Collective reserving using individual claims data

SCANDINAVIAN ACTUARIAL JOURNAL
2022, VOL. 2022, NO. 1, 1–28
https://doi.org/10.1080/03461238.2021.1921836

Taylor & Francis
Taylor & Francis Group

Collective reserving using individual claims data

Łukasz Delong[a], Mathias Lindholm[b] and Mario V. Wüthrich[c]

[a]Institute of Econometrics, SGH Warsaw School of Economics, Warsaw, Poland; [b]Department of Mathematics, Stockholm University, Stockholm, Sweden; [c]RiskLab, Department of Mathematics, ETH Zurich, Zurich, Switzerland

ABSTRACT

The aim of this paper is to operationalize claims reserving based on individual claims data. We design a modeling architecture that is based on six different neural networks. Each network is a separate module that serves a certain modeling purpose. We apply our architecture to individual claims data and predict their settlement processes on a monthly time grid. A proof of concept is provided by benchmarking the resulting claims reserves with the ones received from the classical chain-ladder method which uses much coarser (aggregated) data.

ARTICLE HISTORY
Received 22 April 2020
Accepted 21 April 2021

KEYWORDS
Claims reserving; individual claims data; micro-level reserving; neural networks; IBNR claims; RBNS claims; chain-ladder method; over-dispersed Poisson model

1. Introduction

The aim of this paper is to operationalize claims reserving based on individual claims data. Stochastic models for individual claims reserving were introduced by Arjas (1989) and Norberg (1993, 1999). They used the theory of marked point processes to describe individual claims, but they provided little guidance on the use of these models on real data. In recent years, more statistically oriented (semi-) parametric models have been studied by Larsen (2007), Taylor et al. (2008), Zhao et al. (2009), Jessen et al. (2011), Pigeon et al. (2013), Antonio & Plat (2014), Badescu et al. (2019), Maciak et al. (2019), Okine et al. (2019) and Wang et al. (2021). Still some of these proposals are not very operational, yet, because the assumptions made are too rigid from an applied perspective. Lately, machine learning methods find one's way into individual claims reserving, allowing for more flexible regression structures. Some recent papers are based on regression trees and gradient boosting, see Wüthrich (2018), Lopez et al. (2019), Lopez & Milhaud (2021), De Felice & Moriconi (2019) and Duval & Pigeon (2019); others are based on non-parametric and kernel methods, see Rosenlund (2012), Bischofberger et al. (2019), Baudry & Robert (2019), or on neural networks, see Gabrielli (2020), Kuo (2020) and Delong & Wüthrich (2020).

Surprisingly, many of these approaches pay little attention to the data itself, but they describe the methods used in much detail. This is surprising because the general aim here is to solve a data-driven real world problem and, thus, the data should play the most important and central role in this modeling problem. We take a different approach here, namely, we start from the data itself using exploratory data analysis to build an appropriate model around the data. In statistics, exploratory data analysis serves at formulating appropriate hypotheses, see Tukey (1977), which are then verified (or rejected) using statistical testing, which often results in conclusions about causal relations. In machine learning,

CONTACT Mario V. Wüthrich mario.wuethrich@math.ethz.ch

Supplemental data for this article can be accessed here. https://doi.org/10.1080/03461238.2021.1921836

one usually only infers correlations from the given data, which is sufficient for accurate predictions, see Breiman (2001). This seems questionable from a statistical point of view and a good compromise between these two modeling cultures and communities still needs to be found. We use exploratory tools to build reasonable models around the data, and we discuss the assumptions made.

This paper builds on our previous work (Delong & Wüthrich 2020) on individual claims reserving. There are two essential new parts here. First, we derive an estimate of claims reserves for both reported *and* incurred but not reported (IBNR) claims. The latter reserves are not considered in Delong & Wüthrich (2020), and we propose a conceptually new approach to deal with this part of the claims. The second contribution is to come up with a smaller architecture compared to Delong & Wüthrich (2020). Using this reduced architecture, we benefit from improved run times which is essential in practical applications. This improved run times allow us to consider finer time scales, and we verify that this smaller model provides accurate reserves by benchmarking it with the well-known chain-ladder (CL) method. Finer time scales are important for capturing seasonal patterns and, for instance, sensitivities in insurance product features such as waiting periods in accident and health insurance. The price we pay by this reduction is that, in contrast to Delong & Wüthrich (2020), we no longer have a full simulation model, but we only design an optimal architecture to receive accurate claims reserves.

1.1. Overview of our modeling architecture

In Section 2, we introduce the claims reserving problem and state the basic model assumptions. Our claims reserving architecture is based on six neural networks which can be viewed as separate modules each serving a specific modeling purpose; this architecture is illustrated in Figure 1. The first network is introduced in Section 3, and this network serves at predicting IBNR claim counts. In Section 4, we introduce the second network that serves at modeling the two-dimensional process of payment dates and closings of reported but not settled (RBNS) claims. The third and fourth networks in Section 5 model claim payments as well as recovery payments of RBNS claims: the third network indicates whether we have a positive payment or a negative recovery payment, and the fourth network determines its size. In Section 6, we introduce the last two networks that serve at modeling amounts of IBNR claims: these claim amounts are modeled by a compound distribution with the fifth network indicating whether we have a zero IBNR claim or not, and the sixth network determining the amount of a non-zero IBNR claim. Our modeling is done on two different time scales; we are using a weekly time grid for IBNR claims and a monthly time grid for RBNS claims. These choices are mainly determined by the reporting frequency of our data but also by computational and statistical constraints.

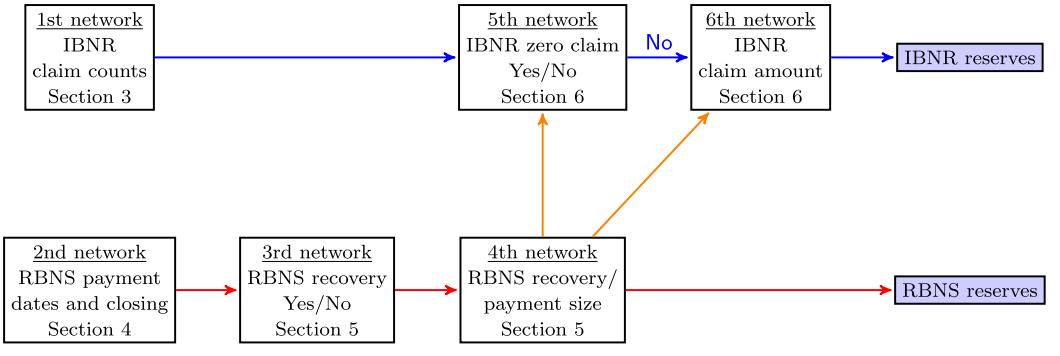

Figure 1. Claims reserving architecture used consisting of six networks.

2. Notation and basic model assumptions

In this section, we introduce the basic model assumptions. We assume that the reader is already familiar with the claims reserving problem. For a general introduction to claims reserving, we refer to Chapter 1 in Wüthrich & Merz (2008) and, in particular, to Figure 1 which illustrates a typical general insurance claim. A general insurance claim is described by an *accident date* T, a *reporting date* U, a *closing date* V and *claims payments* $(Y_s)_s$ made between reporting U and closing V.

2.1. Individual claims and basic model assumptions

To embed our individual claims reserving model into the bigger context of claims reserving, we start by describing a variant of the collective reserving model (CRM) of Wahl et al. (2019) and Lindholm & Verrall (2020). The CRM can be seen as a discrete time and aggregated version of the individual claims model of Norberg (1993, 1999). The original CRM is defined on a yearly time grid. For the moment, we consider a daily time grid; this is usually the smallest time scale available in real insurance data. An individual claim is described by a vector $\mathcal{C} = (T, U, W, (Y_s)_{s\geq 0}, (O_s)_{s\geq 0}; \vartheta)$:

Variable	
T	Accident date (in daily units as yyyy/mm/dd)
U	Reporting date (in daily units as yyyy/mm/dd)
$W = U - T \geq 0$	Non-negative reporting delay (in daily units)
$s \geq 0$	Daily time index, initialized by $s = 0$ being the reporting date $U = T + W$
$(Y_s)_{s\geq 0}$	Claim payment process with payments Y_s at times $U + s$ (in daily time units)
$\mathcal{Y}_s = \mathbb{1}_{\{Y_s \neq 0\}}$	Indicator for claim payments at times $U + s$ (in daily time units)
$(O_s)_{s\geq 0}$	$\{1, 0\}$-valued (daily) claim status process for open/closed claim at times $U + s$
ϑ	Static claim features such as line of business (LoB) and claim type

A claim $\mathcal{C}$ corresponds to the accident date and mark process described in (2.3) of Larsen (2007). Roughly speaking, this process consists of two components: (1) a claim occurrence process describing the accident date T of a claim and (2) a (mark) process describing the remaining claim settlement components $(U, W, (Y_s)_{s\geq 0}, (O_s)_{s\geq 0}; \vartheta)$ of that given claim, starting at the accident date T. The lower index s measures time in daily units elapsed since the reporting date U of a given claim and, thus, it has the interpretation of current settlement delay.

In our data, feature information $\vartheta =$ (LoB, claim code) is static consisting of the two components 'line of business' (LoB) and 'claim code'. Static information substantially simplifies modeling because it remains unchanged over the whole life time of the claim. Dynamic information like number of injured people, which can change over time, would require more complex dynamic modeling.

Assumptions 2.1: Starting from a sufficiently rich probability space $(\Omega, \mathcal{F}, \mathbb{P})$, we assume that all considered claims $\mathcal{C}$ live on $(\Omega, \mathcal{F}, \mathbb{P})$ and are i.i.d. w.r.t. $\mathbb{P}$. Moreover, we assume that there exists a maximal reporting delay $m > 0$ such that all claims $\mathcal{C}$ satisfy $0 \leq W < m$, $\mathbb{P}$-a.s.

The i.i.d. property in Assumptions 2.1 describes the random selection of individual claims $\mathcal{C} = (T, U, W, (Y_s)_{s\geq 0}, (O_s)_{s\geq 0}; \vartheta)$ from the pool of all potential claims. This assumption is not as restrictive as it might seem at first sight, e.g. it allows us to model seasonal patterns which can be described by the two-step approach of Larsen (2007): (1) select accident dates T from a distribution that puts more probability weight, say, to winter claims than to summer claims and (2) a mark process $(U, W, (Y_s)_{s\geq 0}, (O_s)_{s\geq 0}; \vartheta)$ which is defined conditional on T.

2.2. Time scales

We have introduced claims $\mathcal{C} = (T, U, W, (Y_s)_{s\geq 0}, (O_s)_{s\geq 0}; \vartheta)$ on a daily time scale. Usually accident dates T and reporting dates U are available in daily units. However, other variables like payments

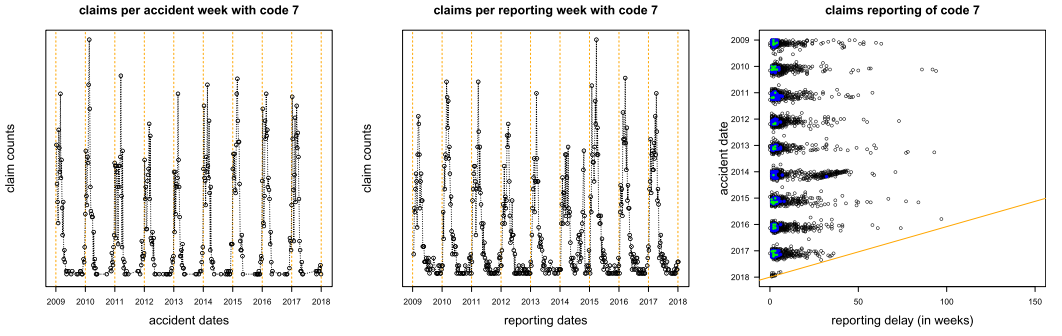

Figure 2. Observed claim counts per accident week (lhs), per reporting week (middle) and scatter plot of reported trapezoid showing accident dates vs. reporting delays (rhs) for claim code 7.

$(Y_s)_{s\geq 0}$ and the claim status process $(O_s)_{s\geq 0}$ may only be available on a coarser time scale due to data warehousing and business process constraints. We are going to use problem and data adapted time scales in this study. We choose a *weekly* time scale to study accident dates T and reporting dates U. The choice of a weekly time scale is justified by the fact that claim occurrences often have a weekly pattern, because weekends may behave differently from working days, see Figure 5 in Verrall & Wüthrich (2016) and Figure 2 in Crevecoeur et al. (2019). By selecting a weekly time scale, we capture an entire weekly cycle in each period. This will be discussed in Sections 3 and 6, where we predict IBNR claims.

For the claim payment and claim status processes $(Y_k)_{k\geq 0}$ and $(O_k)_{k\geq 0}$, we move to a *monthly* time scale because data are only available on this time scale (end-of-month data). To highlight this change in time units, we use the notation k for time (instead of s), thus, $k \geq 0$ will denote months elapsed since reporting for payment process and claim status modeling. This will be discussed in Sections 4 and 5, where we treat RBNS claims.

Whenever we discuss aggregated methods like the CRM or the CL model, we use an *annual* time grid, as it is typically done in industry. Aggregated methods cannot capture individual claim features which especially applies to seasonal factors. Therefore, aggregated methods typically act on a yearly time grid to capture an entire annual cycle.

2.3. The claims reserving problem

The typical situation in industry is to have a cutoff date τ which reflects today's date. Throughout, we assume that the cutoff date is set to an end of a calendar year, i.e. 31st of December.[1] All variables that occur before τ are observable at this cutoff date, and the main task in claims reserving is to predict all variables that occur after τ. We consider the run-off situation meaning that the exposures (for claim occurrence) are capped at τ, thus, all accident dates must fulfill $T \leq \tau$. The claims with reporting dates $U = T + W \leq \tau$ are reported at time τ and the claims with reporting dates $U = T + W > \tau \geq T$ are IBNR claims. The reported claims are further split in claims that are *closed* at time τ and claims that are RBNS at time τ. This is characterized by the claim status variable $O_{\tau-U} \in \{0, 1\}$ for closed and RBNS claims at time τ, respectively. The claims reserving exercise aims at predicting all payments Y_s after time τ, i.e. $s > \tau - U$, for claims with accident dates $T \leq \tau$. This comprises IBNR and RBNS claims, but it also contains closed claims at time τ which may be re-opened at a later time point (due to unexpected relapses).

