71*	0.00

Table B.1. Wald test for the covariates parameters in the NM-GLM A P-value below 0.05 denotes the significance of the covariate parameters at a 95% confidence level. When the parameter is significant an asterisk is reported.

Covariate	Estimate	Std Error	T-statistic	P-value
(Intercept)	4.56*	0.01	325.89	-
AG-s1	0.01	0.02	0.54	0.59
AG-s2	-0.06*	0.01	- 5.73	0.00
AG-s3	0.05	0.03	1.64	0.10
AG-s4	0.02	0.02	0.86	0.39
PE-s1	- 0.01*	0.00	- 2.19	0.03
PE-s2	0.01	0.01	1.54	0.12
PE-s3	0.02*	0.00	4.50	0.00
RE-Basilicata	- 0.02	0.04	- 0.55	0.58
RE-Calabria	0.03	0.03	0.75	0.46
RE-Campania	- 0.09*	0.01	- 6.55	0.00
RE-Emilia Romagna	- 0.00	0.01	- 0.26	0.80
RE-Estero	- 0.05*	0.02	- 2.73	0.01
RE-Friuli	0.01	0.01	0.40	0.69
RE-Lazio	0.02	0.01	1.77	0.08
RE-Liguria	0.05*	0.01	3.52	0.00
RE-Lombardia	0.12*	0.01	9.51	0.00
RE-Marche	- 0.01	0.02	- 0.57	0.57
RE-Molise	- 0.02	0.05	- 0.48	0.63
RE-Piemonte	- 0.05*	0.01	- 3.94	0.00
RE-Puglia	- 0.04*	0.02	- 2.57	0.01
RE-Sardegna	- 0.17*	0.02	- 7.80	0.00
RE-Sicilia	- 0.09*	0.02	- 5.59	0.00
RE-Toscana	0.01	0.01	1.02	0.31
RE-Trentino Alto Adige	- 0.07	0.02	- 4.19	0.00
RE-Umbria	- 0.04*	0.02	- 2.64	0.01
RE-Val d'Aosta	0.11*	0.03	3.21	0.00
RE-Veneto	0.02	0.01	1.74	0.08
GE-M	0.00	0.00	1.09	0.28
DM-s1	0.03*	0.00	7.73	0.00
DM-s2	0.02*	0.01	2.84	0.00
DM-s3	0.04*	0.01	6.76	0.00
FA-Ex-Spouse	- 0.11	0.09	- 1.32	0.19
FA-Parent	0.05	0.08	0.68	0.49
FA-Policyholder	0.05*	0.01	4.95	0.00
FA-Spouse	0.03*	0.01	3.49	0.00

Table B.2. Parameters significance for the covariates entering the *Visits* Gamma GLM. A P-value below 0.05 denotes the significance of the covariate parameters at a 95% confidence level. When the parameter is significant an asterisk is reported.

Covariate	Estimate	Std Error	T-statistic	P-value
(Intercept)	6.05*	0.03	179.42	-
AG-s1	- 0.53*	0.04	- 12.75	0.00
AG-s2	- 0.18*	0.03	- 6.22	0.00
AG-s3	- 1.77*	0.08	- 21.13	0.00
AG-s4	0.08	0.05	1.61	0.11
PE-s1	- 0.10*	0.01	- 7.60	0.00
PE-s2	- 0.01	0.02	- 0.30	0.77
PE-s3	0.01	0.01	1.06	0.29
RE-Basilicata	- 0.04	0.09	- 0.47	0.64
RE-Calabria	- 0.01	0.08	- 0.09	0.93
RE-Campania	- 0.19*	0.03	- 6.19	0.00
RE-Emilia Romagna	0.15 *	0.03	5.32	0.00
RE-Estero	0.14*	0.05	2.96	0.00
RE-Friuli	0.07*	0.03	2.18	0.03
RE-Lazio	0.02	0.03	0.92	0.36
RE-Liguria	- 0.00	0.03	- 0.17	0.86
RE-Lombardia	0.19*	0.03	7.00	0.00
RE-Marche	- 0.00	0.04	- 0.08	0.93
RE-Molise	- 0.06	0.09	- 0.67	0.50
RE-Piemonte	0.13*	0.03	4.60	0.00
RE-Puglia	- 0.08*	0.04	- 2.06	0.04
RE-Sardegna	0.13*	0.06	2.40	0.02
RE-Sicilia	- 0.00	0.04	- 0.11	0.91
RE-Toscana	0.24*	0.03	8.08	0.00
RE-Trentino Alto Adige	0.33*	0.04	8.41	0.00
RE-Umbria	- 0.02	0.04	- 0.57	0.57
RE-Val d'Aosta	0.25 *	0.08	3.29	0.00
RE-Veneto	0.09*	0.03	3.08	0.00
GE-M	- 0.02*	0.01	- 3.31	0.00
DM-s1	0.00	0.01	0.11	0.92
DM-s2	- 0.02	0.02	- 1.35	0.18
DM-s3	- 0.02	0.01	- 1.42	0.15
FA-Ex-Spouse	- 0.02	0.13	- 0.15	0.88
FA-Parent	0.16	0.16	1.04	0.30
FA-Policyholder	0.13*	0.03	5.13	0.00
FA-Spouse	0.12*	0.03	4.55	0.00