[1] Setting the cutoff date to a calendar year end makes some notation simpler, however, this choice is w.l.o.g.

2.4. Claims aggregation

For receiving the classical claims reserving models, we need to aggregate individual claims $(\mathcal{C}_\ell)_\ell$ appropriately, where $\ell \geq 1$ labels all claims that have occurred before the cutoff date, i.e. with $T_\ell \leq \tau$. This claims aggregation is best described with the help of the CRM and the CL model. We choose a fixed time scale (e.g. years, months or weeks) and aggregating all claims within the corresponding time intervals (called time *periods*, here). To transform time from daily units s to a coarser unit, we use the *time ceiling function* $\lceil \cdot \rceil$ which maps daily time s to the corresponding coarser unit denoted by $\lceil s \rceil$ by rounding up, say, to monthly units. The explicit choice of the time ceiling function $s \mapsto \lceil s \rceil$ depends on the chosen time scale (which is always highlighted in the surrounding text). For a more formal definition, we refer to (A.1) in our supplementary material. We select as fixed time periods either years, months or weeks. For occurrence periods $0 \leq i \leq \lceil \tau \rceil$ and reporting delays $j \geq 0$, we define the *number of reported claims* in accident period i with reporting delay j by

$$N_{i,j} = \sum_\ell \mathbb{1}_{\{\lceil T_\ell \rceil = i,\ \lceil U_\ell \rceil - \lceil T_\ell \rceil = j\}}. \tag{1}$$

In Wahl et al. (2019), this quantity is studied on an annual scale and it is assumed to be over-dispersed Poisson (ODP) distributed; we come back to this assumption in Section 3.2. The reported claims at the cutoff date τ give the so-called *upper triangle* generating information

$$\mathcal{N}_\tau = \sigma\left\{N_{i,j};\ 0 \leq i \leq \lceil \tau \rceil,\ j \geq 0,\ i + j \leq \lceil \tau \rceil\right\}. \tag{2}$$

The classical CL prediction for claim counts is based on this information $\mathcal{N}_\tau$. Next, we define the total payments of a given accident period i, reporting delay j and settlement delay $k \geq 0$

$$X_{i,j,k} = \sum_\ell \left(\mathbb{1}_{\{\lceil T_\ell \rceil = i,\ \lceil U_\ell \rceil - \lceil T_\ell \rceil = j\}} \sum_{s \geq 0} Y_{\ell,s} \mathbb{1}_{\{\lceil U_\ell + s \rceil - \lceil U_\ell \rceil = k\}} \right).$$

The payments for accident period i and payment delay j are given by

$$X_{i,j} = \sum_{0 \leq k \leq j} X_{i,j-k,k}.$$

This provides aggregate information at the cutoff date τ

$$\mathcal{X}_\tau = \sigma\left\{X_{i,j};\ 0 \leq i \leq \lceil \tau \rceil,\ j \geq 0,\ i + j \leq \lceil \tau \rceil\right\} \quad \text{and} \quad \mathcal{F}_\tau = \mathcal{N}_\tau \vee \mathcal{X}_\tau.$$

The classical CL method for (cumulative) payments, as well as many other claims reserving methods, use information $\mathcal{X}_\tau$ on an annual scale. Information $\mathcal{F}_\tau$ on an annual scale is used, for instance, in the double CL model of Martínez-Miranda et al. (2012a, 2012b).

3. IBNR claim counts modeling

We start by discussing the modeling of claim counts $N_{i,j}$ which involves the modeling of accident dates T_ℓ, reporting dates U_ℓ and static claim features ϑ_ℓ of individual claims $\mathcal{C}_\ell$, $\ell \geq 1$, the lower index ℓ always indicates that we consider a specific individual claim.

3.1. Descriptive data analysis of claim counts

The available data come from two different LoBs[2] with nine different claim codes, thus, $\vartheta \in \Theta = \{1, 2\} \times \{1, \ldots, 9\}$. Each claim $\mathcal{C}_\ell$ is established with an accident date T_ℓ, a reporting date U_ℓ and a

[2] For anonymization reasons we only give aggregated data in plots (over both LoBs), and all claim payments are scaled with a constant. Moreover, some figures do not show the scale on the y-axis, see, e.g. Figure 2 (lhs).

static claim feature $\vartheta_\ell \in \Theta$. Moreover, for each reported claim $\mathcal{C}_\ell$, we know the settlement processes of payments $(Y_{\ell,k})_k$ and of the claim status $(O_{\ell,k})_k$ up to the cutoff date τ.

At the cutoff date $\tau = 2017/12/31$, we have observed $n = 34{,}260$ reported claims with accident dates $\tau_0 = 2009/01/01 \le T_\ell \le U_\ell \le \tau$ and static features $\vartheta_\ell \in \Theta$. These data are sufficient to model claim counts, their reporting delays $W_\ell = U_\ell - T_\ell$ and their static claim features ϑ_ℓ. The latter are assumed to be fully known at the reporting date U_ℓ (therefore called static).

We analyze this reporting data for different claim codes. Figure 2 (lhs) illustrates the claim counts of claim code 7 aggregated over the accident dates T_ℓ on a weekly time scale (and reported up to the cutoff date τ). In this graph, IBNR claims (with $U_\ell > \tau$) are missing, therefore the values in Figure 2 (lhs) may still increase, in particular, close to the cutoff date. The crucial observation is that we have a strong seasonal pattern for claims with code 7 (with mid year lows). Figure 2 (middle) shows exactly the same claims, but the x-axis has been changed to reporting dates U_ℓ on a weekly time scale. In general, the claims move slightly to the right compared to the accident date plot because of positive reporting delays $W_\ell = U_\ell - T_\ell \ge 0$. However, the main structure of the seasonal pattern remains. Finally, Figure 2 (rhs) gives the resulting scatter plot with accident dates T_ℓ on the y-axis and reporting delays W_ℓ on the x-axis (in weekly units). The orange (diagonal) line shows the cutoff date τ, which separates the so-called upper trapezoid (observations) from the lower triangle (to be predicted). We observe the same seasonal pattern in the scatter plot as in the other two plots, and we also observe that all reported claims have a reporting delay W_ℓ of less than 100 weeks. Figure 1 in the supplementary material shows the corresponding scatter plots of all other claim codes. Some of them have a stronger seasonal pattern, but none of them shows reporting delays bigger than 100 weeks. This justifies setting a maximal reporting delay m in Assumptions 2.1.

3.2. Predicting the number of IBNR claims

Our first goal is to predict the number of claims which will be reported after the cutoff date τ, that is, satisfying $\tau_0 \le T_\ell \le \tau < U_\ell$. We choose a weekly time grid for this prediction problem, with $0 \le i \le I$ denoting accident weeks and with $0 \le j \le J$ denoting reporting delay weeks; we also refer to Section 2.2. The reason for choosing a weekly time grid is that this provides a good balance between computational efficiency and (seasonal) accuracy. Choosing a finer grid size is computationally more demanding without much benefits in better accuracy for IBNR claim counts prediction. In particular, on a daily time grid, one would have to bother about weekday differences in claim frequencies which can be very cumbersome, see Verrall & Wüthrich (2016). Going over to a monthly time grid is not advisable at this stage because claim occurrences are not sufficiently homogeneous across an entire calendar month; note that there are sharp seasonal increases and decreases in claim counts, see Figure 2, which may lead to too much heterogeneity on a monthly time grid. Moreover, a monthly time grid is not equidistant because different calendar months have different numbers of days. Equidistance is not crucial, but it is still a nice property to have since it implies equal time exposures for all considered periods.

In our data, we have $I = 470$ accident weeks from $\tau_0 = 2009/01/01$ to $\tau = 2017/12/31$, and the maximal reporting delay is $J = 100$ weeks, the latter is justified by Figure 1 in the supplementary material. In the remainder of this section, we choose the time ceiling function $\lceil \cdot \rceil$ on the weekly scale. In view of (1), this allows us to define the claim counts on that weekly time scale for fixed static feature $\vartheta = (\text{LoB}, \text{claimcode}) \in \Theta$

$$N_{i,j|\vartheta} = \sum_\ell \mathbb{1}_{\{\lceil T_\ell \rceil = i,\ \lceil U_\ell \rceil - \lceil T_\ell \rceil = j\}} \mathbb{1}_{\{\vartheta_\ell = \vartheta\}}.$$

The resulting observations at the cutoff date τ are denoted by $\mathcal{N}_{\tau|\vartheta}$, see also (2). The upper trapezoid in Figure 2 (rhs) illustrates the observations at the cutoff date τ (above the orange line) for claim code 7 (aggregated over both LoBs).

Assumptions 3.1: The claim arrival processes $(T_\ell^\vartheta, W_\ell^\vartheta)_{\ell\geq 1}$ for static features $\vartheta \in \Theta$ are described by independent (in ϑ) two-dimensional inhomogeneous Poisson point processes on $[\tau_0, \tau] \times [0, 840]$ (in daily units).

We describe the two-dimensional claim arrival processes by independent (for different ϑ's) inhomogeneous Poisson point processes with accident dates $T_\ell^\vartheta \in [\tau_0, \tau]$ and reporting delays $W_\ell^\vartheta \in [0, 840]$, i.e. with a maximal reporting delay of 120 weeks ($m = 7 \cdot 120 = 840$ days). We have slightly changed the notation here from (T_ℓ, W_ℓ) to $(T_\ell^\vartheta, W_\ell^\vartheta)$ indicating that the latter are accident dates and reporting delays of claims with a given static feature $\vartheta \in \Theta$.

We assume that the inhomogeneous Poisson point processes of Assumptions 3.1 can be described by a (measurable and bounded) intensity function

$$\lambda : [\tau_0, \tau] \times [0, 840] \times [0, 1] \times \Theta \to \mathbb{R}_+, \quad (t, w, v(t), \vartheta) \mapsto \lambda(t, w, v(t), \vartheta) > 0, \tag{3}$$

where $t \mapsto v = v(t) \in [0, 1]$ models the occurrence week *within* a calendar year using a linear function with initialization $v(\text{calendar week } 1) = 0$ and $v(\text{calendar week } 53) = 1$. This variable $v(t) \in [0, 1]$ will allow us to model annual seasonal patterns which are motivated by Figure 2.

Remark 3.2:

- We could drop component $v(t)$ in (3) because it is fully encoded in the accident date component t. We prefer the approach above because it explicitly highlights that we model an annual seasonal feature, and we use it as an input variable to the network regression model used below to give a hint to the network to search for an annual seasonal pattern.
- At first sight, it would be more natural to model the annual seasonal pattern by a sine–cosine function $t \mapsto (\sin(\pi v(t)), \cos(\pi v(t)))$. For our data, it seems to work better to choose a linear input $t \mapsto v(t)$ and let the network find the seasonal functional form (depending on ϑ). By our intuition, it may be because the seasonal pattern is non-symmetric around the peaks and/or because the seasonal peak times differ for different $\vartheta \in \Theta$.

Assumptions 3.1 imply that $(N_{i,j|\vartheta})_{i,j,\vartheta}$ are independent and Poisson distributed with expected numbers of claims

$$\Lambda(i, j, v(i), \vartheta) \overset{\text{def.}}{=} \mathbb{E}\left[N_{i,j|\vartheta}\right] = \int_{\{\lceil t\rceil = i, \lceil t+w\rceil - \lceil t\rceil = j\}} \lambda(t, w, v(t), \vartheta)\, \mathrm{d}t\, \mathrm{d}w. \tag{4}$$

As emphasized in Remarks 3.2, it would not be necessary to include $v(i)$ in $\Lambda(i, j, v(i), \vartheta)$ because the variable i fully encodes the calendar week. However, we estimate regression function (4) with a first neural network, and this network will find seasonal structure more easily if we explicitly include a seasonal feature $v(i)$. We model the following regression function by a neural network

$$(i, j, v(i), \vartheta) \;\mapsto\; \Lambda(i, j, v(i), \vartheta) \tag{5}$$

having depth 3 and with $(q_1, q_2, q_3) = (20, 15, 10)$ hidden neurons in these three hidden layers. The explicit modeling choices are provided in Section B of the supplementary material; this supplementary material also provides the corresponding R code in Listing 1, for information to neural network regression modeling we also refer to Wüthrich & Merz (2021). Moreover, the choices for gradient descent fitting are described in this supplementary material. As objective function for this model fitting, we choose the Poisson deviance loss given by

$$\mathcal{L} = \sum_\vartheta \sum_{i+j\leq \lceil \tau\rceil} N_{i,j|\vartheta} \left[\frac{\Lambda(i, j, v(i), \vartheta)}{N_{i,j|\vartheta}} - 1 - \log\left(\frac{\Lambda(i, j, v(i), \vartheta)}{N_{i,j|\vartheta}}\right)\right].$$

This fitting provides us with the first estimated network function $(i, j, v(i), \vartheta) \mapsto \widehat{\Lambda}(i, j, v(i), \vartheta)$.

Table 1. Claim counts: run time of calibration, number of model parameters, estimated number of claims on the weekly time grid in the upper triangle and predicted number of IBNR claims.

	Run time	No. of parameters	Average per (i,j)	Number of IBNR claims
Neural network	427 s	644	7.7277%	402
ODP CL method	–	11	7.7277%	458

Remark 3.3:

- If we assumed a cross-classified structure $\Lambda(i,j,v(i),\vartheta) = \alpha_i\beta_j$, we would obtain the Poisson version of the CL method. The latter is solved within generalized linear models (GLMs). This GLM may miss important interactions and seasonal patterns (because we only have multiplicative interactions in the cross-classified case). For this reason, we prefer the neural network approach. Alternatively, one could try a more complex GLM or a generalized additive model (GAM), but such approaches require that one explicitly models interactions, whereas a neural network tries to find such interactions itself.
- Similarly to the CL method we do not include any exposure measure. We do this because of lack of information. If we would have exposure information, it could be included in the feature information, and it would lead to a Bornhuetter–Ferguson type of model.

The network regression function $(i,j,v(i),\vartheta) \mapsto \widehat{\Lambda}(i,j,v(i),\vartheta)$ is fitted on the upper triangle $i+j \le \lceil\tau\rceil$, and its extrapolation to the lower triangle $i+j > \lceil\tau\rceil$ gives us the predicted numbers $\widehat{\Lambda}(i,j,v(i),\vartheta)$ of IBNR claims. In the upper triangle we have on average 0.077277 claims per accident week i and reporting delay week j with $i+j \le \lceil\tau\rceil$, see Table 1.[3] The extrapolation to the lower triangle gives us in total 402 predicted IBNR claims.

Figure 3 illustrates the results for claim code 7. The left-hand side and the middle plots show the estimated marginal claim counts for accident weeks and reporting weeks, the red dashed lines exclude IBNR claims (estimated model in upper triangle) and the cyan and green lines include IBNR claims (prediction in lower triangle). This is clearly visible at the right end of the plots by a steeply increasing cyan curve for accident dates, and by the light-green extrapolation for reporting dates. Focusing on Figure 3 (lhs), we compare the observations (black circles) to the neural network estimated means (red dashed line). Around the turn of the year, we observe a few outliers that do not seem to be explained by the estimated means. We confirm that, indeed, the black dots fluctuate around the red dashed line with an empirical mean being equal to the estimated mean. It is the Poisson assumption that may need further reflection because the outliers (which occur in different calendar weeks) are a sign of over-dispersion. This could be a motivation to study Cox processes, see Badescu et al. (2019); here we refrain from doing so because IBNR claims play a marginal role in our reserves calculations (as we will see below).

Using this estimated model, we run one simulation for the lower triangle (for given $\vartheta \in \Theta$) resulting in the red dots (claims) of Figure 3 (rhs). This extrapolates the structure in the upper triangle to the lower one. The cyan line in Figure 3 (rhs) gives the estimated average reporting delay per accident week. We observe a strong seasonal pattern in these average reporting delays; however, this pattern has to be considered with care because there are almost no claims, here, during European summer months. These estimated average reporting delays can be interpreted in a similar way as the operational times considered in Bischofberger (2019), see Figure 1 of Bischofberger (2019). In the supplementary material, we give the resulting scatter plots for all claim codes, see corresponding Figure 1. We note that most claim codes considered have a seasonal pattern.

Aggregating the predicted numbers $\widehat{\Lambda}(i,j,v(i),\vartheta)$ over the lower triangle $i+j > \lceil\tau\rceil$ provides a prediction of 402 IBNR claims in the lower triangle. We compare this prediction to the one obtained

[3] Run times are measured on a personal laptop with CPU at 2.50 GHz (4 CPUs) with 16GB RAM (which is a comparably modest computational power).

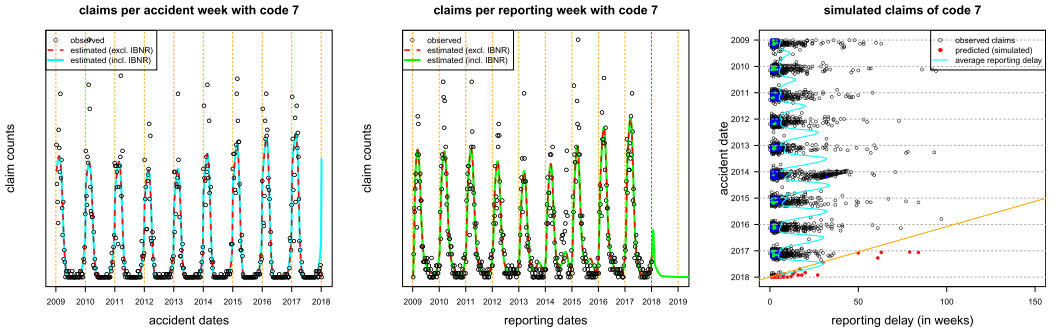

Figure 3. Observed and predicted claims per accident week (lhs), per reporting week (middle) and reported (black) and predicted (red) triangles (rhs) for claim code 7.

by the CL method based on yearly observations (2). From the CL method, we receive 458 IBNR claims which is bigger, see Table 1. Performing the ODP bootstrap described in England & Verrall (1999), we receive a rooted mean square error of prediction (RMSEP) of 49 (this is based on an estimated over-dispersion of 3.89). Thus, the neural network and the CL predictions differ by roughly 1.1 RMSEPs which almost explains the differences between the two methods.

Remark 3.4: Typically, in claims reserving, the lower triangle is unknown; therefore, we cannot perform a proper out-of-sample test. Often, new methods are benchmarked with known claims reserving methods that have proved to be credible on certain types of insurance business. If the resulting two predictions are too far apart, then obviously there is a problem. Otherwise, there is support that the established and the new methods judge information and relevant structure in a similar way. In our analysis, we benchmark each modeling step with the CL method, because it has been applied successfully to this data set in the past. The benefit of our refined method is that it may support finding weaknesses in the CL prediction, and it allows for a more granular study determining seasonal patterns, etc. Note, we use different time scales in the two models.

4. Payment indicator and claim status modeling of RBNS claims

In Sections 4 and 5, we model the settlement processes of reported claims. This includes the payment indicator and claim status process $(\mathcal{Y}_s, O_s)_{s\geq 0}$ of each individual claim (Section 4) as well as the payment process $(Y_s)_{s\geq 0}$ in case of payments (Section 5). The data for this modeling part are always available at the end of a calendar month (end-of-month data); thus, the granularity of these observed settlement processes is more coarse than the claim occurrence data. For this reason, we move to a monthly grid size in this and the next section. To highlight this we change the time notation from the daily $s \geq 0$ to a monthly $k \in \mathbb{N}_0$, see Section 2.2. The time ceiling function $\lceil\cdot\rceil$ is understood on the monthly time scale in this and the next section. The claim status process $(O_k)_{k\in\mathbb{N}_0}$ always denotes end-of-month statuses, and the monthly payments and (monthly) payment indicators are for $k \in \mathbb{N}_0$ defined as follows:

$$Y_k = \sum_{s\geq 0} Y_s \mathbb{1}_{\{\lceil U+s\rceil - \lceil U\rceil = k\}} \quad \text{and} \quad \mathcal{Y}_k = \mathbb{1}_{\{Y_k \neq 0\}}. \tag{6}$$

This is just a complicated way of saying that Y_k are the total individual claim payments within the kth settlement month for a given claim, and k is called settlement delay (after reporting).

4.1. Description of our RBNS claims modeling approach

Kuo (2020) performs a simultaneous multi-period forecast after the cutoff date τ for the entire run-off $(Y_k)_{k>\lceil\tau\rceil-\lceil U\rceil}$ of individual reported claims. This multi-period forecast is based on a long short-term memory network model. On a yearly time grid, this is computationally feasible on our finer

monthly time grid and we prefer a model that performs successive one-period-ahead forecasts. This allows us to keep tractability, interpretability as well as the possibility to intervene the process during different modeling steps of the settlement process. The price we pay by this reduction is that we do no longer receive a full simulation model, but we aim at direct accurate claims reserves estimation. Successive one-period-ahead forecasts means that we recursively extrapolate the settlement process into the lower triangle. This seems simple, but it requires special attention. The observations $(Y_k, \mathcal{Y}_k, O_k)_{k\leq \lceil \tau \rceil - \lceil U \rceil}$ in the *upper triangle* are on a random variable level, and predictions in the *lower triangle* typically are on the level of estimated means $(\widehat{Y}_k, \widehat{\mathcal{Y}}_k, \widehat{O}_k) = (\widehat{\mathbb{E}}[Y_k], \widehat{\mathbb{E}}[\mathcal{Y}_k], \widehat{\mathbb{E}}[O_k])$, for $k > \lceil \tau \rceil - \lceil U \rceil$. Thus, the network outputs an expected value while the input of the recursion is defined in terms of random variables. By using the expected values as inputs in the recursion, the inputs are smoothed, and the output may not be appropriate (because the networks have not been trained on smoothed inputs). This problem can be overcome if we have a reasonably good stochastic model from which we can Monte Carlo simulate observations that are of the same quality as the observations in the upper triangle. These simulated observations can then be used as inputs for the next step of the one-period-ahead forecasts, this is the approach taken in Delong & Wüthrich (2020). Most papers in the literature do not make this clear distinction and corresponding results should be considered with care, Gabrielli (2020) being an exception making a proper distinction between random variables and expected values in his architecture.

Monte Carlo extrapolation of the indicator process $(\mathcal{Y}_k, O_k)_{k\in\mathbb{N}_0}$ is not difficult because it involves a stochastic process on a (finite) categorical state space. Monte Carlo extrapolation of the claim payment process $(Y_k)_{k\in\mathbb{N}_0}$ poses a challenge because there is no simple off-the-shelf model that is sufficiently accurate in simulating claim payments, since real payments are often of a rather complex nature. For this reason, we divide this modeling step into two parts: the first part, described here in Section 4, concerns modeling of the indicator process $(\mathcal{Y}_k, O_k)_{k\in\mathbb{N}_0}$. This modeling part will *not* rely on any claim payment size information $(Y_k)_{k\in\mathbb{N}_0}$ and can, therefore, easily be extrapolated into the lower triangle by Monte Carlo simulation. The second part, described in Section 5, concerns modeling of the claim payment process $(Y_k)_{k\in\mathbb{N}_0}$, conditionally on knowing the indicator process. Since claims reserves are *expected values*, we are mainly interested in appropriate *expected claim payments*. Therefore, this modeling step is designed to avoid simulation, and we try to directly assess expected future payments as accurately as possible. This approach has the advantage that we only need to simulate random variables for which we have an accurate distributional description, but this approach has also its limitations as discussed in Limitations 5.1.

4.2. Stylized facts about claim payments and closings

We define the pattern of the observed relative number of payments per claim on a monthly scale

$$\widehat{\pi}_k = \frac{\sum_{\ell=1}^n \mathcal{Y}_{\ell,k} \mathbb{1}_{\{\lceil U_\ell \rceil + k \leq \lceil \tau \rceil\}}}{\sum_{\ell=1}^n \mathbb{1}_{\{\lceil U_\ell \rceil + k \leq \lceil \tau \rceil\}}} \in [0,1], \quad \text{for } k \geq 0,$$

where $\mathcal{Y}_{\ell,k} \in \{0,1\}$ is the monthly payment indicator of claim ℓ in settlement month k, see (6). Figure 4 gives the cumulative empirical payment pattern $(\widehat{\Pi}_k = \sum_{j=0}^k \widehat{\pi}_j)_{k\geq 0}$ for the different claim codes 1–9. We observe that claim codes 3, 4, 6 and 9 have on average less than one monthly payment, and claim codes 1, 2, 5, 7 and 8 have on average significantly more than one monthly payment, claim code 7 having on average close to 4 monthly payments per claim.

Next, we plot the empirical ratio of open claims for settlement month $k \geq 0$ given by

$$\widehat{\varpi}_k = \frac{\sum_{\ell=1}^n O_{\ell,k} \mathbb{1}_{\{\lceil U_\ell \rceil + k \leq \lceil \tau \rceil\}}}{\sum_{\ell=1}^n \mathbb{1}_{\{\lceil U_\ell \rceil + k \leq \lceil \tau \rceil\}}} \in [0,1].$$

We observe a fast settlement of the claims with claim codes 1, 3, 4, 6, 8 and 9, but we still expect open claims for claim codes 2, 5, 7 after 10 years of settlement delay, see Figure 4 (rhs). Henceforth, the settlement process is expected to be long-tailed for some claim types.

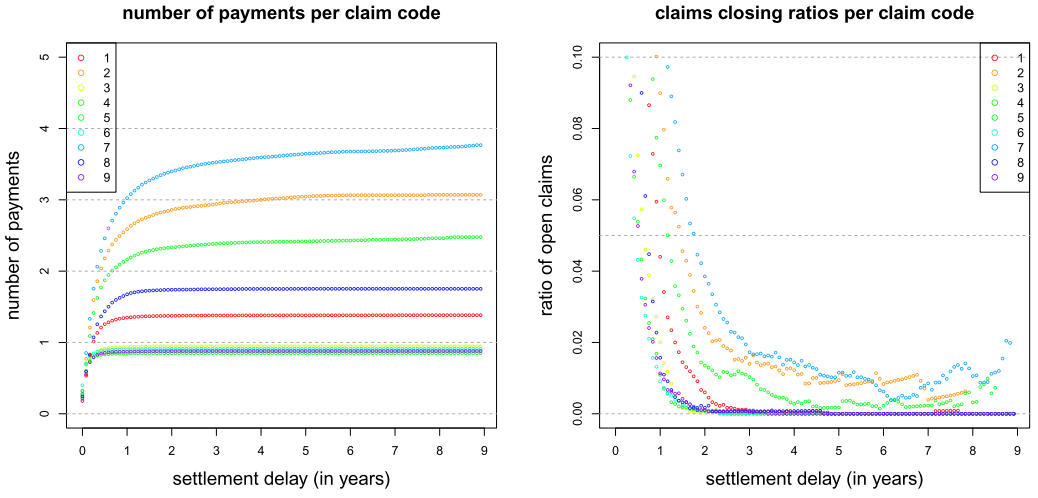

Figure 4. (lhs) Cumulative empirical payment pattern $(\widehat{\Pi}_k)_{k\geq 0}$ and (rhs) claim status ratios $(\widehat{\varpi}_k)_{k\geq 0}$ for the different claim codes.