Table B.3. Parameters significance for the covariates entering the *Dentalcare* Gamma GLM. A P-value below 0.05 denotes the significance of the covariate parameters at a 95% confidence level. When the parameter is significant an asterisk is reported.

Covariate	Estimate	Std Error	T-statistic	P-value
(Intercept)	3.83*	0.04	91.65	-
AG-s1	0.01	0.04	0.24	0.81
AG-s2	- 0.10*	0.03	- 3.29	0.00
AG-s3	- 0.42*	0.09	- 4.56	0.00
AG-s4	- 0.75*	0.04	- 17.34	0.00
PE-s1	0.00	0.01	0.04	0.96
PE-s2	- 0.05*	0.02	- 2.54	0.01
PE-s3	0.04*	0.01	4.54	0.00
RE-Basilicata	- 0.30*	0.11	- 2.79	0.01
RE-Calabria	0.01	0.09	0.17	0.86
RE-Campania	- 0.85*	0.04	- 23.62	0.00
RE-Emilia Romagna	- 0.29*	0.04	- 8.21	0.00
RE-Estero	0.20*	0.05	3.98	0.00
RE-Friuli	- 0.42*	0.04	- 10.86	0.00
RE-Lazio	- 0.44*	0.03	- 12.65	0.00
RE-Liguria	- 0.57*	0.04	- 16.15	0.00
RE-Lombardia	0.03	0.03	0.79	0.43
RE-Marche	0.09*	0.05	1.93	0.05
RE-Molise	0.03	0.13	0.22	0.83
RE-Piemonte	- 0.44*	0.03	- 12.56	0.00
RE-Puglia	- 0.35*	0.04	- 8.27	0.00
RE-Sardegna	0.07	0.07	1.04	0.30
RE-Sicilia	- 0.24*	0.04	- 5.50	0.00
RE-Toscana	- 0.21*	0.04	- 5.70	0.00
RE-Trentino Alto Adige	0.34*	0.06	5.72	0.00
RE-Umbria	- 0.54*	0.04	- 12.90	0.00
RE-Val d'Aosta	0.20	0.11	1.88	0.06
RE-Veneto	0.07	0.04	1.99	0.05
GE-M	- 0.02*	0.01	- 3.06	0.00
DM-s1	0.02*	0.01	2.30	0.02
DM-s2	0.03*	0.02	2.03	0.04
DM-s3	0.03	0.01	1.97	0.05
FA-Ex-Spouse	0.08	0.17	0.45	0.65
FA-Parent	- 0.30*	0.13	- 2.37	0.02
FA-Policyholder	0.13*	0.03	4.85	0.00
FA-Spouse	0.14*	0.03	5.35	0.00

Table B.4. Parameters significance for the covariates entering the *Diagnostic* Gamma GLM. A P-value below 0.05 denotes the significance of the covariate parameters at a 95% confidence level. When the parameter is significant an asterisk is reported.