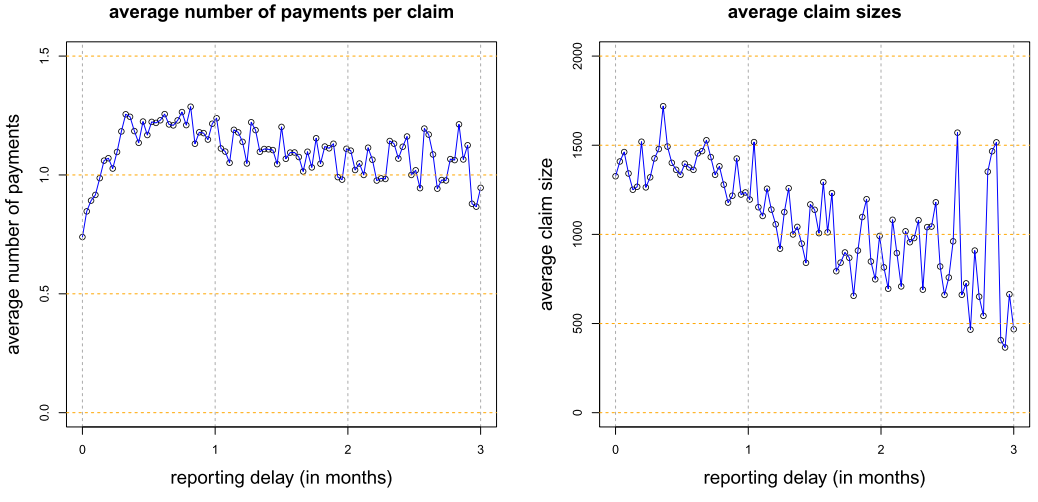

Figure 5. (lhs) Observed average number of payments per claim as a function of reporting delay (in months) and (rhs) average claim sizes as a function of reporting delay over all claim codes.

In Figure 5, we plot the observed average number of monthly payments per claim and the average claim sizes as a function of reporting delay W; we calculate these numbers on a fully developed rectangle to receive consistent results. Therefore, the plots in Figure 5 do not reflect settled claims because they may still develop beyond these rectangles. However, they illustrate that the payments depend on reporting delays, which justifies the inclusion of the reporting delay covariate W for payment prediction.

4.3. Indicator process modeling

4.3.1. Neural network model

First, we modify our two-dimensional Bernoulli process to a one-dimensional categorical process by defining for $k \geq 1$ the response variables (on the monthly scale)

$$\mathcal{I}_k = \mathcal{Y}_k + 2 \cdot O_k \in \{0, 1, 2, 3\}.$$

We model response variable $\mathcal{I}_k$ with a categorical distribution having probability vector

$$\boldsymbol{x}_k \;\mapsto\; \boldsymbol{p}(\boldsymbol{x}_k) = \big(p_0(\boldsymbol{x}_k), p_1(\boldsymbol{x}_k), p_2(\boldsymbol{x}_k), p_3(\boldsymbol{x}_k)\big)^\top \in \mathbb{R}_+^4 \tag{7}$$

for given feature information $\boldsymbol{x}_k$. We first describe the choice of the feature information $\boldsymbol{x}_k$. Similarly to (4), it contains the reporting delay $W = U - T \geq 0$, the week within the calendar year $\nu(T) \in [0, 1]$ of claim occurrence (for seasonal pattern modeling) and the static feature information $\vartheta =$ (LoB, claimcode) $\in \Theta$. We do not include the accident date T information for stability reasons. It has turned out in experiments that the inclusion of T may lead to an undesired extrapolation of trends, which might be problematic if it affects the extrapolation of the upper right triangle (over a long time period).[4]

Additionally, we are going to include past trajectory $(\mathcal{Y}_l, O_l)_{0\leq l\leq k-1}$ information in the feature. Since claim re-openings are scarce (this is verified in Figure 7), we are only going to include the last claim status O_{k-1}; this does not cause a major loss of information and it reduces the dimension of the problem substantially; we emphasize that re-openings are still modeled but we make a Markov assumption in this variable. Next, we include one year of past monthly payment indicator observations $(\mathcal{Y}_l)_{k-12\leq l\leq k-1}$; again to reduce the dimension of the problem we do not provide the whole past payment indicator process. To still keep track of multiple monthly payments over the entire past we add a variable $\mathfrak{y}_{k-1} = \sum_{l=0}^{k-1} \mathcal{Y}_l$. Finally, we would like to use regression function (7) simultaneously for all settlement periods $k \geq 1$. Therefore, we need to specify the undefined part $(\mathcal{Y}_l)_{l<0}$ in $(\mathcal{Y}_l)_{k-12\leq l\leq k-1}$ for small settlement periods k. We do this by setting the undefined values $(\mathcal{Y}_l)_{l<0}$ identically equal to -1 (we call this masking). Moreover, we add k to the feature information $\boldsymbol{x}_k$ to highlight to the network which settlement month we are considering (this is beneficial for model calibration because we guide the network in the right direction for finding model structure). Summarizing, we choose feature information

$$\boldsymbol{x}_k = \big(W, \nu(T), (\mathcal{Y}_l)_{k-12\leq l\leq k-1}, \mathfrak{y}_{k-1}, O_{k-1}, k, \vartheta\big)^\top \in \mathbb{R}^{17} \times \Theta. \tag{8}$$

We model the probability vector $\boldsymbol{x}_k \mapsto \boldsymbol{p}(\boldsymbol{x}_k)$ given in (7) by a second neural network of depth 3 having (20, 15, 10) hidden neurons in the hidden layers. The explicit network modeling choices are provided in Section C.1 of the supplementary material.

We pre-process our data so that we obtain the observations $(\mathcal{I}_{\ell,k}, \boldsymbol{x}_{\ell,k})$ of all claims ℓ for which we have observed at least two settlement periods up to the cutoff date. This results in 1,753,173 observations $(\mathcal{I}_{\ell,k}, \boldsymbol{x}_{\ell,k})$ that can be used to train the second network. We perform the same training strategy as above, using 20% of the data as validation data to track over-fitting, and applying the bias regularization step, see Section C.1 of the supplementary material.

The results are provided in Table 2, and these are benchmarked with a multinomial logistic GLM (this is described in Section C.1 of the supplementary material) and the homogeneous model that does not consider any covariate information. The in-sample losses support the use of the (more complex) neural network approach over the multinomial logistic GLM (note that this in-sample loss does not over-fit because we exercise early stopping in the network calibration).

We start by analyzing the in-sample figures in the upper triangle. Figure 6 gives the observed and predicted one-period-ahead variables $\mathcal{I}_k$, $k \geq 1$, averaged over all open claims $O_{k-1} = 1$ at the end of month $k-1$: the first graph shows the average probability of an open claim being closed in settlement month k without payment (average 6%), the second graph gives the average probability of a closing with a payment (average 5%), the third graph shows the probability that the claim remains

[4] This undesired extrapolation of trends may best be illustrated by giving an explicit example. If we consider settlement month $k = 1$, inclusion of T in feature $\boldsymbol{x}_1$ may model inflation of payments from 2009 to 2017. Inclusion of T in feature $\boldsymbol{x}_k$ for settlement month $k = 9 \cdot 12 = 108$ would then extrapolate this inflation to years 2018–2026. It is clear that this might be problematic, in particular, because claim payments in early settlement periods (small k) may be of a different nature than payments for later settlement periods (large k), since the former rather relate to simple claims and the latter to complex claims.

Table 2. Indicator process: run time of calibration (including the bias regularization step), number of model parameters, in-sample losses and estimated average probabilities.

	Run time	No. of in-sample parameters	Estimated average loss	Probabilities $\widehat{\boldsymbol{p}}(\cdot)$
Homogeneous model	–	3	0.5838	(93.85%, 1.39%, 3.44%, 1.32%)
Multinomial logistic GLM	590 s	111	0.1660	(93.85%, 1.39%, 3.44%, 1.32%)
Neural network	1493 s	957	0.1576	(93.85%, 1.39%, 3.44%, 1.32%)

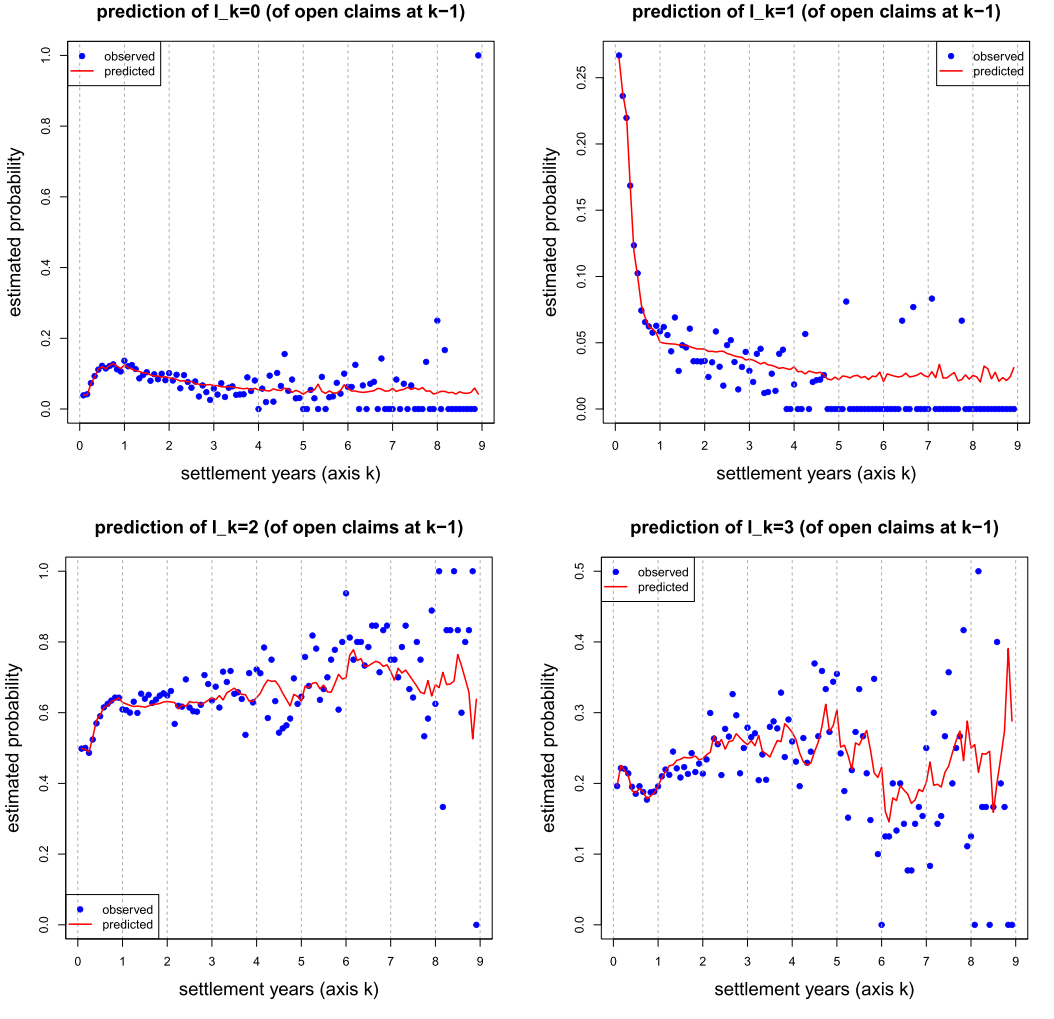

Figure 6. In-sample observed (blue) and predicted (red) one-period-ahead variables $\mathcal{I}_k = \{0, 1, 2, 3\}$ for open claims $O_{k-1} = 1$ one month before $k \geq 1$.

open without a payment (average 67%) and the fourth graph gives the probability of remaining open with a payment in settlement month k (average 22%). The latter graph is quite wiggly for higher settlement months k, this is because we only have very few open claims in later periods (as underlying volume), and the estimated probabilities also depend on the other feature information in $\boldsymbol{x}_k$. Finally, Figure 7 gives the predicted re-openings $\mathcal{I}_k = \{1, 2, 3\}$ for closed claims $O_{k-1} = 0$ one month before. This number is less than 4% and even much smaller for late periods, thus this confirms our scarcity statement about re-openings above.

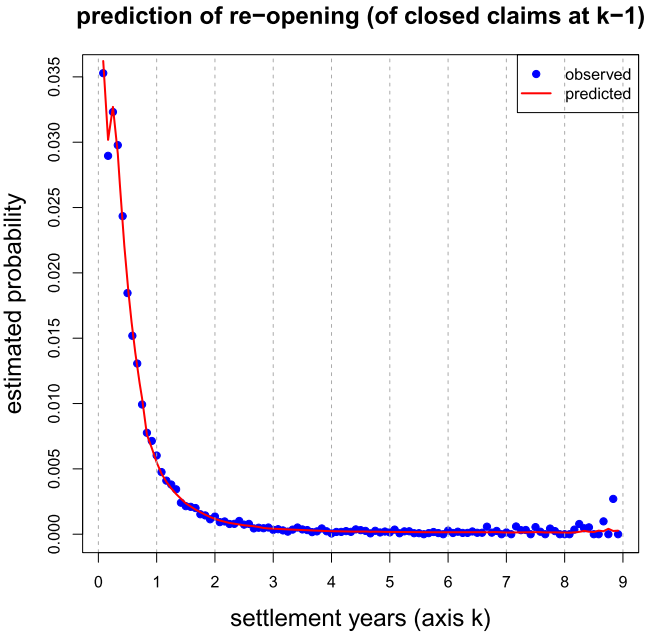

Figure 7. In-sample observed (blue) and predicted (red) re-openings $\mathcal{I}_k = \{1, 2, 3\}$ for closed claims $O_{k-1} = 0$ one month before $k \geq 1$.

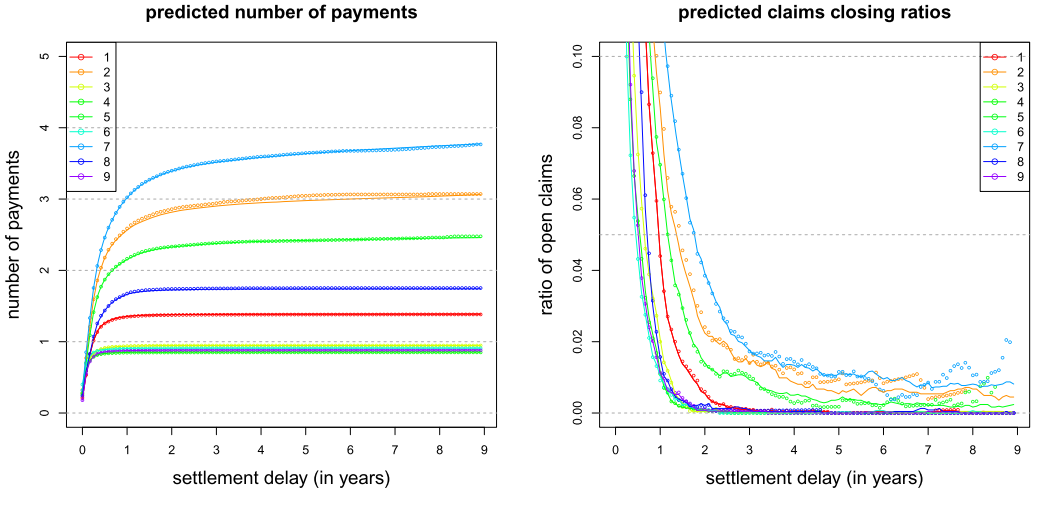

Figure 8. (lhs) Cumulative payment pattern $(\widehat{\Pi}_k)_{k\geq 0}$ and (rhs) claim status ratios $(\widehat{\varpi}_k)_{k\geq 0}$ for the different claim codes: dots show empirical values in the upper triangle (identical to Figure 4) and lines show predicted values extrapolated to the lower triangle (out-of-sample).

4.3.2. Extrapolation of reported claims to the lower triangle

We are now in the position where we can extrapolate the indicator process $(\mathcal{I}_k)_k$ of reported claims to the lower triangle. As described above, we do this by successive one-period-ahead Monte Carlo simulations (based on the estimated categorical probabilities (7)). Monte Carlo simulations are necessary to carry forward the right feature information $\boldsymbol{x}_{k-1}$ for $k > \lceil \tau \rceil - \lceil U \rceil$.

Figure 8 summarizes this successive one-period-ahead Monte Carlo extrapolation to the lower triangle averaged over all reported claims. The dots are exactly the same graphs as in Figure 4; these

are complemented with lines that reflect the Monte Carlo extrapolations to the lower triangle (out-of-sample), based on the estimated claim indicator process described by (7). With regards to payments, only claim code 2 shows some differences (Figure 4, lhs), which emphasizes a small trend in the process. The right-hand side of Figure 4 shows smoothed versions of the claim closings. The claims with codes 2, 5 and 7 have still 0.40%, 0.24% and 0.57%, respectively, of open claims after 10 years of settlement. We conclude that the estimated model extrapolates the indicator processes $(\mathcal{Y}_k, O_k)_{k\in\mathbb{N}_0}$ of reported claims from the upper to the lower triangle such that the basic payment and closing patterns per claim code are preserved, and it does not add an accounting year effect in our case. We emphasize that the statements made in this section are based on one single Monte Carlo extrapolation to the whole lower triangle. In Section 4.3.4, we will average over multiple extrapolations to diminish Monte Carlo simulation error.

4.3.3. Analysis of the extrapolation of reported claims

Using the above Monte Carlo extrapolation for every reported claim $\ell = 1, \ldots, 34,260$, we now have a full trajectory $(\mathcal{Y}_{\ell,k}, O_{\ell,k})_{k=0,\ldots,d}$ over the maximal settlement delay of $d = 107$ months for all claims reported within $\tau_0 = 2009/01/01$ and $\tau = 2017/12/31$. The Monte Carlo simulation provides us with a total of 3,437 monthly payments in the lower triangle for RBNS claims.

Using this extrapolation of the payment indicator process for reported claims we can estimate the number of monthly payments for each reported claim. There is one claim that has predicted 54 monthly payments within the $d+1 = 108$ settlement months, and on average we have 1.7597 monthly payments per claim. The summary statistics over all 34,260 reported claims is given in Table 3. Remarkable is that 9.9% of the claims do not suffer any payments, 58.9% of the claims have one monthly payment and 31.2% of the claims have more than one monthly payment.

In Figure 9, we give the distributions of the total number of (monthly) payments per claim as a function of different features (note that these distributions are discrete but for better visualization we use line plots). The left-hand side shows the distributions for different claim codes, these distributions reflect Figure 8 (lhs) and they illustrate that the claim code feature results in important differences in total number of payments. Similarly for the remaining panels in Figure 9. These plots justify that the chosen feature information is relevant for payment prediction.

Table 3. Estimated probabilities of number of monthly payments per reported claim over $d+1 = 109$ settlement months, with an average of 1.7597 monthly payments per claim (based on one Monte Carlo extrapolation).

# of payments	0	1	2	3	4	5	6	7	8	9	≥ 10
Probability	9.9%	58.9%	16.1%	6.2%	2.6%	1.6%	1.1%	0.8%	0.6%	0.5%	1.7%

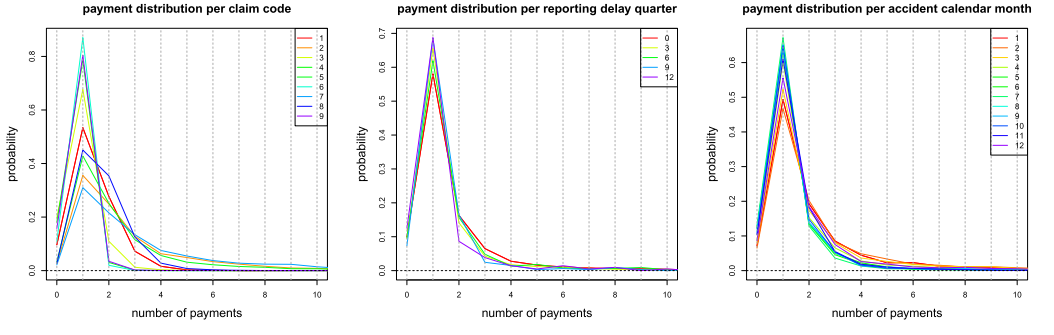

Figure 9. Estimated distributions of number of monthly payments (lhs) for different claim codes, (middle) for different reporting delay quarters, (rhs) for different calendar months of claim occurrences; discrete distributions are for better visualization illustrated by line plots.

Table 4. Number of outstanding monthly payments after the cutoff date restricted to the domain of the CL method; the parentheses show one standard deviation Monte Carlo uncertainty.

	# RBNS payments	# IBNR payments	Total number of payments
Neural network	3,393(19)	708	4,101
ODP CL method	NA	NA	4,064

4.3.4. Comparison to the CL method

We compare our predicted number of monthly payments to the one we receive from the CL method. Intuitively, we expect quite similar results from the two methods because the estimated payment patterns in the upper and lower triangles look rather similar, see Figure 8 (lhs), which supports the CL assumptions. For the CL method we can consider a maximal reporting *plus* settlement delay of $m = 107$ months because the CL model does not differentiate payments w.r.t. reporting delays. For this reason, we also need to cap accordingly the number of payments received from our neural network prediction, because the latter allows us to predict payments up to a maximal delay of $\lceil U \rceil - \lceil T \rceil + d = \lceil U \rceil - \lceil T \rceil + 107 \geq 107$ months. The capping to the CL domain provides us with a predicted number of 3,365 RBNS payments (compared to the 3,437 RBNS payments stated in Section 4.3.3). This number of 3,365 RBNS payments comes from one single Monte Carlo extrapolation to the lower triangle. To diminish Monte Carlo simulation error we repeat this procedure 50 times and average the results. This provides us with an average of 3,466 RBNS payments, and if we truncate this estimate to the CL domain we receive on average 3,393 RBNS payments. This average has an estimated Monte Carlo simulation error (standard deviation) estimate of 19 RBNS payments, see Table 4.

Second, we need to estimate the number of payments for the 402 IBNR claims of Table 1. These IBNR claims are established with feature information like claim type, accident date, etc., and we could design a network to estimate the number of payments for these IBNR claims. We will do this in detail for payment amounts, below. For the moment, we use a crude estimate, which is obtained by multiplying the number of IBNR claims with the average number of payments per claim, which is given by 1.7606 (this corresponds to the number in the caption of Table 3 but averaged over the 50 Monte Carlo extrapolations). This results in 708 payments for IBNR claims, thus, altogether we predict 4,101 outstanding monthly payments after τ, see Table 4.

We compare the above results to the CL method on annualized data. We therefore consider the monthly payment indicators $(\mathcal{Y}_{\ell,k})_k$ of all reported claims ℓ and aggregate them on an annual time grid, so that $\mathcal{Y}^{\text{ann}}_{i,j}$ counts the number of monthly payments of claims with occurrence year $i = \lceil T \rceil$ in development year $j = \lceil U + k \rceil$, where the time ceiling function refers to the annual grid, here. We then apply the CL method to the upper triangle $(\mathcal{Y}^{\text{ann}}_{i,j})_{i+j \leq \lceil \tau \rceil}$. This provides us with an estimate of 4,064 monthly payments in the lower triangle, thus, slightly less than the 4,101 monthly payments obtained from the neural network approach. But, the difference is (surprisingly) small. Applying the ODP bootstrap method of England & Verrall (1999) (with dispersion estimate 6.79) we receive an RMSEP of 279, i.e. of a bigger magnitude than the difference between the two methods in Table 4. Basically, this means that on the aggregate level we cannot distinguish the results. Of course, this is a very promising result for individual claims reserving because it supports that we meet the right level of total predictions, but beyond the ODP CL method we have an allocation to individual claim types, see, e.g. Figure 9.

5. Payment size modeling of RBNS claims

We are on a monthly time scale and we predict the payment process $(Y_k)_{k \geq 1}$ of reported claims, given the indicator process $(\mathcal{Y}_k, O_k)_{k \geq 1}$ (modeled in the previous section), see also Figure 1. We

Table 5. Recovery indicator process: run time of calibration, number of model parameters, loss and estimated average recovery probability.

	Run time	# parameters	In-sample loss	Estimated average probability $\widehat{p}(\cdot)$
Homogeneous model	–	1	0.2235	2.36%
Logistic GLM	1 s	37	0.1900	2.36%
Neural network	30 s	944	0.1580	2.36%

emphasize that the modeling of the indicator process $(\mathcal{Y}_k, O_k)_{k\geq 1}$ does not use any payment size information $(Y_k)_{k\geq 1}$ to make consistent extrapolation to the lower triangle feasible, and we also refer to Limitations 5.1, below, for this extrapolation strategy.

5.1. Recovery payments

In total we have observed 56,852 payments from $\tau_0 = 2009/01/01$ to $\tau = 2017/12/31$ on all reported claims. 1,122 of these payments have a negative sign, i.e. in these cases the insurance company can recover claim expenses, thus, roughly 2% of the payments $Y_{\ell,k}$ are negative (this includes $k = 0$). We choose a next third neural network to model the recovery indicator. We do this as follows, assume that the random variables $\mathcal{R}_k = \mathbb{1}_{\{Y_k<0\}}$ are Bernoulli distributed, conditionally on having a payment $\mathcal{Y}_k = \mathbb{1}_{\{Y_k\neq 0\}} = 1$ in settlement months $k \geq 1$, with recovery probabilities

$$\boldsymbol{x}_k \;\mapsto\; p(\boldsymbol{x}_k) \;\in\; (0,1)$$

for feature information

$$\boldsymbol{x}_k = \left(W, \nu(T), (\mathcal{Y}_l)_{k-12\leq l\leq k-1}, \mathfrak{y}_{k-1}, O_{k-1}, O_k, k, \vartheta\right)^\top \;\in\; \mathbb{R}^{19} \times \{\text{claim code}\}. \tag{9}$$

This feature information differs from (8), because to model recovery indicators we additionally know from the previous modeling step whether the claim has been closed at the end of settlement month k, i.e. we know O_k.

We set up the third neural network for these binary recovery indicators. This is done analogously to the ones used above; the explicit specifications of this logistic neural network are provided in Section C.2 of the supplementary material. Table 5 provides the calibration statistics. We benchmark it against a logistic GLM and the homogeneous model that disregards all covariate information. The (early stopped) neural network again shows superior performance over the GLM, and we continue with the network regression model. Moreover, the last column of Table 5 confirms unbiasedness of the network model, see also Section C.2 of the supplementary material.

Figure 10 compares the estimated neural network model to the observations. We show the probabilities of the recovery rates as a function of settlement month $k \geq 1$ (lhs) and of reporting delay W (rhs). The regression function is less sensitive in other variables such as the (seasonal) calendar week $\nu(T)$ of the accident date T. Remark that the recovery process can easily be extrapolated to the lower triangle with Monte Carlo simulations.

5.2. Descriptive data analysis of claim payment sizes

All previous models have been considering point processes (Poisson, categorical and Bernoulli). Claim sizes are typically modeled with absolutely continuous distribution functions. Before doing so we try to understand claim payment sizes empirically.

Figure 11 shows logarithmized absolute payments $\log|Y_{\ell,k}|$ for settlement months $k = 1, \ldots, 8$ of all claims ℓ with $\mathcal{Y}_{\ell,k} = 1$. The orange curve gives the empirical density and the blue dashed line is a Gaussian approximation (method of moments fit). We observe that the Gaussian distribution

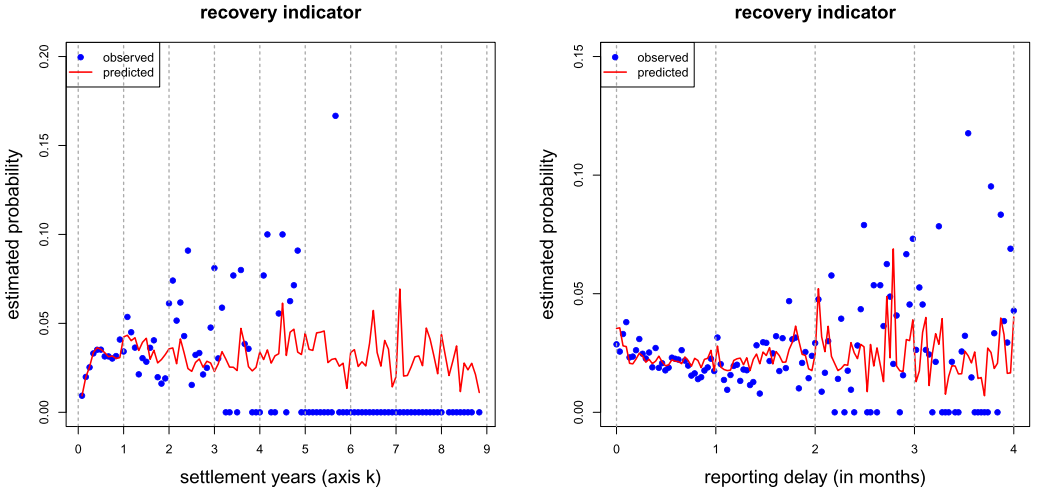

Figure 10. Observed and estimated recovery payment indicators as a function of (lhs) settlement month and (rhs) reporting delay.