Covariate	Estimate	Std Error	T-statistic	P-value
(Intercept)	6.82*	0.08	89.42	-
AG-s1	0.24*	0.07	3.53	0.00
AG-s2	0.81*	0.05	15.37	-
AG-s3	- 0.10	0.14	- 0.71	0.48
AG-s4	0.15	0.10	1.57	0.12
PE-s1	- 0.12*	0.03	- 4.49	0.00
PE-s2	- 0.06	0.04	- 1.41	0.16
PE-s3	- 0.01	0.03	- 0.39	0.69
RE-Basilicata	- 0.56*	0.27	- 2.11	0.03
RE-Calabria	- 0.05	0.16	- 0.32	0.75
RE-Campania	- 0.06	0.07	- 0.74	0.46
RE-Emilia Romagna	0.02	0.07	0.28	0.78
RE-Estero	0.20	0.13	1.52	0.13
RE-Friuli	- 0.04	0.08	- 0.58	0.56
RE-Lazio	0.42*	0.07	6.42	0.00
RE-Liguria	0.12	0.07	1.73	0.08
RE-Lombardia	0.23*	0.06	3.53	0.00
RE-Marche	- 0.13	0.09	- 1.52	0.13
RE-Molise	0.05	0.13	0.40	0.69
RE-Piemonte	0.16*	0.07	2.41	0.02
RE-Puglia	- 0.11	0.08	- 1.48	0.14
RE-Sardegna	- 0.18*	0.07	- 2.45	0.01
RE-Sicilia	- 0.12	0.09	- 1.29	0.20
RE-Toscana	- 0.05	0.07	- 0.67	0.50
RE-TRE-ntino Alto Adige	0.01	0.11	0.07	0.95
RE-Umbria	- 0.12	0.07	- 1.61	0.11
RE-Val d'Aosta	0.19	0.40	0.49	0.63
RE-Veneto	- 0.08	0.07	- 1.10	0.27
GE-M	- 0.08*	0.01	- 6.00	0.00
DM	0.01	0.00*	3.33	0.00
FA-Ex-Spouse	- 0.34*	0.14	- 2.51	0.01
FA-PaRE-nt	0.28	0.53	0.52	0.60
FA-Policyholder	0.25*	0.04	6.34	0.00
FA-Spouse	0.15*	0.04	3.72	0.00

Table B.5. Parameters significance for the covariates entering the Quantile Regression. A P-value below 0.05 denotes the significance of the covariate parameters at a 95% confidence level. When the parameter is significant an asterisk is reported.

Appendix C

Backtesting - Kupiec test

Since the end of the 90's a wide variety of tests have been proposed to assess the accuracy and the predictive ability of VaR models. Although those tests may differ in some details many of them focus on the comparison of the predicted risk measure (in our case $\hat{Q}_{\tilde{S}_i}(\tau^\star|\boldsymbol{x}_i)$) and the actual loss. In particular, let $\tilde{S}_i$ be the aggregate claim severity for the i-th claimant, we de

fine the following hit function:

$$\text{HT}_i(\tau^\star) = \Big\{ 0 \qquad \text{if} \qquad \tilde{S}_i \leq \hat{Q}_{\tilde{S}_i}(\tau^\star|\boldsymbol{x}_i) \qquad \text{or} \qquad 1 \qquad \tilde{S}_i > \hat{Q}_{\tilde{S}_i}(\tau^\star|\boldsymbol{x}_i) \Big\} \tag{C.1}$$

where we say that a violation occurs whenever the loss suffered by the claimant is higher than the predicted risk measure. Hence the $\text{HT}_i(\tau^\star)$ function yields 1 when a violation is observed, while it returns 0 instead.

A good Value-ar-Risk model must be able to satisfy the so-called *unconditional coverage property*. According to this property the probability of observing a violation must be exactly $\tau^\star$, more formally $\mathbb{P}(\text{HT}_i(\tau^\star) = 1) = \tau^\star$. If the proportion of violations is systematically greater (less) than the confidence level $\tau^\star$, the model overestimates (underestimates) the true level of risk. In other terms, in the former case the model is thick tailed compared to the data; in the latter the model is thin tailed and excessively conservative.

Conventional tests proposed by [37] can be conducted to examine if the unconditional coverage property is fulfilled. Kupiec's unconditional coverage (LR_{uc}), by means of a likelihood ratio, inspects whether the true probability of having a violation is equal to the tail level $\tau^\star$ of the risk measure, meaning that such risk measure correctly estimates the desired quantile.

Given a sample of I_p observations and T violations, the aim of this test is to determine whether $\hat{p} = \frac{T}{I_p}$ is statistically equivalent to $\tau^\star$ (that is the level of the tail selected to compute the VaR). Thus we have this null hypothesis:

$$H_0 : \hat{p} = E(\text{HT}_i) = \tau^\star \tag{C.2}$$

where HT_i is the same as in Eq. C.1. The number of violations is obtained as $T = \sum_{i=1}^{I_p} \text{HT}_i$. The probability of observing T violations over a sample of I_p observations follows the Binomial distribution. The aforementioned null hypothesis can be tested using the following statistic:

$$LR_{uc} = 2\ln \frac{\hat{p}^T (1-\hat{p})^{I_p - T}}{\tau^{\star T}(1-\tau^{\star})^{I_p - T}} \tag{C.3}$$

that follows a chi-squared distribution with one degree of freedom. The interpretation of the LR_{uc} is straightforward: indeed, if the number of violations is greater (less) than the tail level $\tau^{\star}$, the model overestimates (underestimates) the true level of risk. In other terms, in the former case the model is thick tailed compared to the data; in the latter the model is thin tailed.

Bibliography

[1] Book reviews. *Journal of the American Statistical Association*, **83** (1988), 902. doi:10.1080/01621459.1988.10478680.

[2] Log-linear modeling with the negative multinomial distribution. *Biometrics*, **53** (1997), 971. Available from: http://www.jstor.org/stable/2533557.

[3] Adrian, T. and Brunnermeier, M. K. Covar. *American Economic Review*, **106** (2016), 1705. Predicting and measuring a financial institutions contribution to systemic risk that internalizes externalities and avoids procyclicality. Available from: StaffReports.

[4] Albrecher, H., Beirlant, J., and Teugels, J. L. *Reinsurance: actuarial and statistical aspects*. John Wiley & Sons (2017).

[5] Baione, F. and Biancalana, D. An individual risk model for premium calculation based on quantile: A comparison between generalized linear models and quantile regression. *North American Actuarial Journal*, **23** (2019), 573. arXiv:https://doi.org/10.1080/10920277.2019.1604238, doi:10.1080/10920277.2019.1604238.

[6] Bernardi, A. and Pegoraro, R. Italian drug policy: Ethical aims of essential assistance levels. *Health Care Analysis*, **11** (2003), 279. doi:10.1023/B:HCAN.0000010056.05684.22.

[7] Blier-Wong, C., Cossette, H., Lamontagne, L., and Marceau, E. Machine learning in p&c insurance: A review for pricing and reserving. *Risks*, **9** (2021). Available from: https://www.mdpi.com/2227-9091/9/1/4, doi:10.3390/risks9010004.

[8] Breiman, L. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statist. Sci.*, **16** (2001), 199. Available from: https://doi.org/10.1214/ss/1009213726, doi:10.1214/ss/1009213726.

[9] Breiman, L., Friedman, J., Stone, C., and Olshen, R. *Classification and Regression Trees*. Taylor & Francis (1984). ISBN 9780412048418. Available from: https://books.google.it/books?id=JwQx-WOmSyQC.

[10] Cannon, A. J. Quantile regression neural networks: Implementation in r and application to precipitation downscaling. *Computers and Geosciences*, **37** (2011), 1277 . Available from: http://www.sciencedirect.

com/science/article/pii/S009830041000292X, doi:https://doi.org/10.1016/j.cageo.2010.07.005.

[11] CENSIS and RBM. *RMB-Censis. VIII Rapporto RBM–Censis sulla SanitàPubblica, Privata ed Intermediata* (2019). Available from: https://www.censis.it/welfare-e-salute/welfare-day-2019.

[12] Chairman, J. L. and Pryor, L. Neural networks and glms in pricing general insurance (2004).

[13] Chapados, N., Bengio, Y., Vincent, P., Ghosn, J., Dugas, C., Takeuchi, I., and Meng, L. Estimating car insurance premia: A case study in high-dimensional data inference. *Advances in Neural Information Processing Systems*, **2** (2002), 1369.

[14] Denuit, M. and Lang, S. Non-life rate-making with bayesian gams. *Insurance: Mathematics and Economics*, **35** (2004), 627. Available from: https://www.sciencedirect.com/science/article/pii/S0167668704000940, doi:https://doi.org/10.1016/j.insmatheco.2004.08.001.