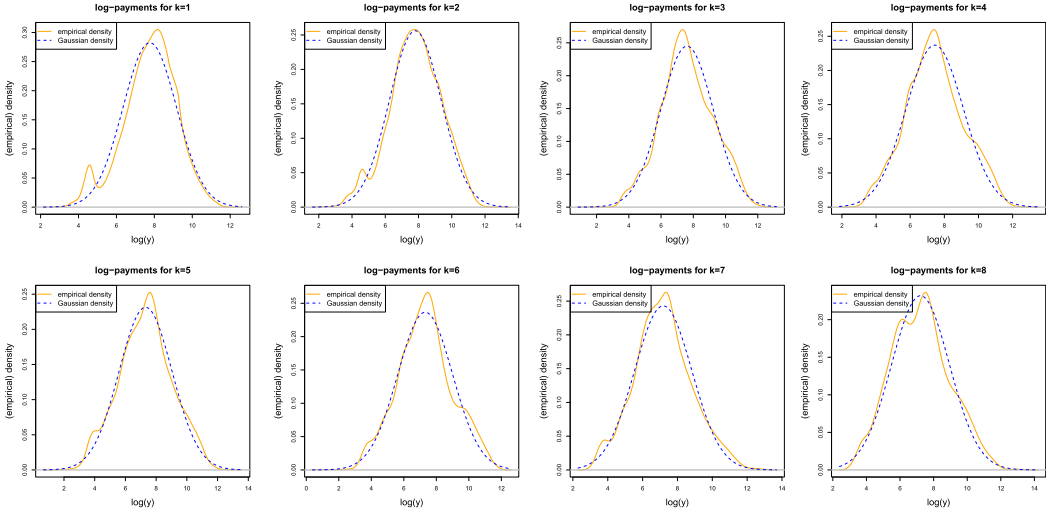

Figure 11. Logarithmized absolute payments $\log|Y_k|$ in settlement months $k = 1, \ldots, 8$.

provides a surprisingly good fit to the logged absolute values $\log|Y|$ of non-zero payments. However, there are certain default values for payments, see for instance settlement month $k = 1$ (top-left). These default values may cause some difficulties in modeling because they rather come from a discrete than an absolutely continuous distribution function.

Figure 12 (first row, lhs) gives the log–log plot of all absolute values of payments. The vertical green lines are at values $\log(50,000)$, $\log(100,000)$ and $\log(250,000)$, the orange straight line gives a Pareto distribution approximation to all payments above 50,000; the maximum likelihood estimator (MLE) of the Pareto tail parameter α is $\widehat{\alpha} = 2.75$ which indicates (together with the plot) heavy-tailedness of large payments. Figure 12 (first row, middle) shows the boxplots for different claim codes. This plot tells us that the claim code is important feature information for payment modeling. Figure 12 (first row, rhs) gives the boxplots as a function of settlement months $k \geq 1$. This plot shows time dependence and, henceforth, rejects many models proposed in the actuarial literature. The second row of Figure 12 shows payments as a function of accident months (related to $v(T)$), the number of

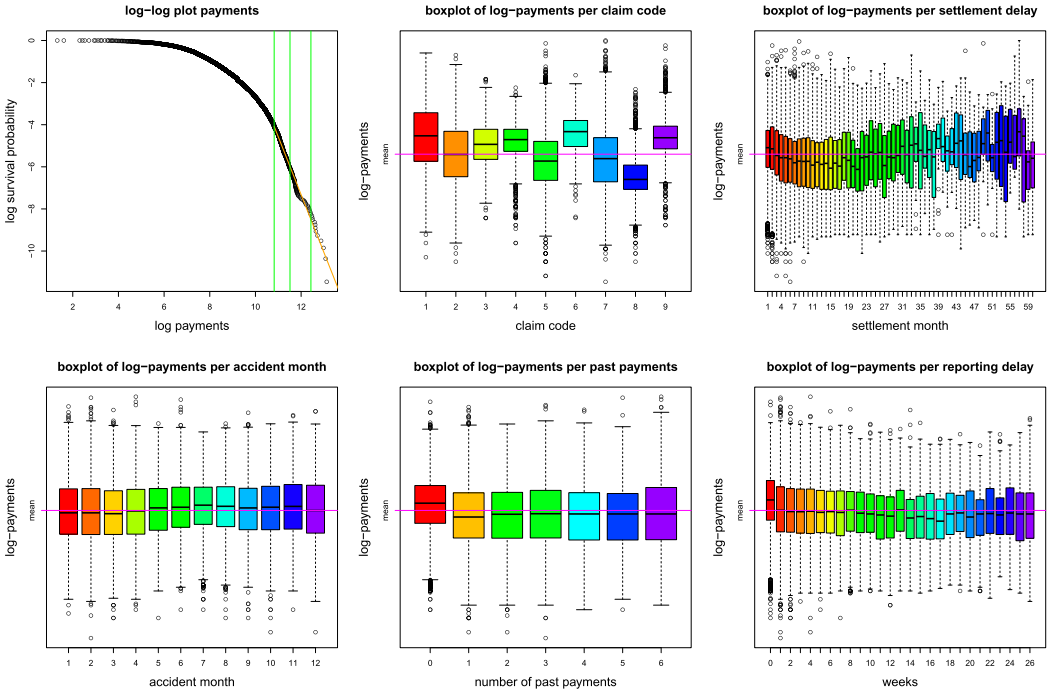

Figure 12. Top row: (lhs) Log–log plot of absolute values of payments $\log|Y_k|$, $k \geq 1$, with Pareto tail approximation, (middle) boxplots of payments w.r.t. claim codes, (rhs) w.r.t. settlement month $k \geq 1$; bottom row boxplots of payments w.r.t. (lhs) accident months, (middle) the number of previous payments, and (rhs) reporting delay W (the magenta lines are the averages over all payments).

previous monthly payments $\mathfrak{y}_{k-1} \in \mathbb{N}_0$ and of reporting delay W. From these plots we conclude that payments are less sensitive in these variables. However, the latter may be misleading because it does not consider interactions of covariates (and, in fact, there are some).

5.3. Payment size regression modeling

We set up a fourth neural network to model the payment sizes. We have observed 56,852 payments. This is a rather small number of observations, and to prevent from over-fitting, we reduce feature information compared to (9). In particular, the masking $(\mathcal{Y}_l)_{l<0} \equiv -1$ for early periods has caused some smoothness issues in our experiments on payment size modeling, therefore, we reduce the payment indicator history in the feature information. On the other hand, we include the recovery indicator $\mathcal{R}_k$, which is available due to the previous modeling step (presented in Section 5.1). We choose the feature information for the modeling of Y_k, conditional on $\mathcal{Y}_k = 1$,

$$\boldsymbol{x}_k = \left(W, v(T), \mathcal{Y}_{k-1}, \mathfrak{y}_{k-1}, O_{k-1}, O_k, \mathcal{R}_k, k, \vartheta\right)^\top \in \mathbb{R}^9 \times \{\text{claim code}\}. \tag{10}$$

We emphasize that this feature information does not include past payment information $(Y_l)_{l<k}$, see discussion in Section 4.1, and therefore no Monte Carlo extrapolation of Y_k is needed.

We explore two different neural networks for claim payments $|Y_k|$ modeling. The architectures of the two networks only differ in the output layer: (a) linear output function for log-payments $\log|Y_k|$ with square loss function and (b) exponential output function for payments $|Y_k|$ with gamma deviance loss function. Besides this difference the two architectures are chosen identical, the explicit specifications are provided in Section C.3 of the supplementary material.

We apply the same fitting strategy as in the previous modeling steps using 20% of the data as validation data to track over-fitting. Here, the choice of the validation data is done in a stratified way

Table 6. Claim size modeling: run time of calibration, number of model parameters, loss, estimated average for: (a0)–(a1) Gaussian model on log-scale with square loss, (a2) bias corrected on original scale with rooted square loss, (b0)–(b1) gamma model on original scale with gamma deviance loss, (b2) bias corrected on original scale with rooted square loss, (c1)–(c2) gamma GLM benchmark.

	Run time	# parameters	In-sample loss	Estimated average
(a0) Homogeneous model on log-scale $\mathbb{E}[\log\lvert Y_k\rvert]$	–	1	2.6221	7.5622
(a1) Gaussian neural network on log-scale $\mathbb{E}[\log\lvert Y_k\rvert]$	89 s	744	2.0352	7.5661
(a2) Bias corrected on original scale $\mathbb{E}[\lvert Y_k\rvert]$			13,208	6,254
(b0) Homogeneous model on original scale $\mathbb{E}[\lvert Y_k\rvert]$	–	1	2.3574	6,254
(b1) Gamma neural network on original scale $\mathbb{E}[\lvert Y_k\rvert]$	91 s	744	1.8137	6,315
(b2) Bias corrected on original scale $\mathbb{E}[\lvert Y_k\rvert]$			12,851	6,254
(c1) Gamma GLM benchmark on original scale $\mathbb{E}[\lvert Y_k\rvert]$	1s	17	1.9352	6,357
(c2) Bias corrected on original scale $\mathbb{E}[\lvert Y_k\rvert]$			13,186	6,254

w.r.t. payment sizes to receive a well-balanced validation set (compared to the training data). The corresponding results are given in Table 6 on lines (a0)–(a1) for the Gaussian case on the log-scale, and on lines (b0)–(b1) for the gamma case on the original scale. Both networks result in a biased estimated model on the original scale because we work on the log-scale in the former and because we do not use the canonical link in the latter. Therefore, we correct for the biases by shifting the intercept parameter in the output layer accordingly, see supplementary material. This provides the results on lines (a2) and (b2) on the original scale (note that the estimated averages of 6,254 are now in line with the empirical estimate for all bias corrected models). To compare the two network models, we calculate the rooted square losses, and we note that the gamma model should be preferred according to this figure.

It is commonly the case that the gamma model has a better performance on the original scale than the log-normal one (the latter being estimated on the log-scale). Usually, the log-normal model is very accurate on the log-scale, but the back-transformation to the original scale would require that also the variance parameter is modeled as a regression function w.r.t. feature information to get sufficient accuracy on the original scale; thus, we have to estimate two parameters with accurate regression models in the log-normal model. On the other hand, the gamma model directly estimates the mean parameter on the original scale, which typically leads to better estimates for means compared to the log-normal model (even if we correct with a constant for the bias, accounting for the fact that we do not use the canonical link). For this reason, we focus on the gamma model in the sequel and we denote the resulting regression function by $\boldsymbol{x}_k \mapsto \mu(\boldsymbol{x}_k)$. This gamma neural network model is benchmarked against a gamma GLM in Table 6; the figures indicate that we should continue with the more complex neural network.[5]

Figure 13 (lhs) shows the quantile-quantile (QQ)-plot of the shifted residuals $\text{Res}_k + 1 = (|Y_k| - \mu(\boldsymbol{x}_k))/\mu(\boldsymbol{x}_k) + 1$ which should have a gamma distribution with mean 1 and constant dispersion. Figure 13 (lhs) shows that the observations are clearly more heavy tailed than the gamma model allows for. This is also supported by Figure 13 (rhs) which provides the Tukey–Anscombe plot. Additionally, we have explored a small-large payments split (composite model), and we have fitted a gamma model to the small payments. The resulting Tukey–Anscombe plot looked very similar to the one in Figure 13 (rhs); moreover, the manual intervention of choosing a large payment threshold was clearly visible in the plot (which is unsatisfactory).

Conclusion. We continue with gamma model (b2) of Table 6. This only requires one single network to be fitted and one single (mean) parameter to be estimated. Because we are *only* interested in having good mean estimates this is sufficient for our purposes.

Limitations 5.1: Obviously, the gamma model is not sufficiently accurate to simulate payment sizes Y_k, it is only suitable for mean estimation $\mu(\boldsymbol{x}_k)$. This can be seen as a limitation of our approach

[5] Note that a gamma GAM has a similar performance as the gamma GLM. The corresponding gamma loss of the GAM is 1.9246 and the rooted mean square loss is 13,172.

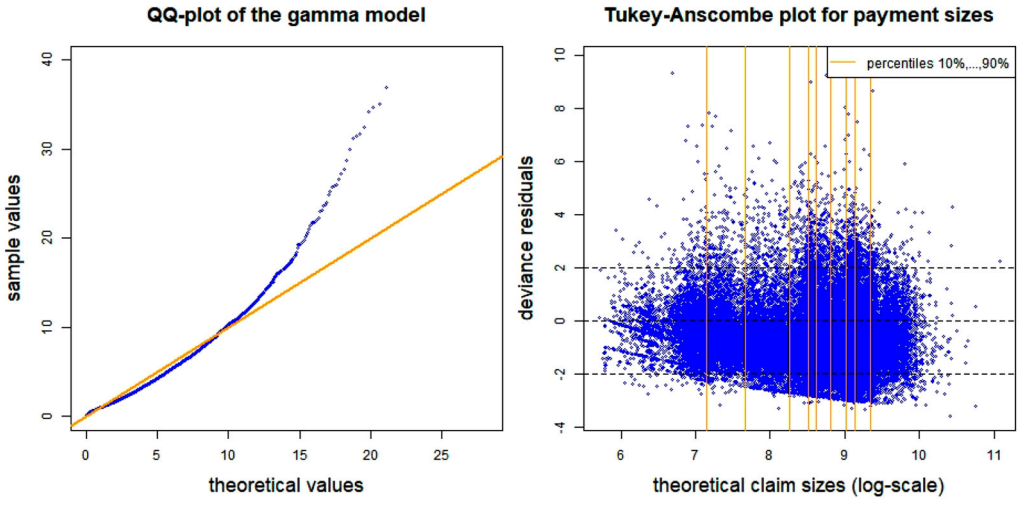

Figure 13. (lhs) QQ-plot of the estimated gamma model (b2) of Table 6 and (rhs) Tukey–Anscombe plot of the residuals against the estimated means.