[15] Denuit, M., Sznajder, D., and Trufin, J. Model selection based on lorenz and concentration curves, gini indices and convex order. *Insurance: Mathematics and Economics*, **89** (2019), 128. Available from: https://www.sciencedirect.com/science/article/pii/S0167668719303890, doi:https://doi.org/10.1016/j.insmatheco.2019.09.001.

[16] Duan, N., Manning, W. G., Morris, C. N., and Newhouse, J. P. A comparison of alternative models for the demand for medical care. *Journal of Business & Economic Statistics*, **1** (1983), 115. Available from: http://www.jstor.org/stable/1391852.

[17] Dugas, C., Bengio, Y., Chapados, N., Vincent, P., Denoncourt, G., and Fournier, C. Statistical learning algorithms applied to automobile insurance ratemaking. In *CAS Forum*, vol. 1, pp. 179–214. Citeseer (2003).

[18] Erhardt, V. and Czado, C. Modeling dependent yearly claim totals including zero claims in private health insurance. *Scandinavian Actuarial Journal*, **2012** (2012), 106. doi:10.1080/03461238.2010.489762.

[19] France, G., Taroni, F., and Donatini, A. The italian health-care system. *Health Economics*, **14** (2005), S187. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/hec.1035, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/hec.1035, doi:https://doi.org/10.1002/hec.1035.

[20] Francis, L. Neural networks demystified. In *Casualty Actuarial Society Forum*, pp. 253–320. Citeseer (2001).

[21] Frees, E., Gao, J., and Ab, M. Predicting the frequency amount of health care expenditures. *North American Actuarial Journal*, **15** (2011). doi:10.1080/10920277.2011.10597626.

[22] Frees, E. W. *Regression Modeling with Actuarial and Financial Applications.* International Series on Actuarial Science. Cambridge University Press (2010). doi:10.1017/CBO9780511814372.

[23] Frees, E. W. J., Meyers, G., and Cummings, A. D. Insurance ratemaking and a gini index. *The Journal of Risk and Insurance*, **81** (2014), 335. Available from: http://www.jstor.org/stable/24546807.

[24] Friedman, J. H. and Popescu, B. E. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, **2** (2008), 916 . Available from: https://doi.org/10.1214/07-AOAS148, doi:10.1214/07-AOAS148.

[25] Gabrielli, A., Richman, R., and Wüthrich, M. V. Neural network embedding of the over-dispersed poisson reserving model. *Scandinavian Actuarial Journal*, **2020** (2020), 1. Available from: https://doi.org/10.1080/03461238.2019.1633394, arXiv:https://doi.org/10.1080/03461238.2019.1633394, doi:10.1080/03461238.2019.1633394.

[26] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning.* MIT Press (2016).

[27] Guelman, L. Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Syst. Appl.*, **39** (2012), 3659. doi:10.1016/j.eswa.2011.09.058.

[28] Heiner, D., Steffen, L., and Thorsten, S. Rough waters: European trade unions in a time of crises. *ETUI, The European Trade Union Institute*, (2020). arXiv:https://www.etui.org/publications/books/rough-waters-european-trade-unions-in-a-time-of-crises.

[29] Henckaerts, R., Côté, M.-P., Antonio, K., and Verbelen, R. Boosting insights in insurance tariff plans with tree-based machine learning methods. *North American Actuarial Journal*, **25** (2021), 255. doi:10.1080/10920277.2020.1745656.

[30] Heras, A., Moreno, I., and Vilar-Zanón, J. An application of two-stage quantile regression to insurance ratemaking. *Scandinavian Actuarial Journal*, **2018** (2018), 1. doi:10.1080/03461238.2018.1452786.

[31] Ismail, N. and Zamani, H. Estimation of claim count data using negative binomial, generalized poisson, zero-inflated negative binomial and zero-inflated generalized poisson regression models. In *Casualty Actuarial Society E-Forum*, vol. 41, pp. 1–28. Spring (2013).

[32] Jaderberg, M., et al. Population based training of neural networks. (2017).

[33] Kim, J., Zhang, Y., Day, J., and Zhou, H. Mglm: an r package for multivariate categorical data analysis. *The R journal*, **10** (2018), 73.

[34] Klugman, S., Panjer, H., and Willmot, G. *Loss Models: From Data to Decisions.* Wiley Series in Probability and Statistics. Wiley (2012).

ISBN 9780470391334. Available from: https://books.google.it/books?id=z0qdRiK7I_gC.

[35] Koenker, R. and Bassett, G. Regression quantiles. *Econometrica: Journal of the Econometric Society*, **46** (1978), 33.

[36] Kudryavtsev, A. A. Using quantile regression for rate-making. *Insurance: Mathematics and Economics*, **45** (2009), 296. Available from: https://EconPapers.repec.org/RePEc:eee:insuma:v:45:y:2009:i:2:p:296-304.

[37] Kupiec, P. H. Techniques for verifying the accuracy of risk measurement models. *The Journal of Derivatives*, **3** (1995), 73.

[38] Laporta, A. G., Merlo, L., and Petrella, L. Selection of value at risk models for energy commodities. *Energy Economics*, **74** (2018), 628 . Available from: http://www.sciencedirect.com/science/article/pii/S0140988318302548, doi:https://doi.org/10.1016/j.eneco.2018.07.009.

[39] Lorentzen, C. and Mayer, M. Peeking into the black box: An actuarial case study for interpretable machine learning. *Available at SSRN*, (2020). Available from: https://ssrn.com/abstract=3595944.

[40] Marenzi, A., Rizzi, D., and Zanette, M. Incentives for voluntary health insurance in a national health system: Evidence from italy. *Health Policy*, **125** (2021), 685. doi:https://doi.org/10.1016/j.healthpol.2021.03.007.

[41] Mefop. Guida per la best-practice dei fondi sanitari. (2021). arXiv:https://www.mefop.it/cms/doc/22165/linee-guida-fondi-sanitari-21-04-2021.pdf.

[42] Merlo, L., Maruotti, A., and Petrella, L. Two-part quantile regression models for semi-continuous longitudinal data: A finite mixture approach. *Statistical Modelling*, **0** (2021), 1471082X21993603. doi:10.1177/1471082X21993603.

[43] Neath, A. A. and Cavanaugh, J. E. The bayesian information criterion: background, derivation, and applications. *Wiley Interdisciplinary Reviews: Computational Statistics*, **4** (2012), 199.

[44] Neelon, B., O'Malley, A. J., and Smith, V. A. Modeling zero-modified count and semicontinuous data in health services research part 1: background and overview. *Statistics in Medicine*, **35** (2016), 5070. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.7050, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.7050, doi:https://doi.org/10.1002/sim.7050.

[45] Nelder, J. A. and Wedderburn, R. W. M. Generalized linear models. *Journal of the Royal Statistical Society. Series A (General)*, **135** (1972), 370. Available from: http://www.jstor.org/stable/2344614.

[46] Nigri, A., Levantesi, S., Marino, M., Scognamiglio, S., and Perla, F. A deep learning integrated lee–carter model. *Risks*, **7** (2019). Available from: `https://www.mdpi.com/2227-9091/7/1/33`, `doi:10.3390/risks7010033`.

[47] Ogunnaike, R. M. and Si, D. Prediction of insurance claim severity loss using regression models. In *Machine Learning and Data Mining in Pattern Recognition* (edited by P. Perner), pp. 233–247. Springer International Publishing, Cham (2017).

[48] Pavolini, E. and Seeleib-Kaiser, M. Comparing occupational welfare in europe: The case of occupational pensions. *Social Policy and Administration*, **52** (2018), 477.

[49] Pelessoni, R. and Picech, L. Some applications of unsupervised neural networks in rate making procedure. (1998).

[50] Petrella, L. and Raponi, V. Joint estimation of conditional quantiles in multivariate linear regression models with an application to financial distress. *Journal of Multivariate Analysis*, **173** (2019), 70. Available from: `https://ideas.repec.org/a/eee/jmvana/v173y2019icp70-84.html`, `doi:10.1016/j.jmva.2019.02.00`.

[51] Rajitha, C. and Sakthivel, K. Artificial intelligence for estimation of future claim frequency in non-life insurance. *Global Journal of Pure and Applied Sciences*, **13** (2017).