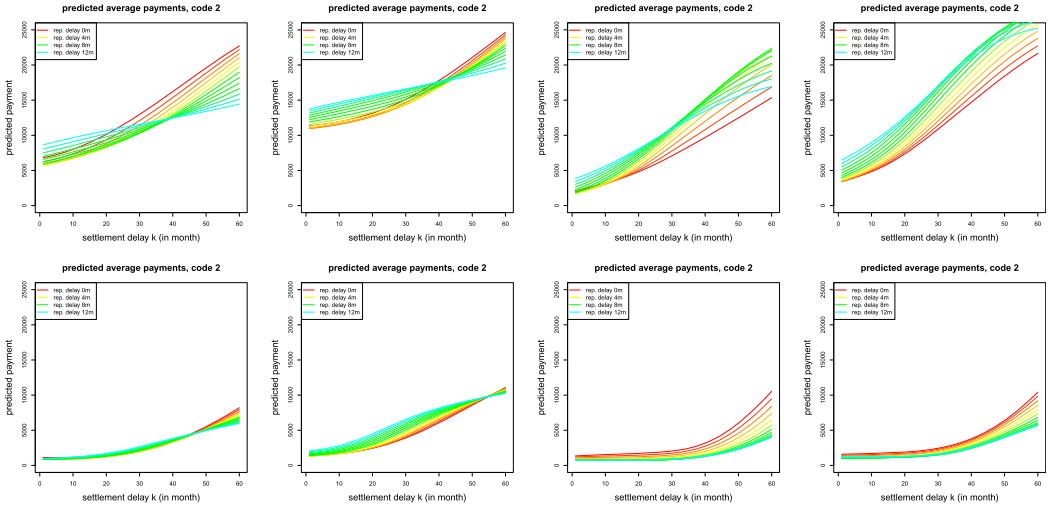

Figure 14. (First row) Predicted payments and (second row) predicted recovery payments for different features $\boldsymbol{x}$ and identical claim code 2: the x-axis illustrates different settlement months $k \in \{0, \ldots, 60\}$, the different colors illustrate different reporting delays from 0 days to 1 year and the plots from left to right show $(\mathcal{I}_{k-1}, O_k) = (0,1), (1,1), (0,0), (1,0)$.

because we cannot simulate payment sizes. This is also the reason for not including payment size information into the covariates for the prediction of the indicator process $(\mathcal{Y}_k, O_k)_{k\geq 0}$. Naturally, we expect that the predictive performance can be improved by including payment size information. However, then extrapolation to the lower triangle is only feasible if one has a reasonable Monte Carlo simulation tool for these payments sizes.

The fitted network allows us to study expected payments $\mu(\boldsymbol{x}_k)$ as a function of $\boldsymbol{x}_k$. Figure 14 illustrates such plots. We fix claim code 2, the x-axis illustrates different settlement months $k \in \{0, \ldots, 60\}$, and the different colors illustrate different reporting delays from 0 days to 1 year. The first row gives predicted claim payments and the second row predicted recovery payments (conditional on $\mathcal{R}_k = 1$). The first column gives the predictions for claims without payments in the previous period $\mathcal{Y}_{k-1} = 0$

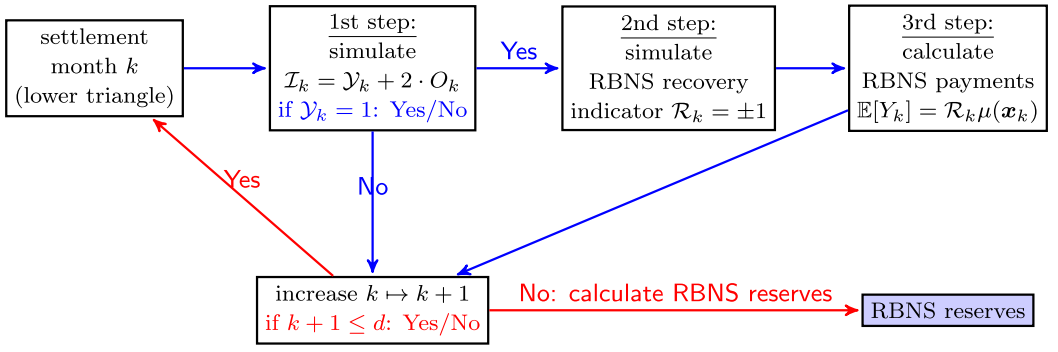

Figure 15. RBNS claims reserves simulation architecture with maximal settlement delay d.

and claims remaining open $O_k = 1$ after settlement period k, the second column for $\mathcal{Y}_{k-1} = 1$ and $O_k = 1$, the third column for $\mathcal{Y}_{k-1} = 0$ and $O_k = 0$, and the fourth column for $\mathcal{Y}_{k-1} = 1$ and $O_k = 0$. We observe that all this feature information influences predicted payments and recoveries. For these plots the undetermined feature components in (10) have been set to $\nu(T) = 1/4$ (March), $O_{k-1} = 1$ (open), $\eta_{k-1} = \mathcal{Y}_{k-1}$ (only a potential payment in the previous period) and the LoB has been chosen to be the more likely one for the given claim code 2.

5.4. Extrapolation of payments to the lower triangle

We expand the payment processes of RBNS claims to the lower triangle (this encloses relapses of closed claims). We do this by successive one-period-ahead predictions. For each reported claim, this includes three steps for each settlement month k in the lower triangle, see Figure 15:

1st step : Simulate the categorical payment and claim status indicator $\mathcal{I}_k = \mathcal{Y}_k + 2 \cdot O_k$ based on feature information (8); if $\mathcal{Y}_k = 1$, process the next two steps, otherwise go to settlement month $k+1$ for this claim.

2nd step : Simulate the Bernoulli recovery indicator $\mathcal{R}_k$ based on feature information (9), this feature information includes the previously simulated claim status indicator O_k.

3rd step : Calculate the expected payment size $\mathbb{E}[Y_k] = \mathcal{R}_k \mu(\boldsymbol{x}_k)$ based on feature $\boldsymbol{x}_k$ given in (10), this includes the previously simulated claim status indicator O_k and the recovery indicator $\mathcal{R}_k = \pm 1$. Proceed to settlement month $k+1$.

This procedure is iterated over the entire lower triangle for each reported claim.

To reduce simulation error, we average the above procedure over 50 Monte Carlo extrapolations in the lower triangle. This provides us with an RBNS reserve of $23,832 \cdot 10^3$ for all reported claims at the cutoff date τ. If we again truncate these payments to the CL domain, see Section 4.3.4, we receive RBNS reserves of $23,160 \cdot 10^3$ (having a Monte Carlo simulation error (standard deviation) of $195 \cdot 10^3$). In the next section we compare these results to the classical CL reserves. We conclude this section by giving some plots of the RBNS reserves.

We are now able to illustrate the reserves for reported claims w.r.t. the available feature information. Figure 16 (lhs, middle) illustrates payments and reserves for reported claims at the cutoff date τ as a function of accident year and of claim codes, respectively. Figure 16 (rhs) shows the split of reserves to open reported claims and closed claims, the latter are reserves for re-openings for relapses. We note that in our data there is quite some influence by relapses (in terms of reserves), though less than 3% of all closed claims are re-opened, see Figure 7.

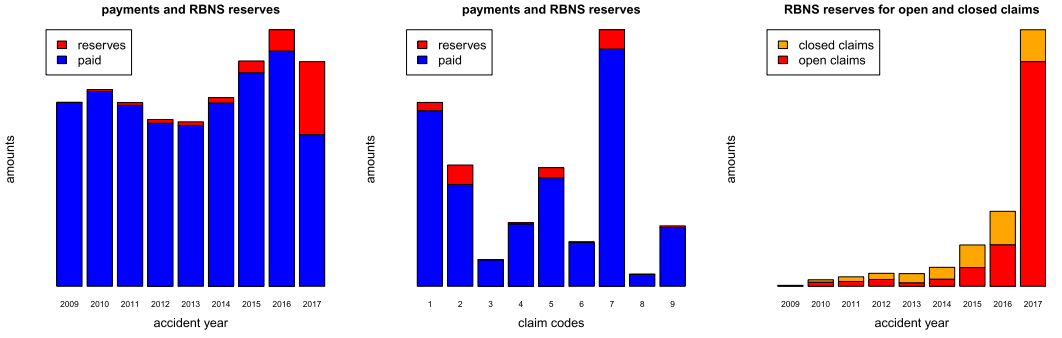

Figure 16. Payments and reserves for reported claims (lhs) as a function of accident year, (middle) split by claim codes, (rhs) reserves split by open and closed claims at the cutoff date (the latter being reserves for re-openings coming from relapses).

6. Claim amounts for IBNR claims

We have calculated the RBNS reserves for reported claims in Sections 4–5. In Section 3, we have predicted the number of IBNR counts, we expect 402 IBNR claims, see Table 1. In the present section we assign claim amounts to these IBNR claims. For this we use a weekly time scale as in Section 3, and the time ceiling function $\lceil \cdot \rceil$ refers to the weekly time scale in this section.

6.1. Description of our IBNR claim modeling approach

The modeling of IBNR claim amounts is done differently compared to the claim payments modeling of RBNS claims. This difference in modeling is common in actuarial practice because the nature of the underlying data is different. For RBNS claims we have partially observed individual claim settlement processes, and the main task in RBNS reserving is to complete these individual claim histories in a path-dependent way. We have done this by successive one-period-ahead predictions, see Figure 15. For IBNR claims we do not have partially observed histories, and there is no point in performing multiple path-dependent Monte Carlo simulations, but, instead, ultimate claim amounts of individual claims are estimated directly, still considering individual (static) claim feature information if available. In our case this information is exactly the (weekly) feature information $(i, j, v(i), \vartheta) = (\lceil T \rceil, \lceil U \rceil - \lceil T \rceil, v(\lceil T \rceil), \vartheta)$ from the IBNR claim counts modeling, see (5).

6.2. IBNR claim amount modeling

To estimate IBNR claim amounts we set up networks 5 and 6. The fifth network serves at predicting whether we have a claim without any payments (called zero claim), and the sixth network will assess the total claim amount in case of a non-zero claim. For both networks we use exactly the same weekly feature information $(i, j, v(i), \vartheta) = (\lceil T \rceil, \lceil U \rceil - \lceil T \rceil, v(\lceil T \rceil), \vartheta)$ as for IBNR claim counts modeling, see (5); this feature information corresponds to the claim occurrence week, the reporting delay in weekly units, the annual seasonality and the static feature information $\vartheta \in \Theta$. The total IBNR reserves are then obtained by

$$\sum_{\vartheta} \sum_{i+j>\lceil \tau \rceil} \widehat{\Lambda}(i, j, v(i), \vartheta) \left(1 - \widehat{q}(i, j, v(i), \vartheta)\right) \widehat{\gamma}(i, j, v(i), \vartheta), \tag{11}$$

where $\widehat{\Lambda}(i, j, v(i), \vartheta)$ is the estimated number of IBNR claims (estimated by the first network given in Section 3.2), $\widehat{q}(i, j, v(i), \vartheta) \in (0, 1)$ is the estimated probability of having a zero claim (coming from the fifth network), and $\widehat{\gamma}(i, j, v(i), \vartheta) > 0$ is the estimated claim amount in case of a non-zero claim (coming from the sixth network).

We start by describing the available data for this prediction. Above, we have designed networks that allow us to complete the path-dependent trajectories of claims reported at the cutoff date τ. Thus, for all reported claims we now have a fully developed (predicted) claim history over the $d+1 = 108$ settlement months which provides total payments per reported claim. There is one tricky issue that has already been mentioned in Section 4.1: in the upper triangle we have observed payments Y_k and in the lower triangle we have predicted payments $\widehat{Y}_k = \mathbb{E}[Y_k] = \mu(\boldsymbol{x}_k)$. Thus, we have an inconsistency between upper and lower triangle, and we are not allowed to simply aggregate these payments and predicted payments to assess the total individual claim amounts, because claims with different settlement delays would be of different nature. For this reason we replace non-zero payments Y_k in the upper triangle by corresponding (in-sample) predictions $\widehat{Y}_k = \mu(\boldsymbol{x}_k)$. The last column of Table 6 tells us that this replacement is unbiased, i.e. the total prediction is exactly equal to the total payments in the upper triangle; this unbiasedness has been enforced by setting the intercept in the output layer correspondingly, see Section C.3 in the supplementary material. After this replacement in the upper triangle we aggregate all predicted payments to receive the total predicted claim amounts $\widehat{C}_\ell = \sum_{0\le k\le d} \widehat{Y}_{\ell,k}$ for all reported claims $\ell \ge 1$. These predicted claim amounts $\widehat{C}_\ell$ are complemented with the static feature information $(\lceil T_\ell \rceil, \lceil U_\ell \rceil - \lceil T_\ell \rceil, \nu(\lceil T_\ell \rceil), \vartheta_\ell)$.

Based on feature information $(i, j, \nu(i), \vartheta)$ we design networks 5 and 6 to estimate the zero claim probability $q(i, j, \nu(i), \vartheta) \in (0, 1)$ and the claim amount $\gamma(i, j, \nu(i), \vartheta) > 0$ in case of a non-zero claim. Similar to Section 3.2, we design a network with logistic Bernoulli responses for assessing zero claims, and one with gamma responses for the total claim amounts for non-zero claims, this is described in more detail in Section D of the supplementary material. The resulting calibration statistics are given in Table 7, and the last column of this table confirms that we work with an unbiased model.