[52] Richman, R. Ai in actuarial science. *SSRN*.

[53] Schelldorfer, J. and Wüthrich, M. V. Nesting classical actuarial models into neural networks (2019).

[54] Sibuya, M., Yoshimura, I., and Shimizu, R. Negative multinomial distribution. *Annals of the Institute of Statistical Mathematics*, **16** (1964), 409.

[55] Spedicato, G., Dutang, C., and Petrini, L. Machine learning methods to perform pricing optimization. a comparison with standard glms. **12** (2018), 69.

[56] Speights, D. B., Brodsky, J. B., and Chudova, D. L. Using neural networks to predict claim duration in the presence of right censoring and covariates. In *Casualty Actuarial Society Forum*, pp. 255–278 (1999).

[57] Taylor, J. W. A quantile regression neural network approach to estimating the conditional density of multiperiod returns. *Journal of Forecasting*, (2000), 299.

[58] Taylor, J. W. Using Exponentially Weighted Quantile Regression to Estimate Value at Risk and Expected Shortfall. *Journal of Financial Econometrics*, **6** (2007), 382. Available from: `https://doi.org/10.1093/jjfinec/nbn007`, `arXiv:https://academic.oup.com/jfec/article-pdf/6/3/382/2471143/nbn007.pdf`, `doi:10.1093/jjfinec/nbn007`.

[59] Taylor, J. W. Forecasting value at risk and expected shortfall using a semiparametric approach based on the asymmetric laplace distribution. *Journal of Business & Economic Statistics*, **37** (2019), 121. Available from: `https://doi.org/10.1080/07350015.2017.1281815`, `arXiv:https://doi.org/10.1080/07350015.2017.1281815`, `doi:10.1080/07350015.2017.1281815`.

[60] Venables W.N., R. B. *Tree-based Methods.* Modern Applied Statistics with S-PLUS. Statistics and Computing. Springer (1999). Available from: `https://doi.org/10.1007/978-1-4757-3121-7_10`.

[61] White, H., Kim, T.-H., and Manganelli, S. Var for var: Measuring tail dependence using multivariate regression quantiles. *Journal of Econometrics*, **187** (2015), 169 . Available from: `http://www.sciencedirect.com/science/article/pii/S0304407615000287`, `doi:https://doi.org/10.1016/j.jeconom.2015.02.004`.

[62] Wüthrich, M. V. Machine learning in individual claims reserving. *Scandinavian Actuarial Journal*, **2018** (2018), 465. Available from: `https://doi.org/10.1080/03461238.2018.1428681`, `arXiv:https://doi.org/10.1080/03461238.2018.1428681`, `doi:10.1080/03461238.2018.1428681`.

[63] Wüthrich, M. V. Bias regularization in neural network models for general insurance pricing. *European Actuarial Journal*, **10** (2020), 179 . `doi:10.1007/s13385-019-00215-z`.

[64] Wüthrich, M. V. and Merz, M. Editorial: Yes, we cann! *ASTIN Bulletin*, **49** (2019), 1–3. `doi:10.1017/asb.2018.42`.

[65] Wuthrich, M. V. From generalized linear models to neural networks, and back. *SSRN*, (2019). `arXiv:https://ssrn.com/abstract=3491790`.

[66] Wuthrich, M. V. and Buser, C. *Data Analytics for Non-Life Insurance Pricing.* SSRN (2020). Available from: `https://ssrn.com/abstract=2870308`.

[67] Xu, Q., Deng, K., Jiang, C., Sun, F., and Huang, X. Composite quantile regression neural network with applications. *Expert Systems with Applications*, **76** (2017), 129 . Available from: `http://www.sciencedirect.com/science/article/pii/S0957417417300726`, `doi:https://doi.org/10.1016/j.eswa.2017.01.054`.

[68] Yunos, Z. M., Ali, A., Shamsyuddin, S. M., Ismail, N., et al. Predictive modelling for motor insurance claims using artificial neural networks. *Int. J. Advance Soft Compu. Appl*, **8** (2016).

[69] Zhang, Y., Zhou, H., Zhou, J., and Sun, W. Regression models for multivariate count data. *Journal of computational and graphical statistics : a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America*, **26** (2017), 1—13. `doi:10.1080/10618600.2016.1154063`.