Established with these two additional networks we complete the predicted IBNR claim counts of Section 3.2 with total individual claim amounts, see (11). This results in IBNR reserves of $3,338 \cdot 10^3$, thus, altogether we receive $26,498 \cdot 10^3$ RBNS and IBNR reserves from the network approach. We compare this estimate to the one received from the CL method, see Table 8. The difference between the two methods is surprisingly small, i.e. only slightly more than half of one RMSEP of $2,730 \cdot 10^3$, received by the bootstrap method of England & Verrall (1999) in the ODP model. This small difference is a second proof of concept that our neural network approach provides reasonable results, given that the CL reserves meets the right level of reserves. In fact, we predict slightly higher claims than in the CL method; this can be justified because the CL method has difficulties to appropriately consider seasonality, change of portfolio mix, etc., but overall the CL reserves have proved (in the past) to provide reasonable reserves on this data set. An advantage of our approach is that it allows us to analyze the reserves as a function of all available features.

Table 7. IBNR zero claim indicators and IBNR claim amounts (in case of non-zero claims): run time of calibration, number of model parameters, loss and estimated averages.

	Run time	# parameters	In-sample loss	Estimated averages $\widehat{q}(\cdot)$ and $\widehat{\gamma}(\cdot)$, respectively
IBNR zero claim indicator: homogeneous model	–	1	0.5217	7.28%
IBNR zero claim indicator: neural network	18 s	644	0.4551	7.28%
IBNR claim amounts: homogeneous model	–	1	2.1498	11,109
IBNR claim amounts: neural network	13 s	644	1.6905	11,109

Table 8. Predicted payments (claims reserves) after the cutoff date restricted to the domain of the CL method (units in 1000).

	RBNS reserves	IBNR reserves	Total reserves
Neural network	23,160	3,338	26,498
ODP CL method	NA	NA	25,011
RMSEP (dispersion $\phi = 108,944$)			2,730

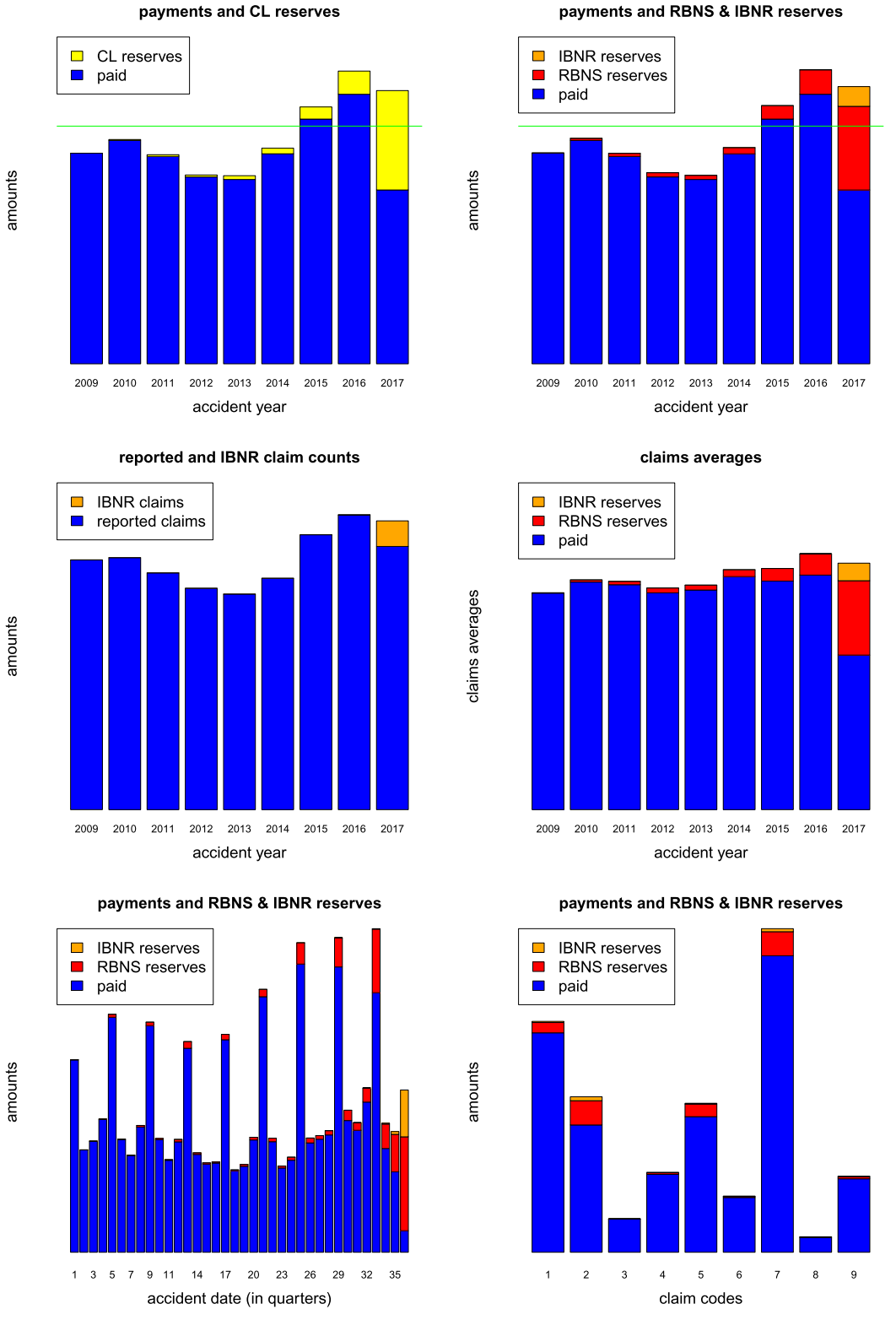

Figure 17. (top lhs) Payments and CL reserves, (top rhs) payments, RBNS and IBNR reserves from network approach, (middle lhs) number of claims, (middle rhs) claims averages, (bottom lhs) payments and reserves per accident quarter, and (bottom rhs) payments and reserves per claim codes.

The top row of Figure 17 shows the two methods split w.r.t. accident years: allocation of reserves to accident years is very similar between the CL method and our network approach. The middle row illustrates the number of claims and the resulting claim averages (based on the neural network prediction). We predict a slight decrease on average claim amounts from 2016 compared to 2017. The bottom row of Figure 17 shows the claims split according to accident quarters (lhs) and claim codes (rhs). Interestingly, we have a much bigger claim payment exposure in the first quarters of the calendar years, and summer time is a comparably quiet period. Moreover, reserves for older claims are mainly set for the first accident quarters. From Figure 1 in the supplementary material, we observe that claim codes 1, 2 and 7 have mainly exposure in winter months, which justifies the claim codes split in Figure 16 (bottom rhs).

7. Conclusion

We have presented an individual claims reserving approach which considers individual claims features and past claims history. This approach is based on an architecture consisting of six neural networks, see Figure 1, each serving a certain purpose:

- network 1 models claim counts of IBNR claims;
- network 2 serves at modeling the payment indicator and the claim status processes of RBNS claims;
- network 3 serves at modeling an indicator process for recovery payments of RBNS claims;
- network 4 serves at modeling expected claim and recovery payments of RBNS claims;
- network 5 models an indicator for having an IBNR claim without payments (zero claim);
- the final network 6 models claim amounts for non-zero IBNR claims.

The basic network architecture is the same for all six networks, only the input dimensions differ (depending on the available feature information in each modeling step), and the output functions differ (depending on the type of distribution function we would like to model). We fit this model on weekly data for IBNR claims and on monthly data for RBNS claims. The run times of these modeling steps are reasonably short (as indicated in the corresponding tables), and the resulting reserves almost meet the benchmark received from the CL method. The latter is surprising and a critical reader may question our complex model if a simple one comes to the same conclusion. The benefits of our more complex model are that we receive claims reserves on a granular level, for instance, we can assess claims reserves for a given claim type (encoded in ϑ) occurring in a certain calendar month. Such information is important for the understanding of all single modules that form an insurance contract and, henceforth, for product development. Another advantage of our approach compared to the CL method is that we can cope with seasonality, trends, changes in portfolio mix and, even, changes in legislation. The latter has not been proved in this paper, but we use the same approach for an industry project on motor third party liability data where the jurisdiction has changed two times over the last 30 years.

The main compromise to achieve good run times, here, is that we cannot simulate individual claim histories. Our model provides reasonable claims reserves, however, our claim size distribution is not suitable for simulation because it underestimates the tails of the claims. If one is interested in a good simulation model, this will add another layer of complexity as has been shown in our previous paper (Delong & Wüthrich 2020). Of course, also this latter approach has its advantageous and disadvantageous, for instance, robustness of results needs to be carefully checked in more complex models.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The research of Ł. Delong was supported by Narodowe Centrum Nauki [2018/31/B/HS4/02150].

References

Antonio, K. & Plat, R. (2014). Micro-level stochastic loss reserving for general insurance. *Scandinavian Actuarial Journal* **2014**(7), 649–669.

Arjas, E. (1989). The claims reserving problem in non-life insurance: some structural ideas. *ASTIN Bulletin* **19**(2), 139–152.

Badescu, A. L., Chen, T., Lin, X. S. & Tang, D. (2019). A marked Cox model for the number of IBNR claims: estimation and application. *ASTIN Bulletin* **49**(3), 709–739.

Baudry, M. & Robert, C. Y. (2019). A machine learning approach for individual claims reserving in insurance. *Applied Stochastic Models in Business and Industry* **35**(5), 1127–1155.

Bischofberger, M. S. (2019). In-sample hazard forecasting based on survival models with operational time. *Risks* **8**(1), 3.

Bischofberger, M. S., Hiabu, M. & Isakson, A. (2019). Continuous chain-ladder with paid data. *Scandinavian Actuarial Journal* **2019**(10), 1–26.

Breiman, L. (2001). Statistical modeling: the two cultures. *Statistical Science* **16**(3), 199–215.

Crevecoeur, J., Antonio, K. & Verbelen, R. (2019). Modeling the number of hidden events subject to observation delay. *European Journal of Operational Research* **277**(3), 930–944.

De Felice, M. & Moriconi, F. (2019). Claim watching and individual claims reserving using classification and regression trees. *Risks* **7**(4), 102.

Delong, Ł. & Wüthrich, M. V. (2020). Neural networks for the joint development of individual payments and claim incurred. *Risks* **8**(2), 33.

Duval, F. & Pigeon, M. (2019). Individual loss reserving using a gradient boosting-based approach. *Risks***7**(3), 79.

England, P. D. & Verrall, R. J. (1999). Analytic and bootstrap estimates of prediction errors in claims reserving. *Insurance: Mathematics and Economics* **25**(3), 281–293.

Gabrielli, A. (2020). A neural network boosted double overdispersed Poisson claims reserving model. *ASTIN Bulletin* **50**(1), 25–60.

Jessen, A. H., Mikosch, T. & Samorodnitsky, G. (2011). Prediction of outstanding payments in a Poisson cluster model. *Scandinavian Actuarial Journal* **2011**(3), 214–237.

Kuo, K. (2020). Individual claims forecasting with Bayesian mixture density networks, arXiv:2003.02453v1.

Larsen, C. R. (2007). An individual claims reserving model. *ASTIN Bulletin* **37**(1), 113–132.

Lindholm, M. & Verrall, R. (2020). Regression based reserving models and partial information. *Insurance: Mathematics and Economics* **94**, 109–124.

Lopez, O. & Milhaud, X. (2021). Individual reserving and nonparametric estimation of claim amounts subject to large reporting delays. *Scandinavian Actuarial Journal* **2021**(1), 34–53.

Lopez, O., Milhaud, X. & Thérond, P.-E. (2019). A tree-based algorithm adapted to microlevel reserving and long development claims. *ASTIN Bulletin* **49**(3), 741–762.

Maciak, M., Okhrin, O. & Pešta, M. (2019). Infinitely stochastic micro forecasting, arXiv:1908.10636v2.

Martínez-Miranda, M. D., Nielsen, J. P. & Verrall, R. (2012a). Double chain ladder. *ASTIN Bulletin***42**(1), 59–76.

Martínez-Miranda, M. D., Nielsen, J. P. & Wüthrich, M. V. (2012b). Statistical modelling and forecasting of outstanding liabilities in non-life insurance. *Statistics and Operations Research Transactions* **36**(2), 195–218.

Norberg, R. (1993). Prediction of outstanding liabilities in non-life insurance. *ASTIN Bulletin* **23**(1), 95–115.

Norberg, R. (1999). Prediction of outstanding liabilities II. Model variations and extensions. *ASTIN Bulletin* **29**(1), 5–25.

Okine, A. N.-A., Frees, E. & Shi, P. (2019). Joint model prediction and application to individual-level loss reserving, preprint.

Pigeon, M., Antonio, K. & Denuit, M. (2013). Individual loss reserving with the multivariate skew normal framework. *ASTIN Bulletin* **43**(3), 399–428.

Rosenlund, S. (2012). Bootstrapping individual claim histories. *ASTIN Bulletin* **42**(1), 291–324.

Taylor, G., McGuire, G. & Sullivan, J. (2008). Individual claim loss reserving conditioned by case estimates. *Annals of Actuarial Science* **3**(1–2), 215–256.

Tukey, J. W. (1977). *Exploratory data analysis*. Reading, MA: Addison-Wesley.

Verrall, R. J. & Wüthrich, M. V. (2016). Understanding reporting delay in general insurance. *Risks***4**(3), 25.

Wahl, F., Lindholm, M. & Verrall, R. (2019). The collective reserving model. *Insurance: Mathematics and Economics* **87**, 34–50.

Wang, Z., Wu, X. & Qiu, C. (2021). The impacts of individual information on loss reserving. *ASTIN Bulletin* **51**(1), 303–347.

Wüthrich, M. V. (2018). Machine learning in individual claims reserving. *Scandinavian Actuarial Journal* **2018**(6), 465–480.

Wüthrich M. V. & Merz M. (2008). *Stochastic claims reserving methods in insurance*. Chichester: Wiley.

Wüthrich, M. V. & Merz, M. (2021). Statistical foundations of actuarial learning and its applications, SSRN Preprint ID 3822407.

Zhao, X. B., Zhou, X. & Wang, J. L. (2009). Semiparametric model for prediction of individual claim loss reserving. *Insurance: Mathematics and Economics* **45**, 1–8.