Chapter 10
IBNR techniques

IBNR reserves represent an important cog in the insurance accounting machinery — Bornhuetter & Ferguson, 1978

10.1 Introduction

In the past, non-life insurance portfolios were financed through a pay-as-you-go system. All claims in a particular year were paid from the premium income of that same year, no matter in which year the claim originated. The financial balance in the portfolio was realized by ensuring that there was an equivalence between the premiums collected and the claims paid in a particular financial year. Technical gains and losses arose because of the difference between the premium income in a year and the claims paid during the year.

The claims originating in a particular year often cannot be finalized in that year. For example, long legal procedures are the rule with liability insurance claims. But there may also be other causes for delay, such as the fact that the exact size of the claim is hard to assess. Also, the claim may be filed only later, or more payments than one have to be made, such as in disability insurance. All these factors will lead to delay of the actual payment of the claims. The claims that have already occurred, but are not sufficiently known, are foreseeable in the sense that one knows that payments will have to be made, but not how much the total payment is going to be. Consider also the case that a premium is paid for the claims in a particular year, and a claim arises of which the insurer is not notified as yet. Here also, we have losses that have to be reimbursed in future years.

Such claims are now connected to the years for which the premiums were actually paid. This means that reserves have to be kept regarding claims that are known to exist, but for which the eventual size is unknown at the time the reserves have to be set. For claims like these, several acronyms are in use. One has IBNR claims (Incurred But Not Reported) for claims that have occurred but have not been filed. Hence the name IBNR methods, IBNR claims and IBNR reserves for all quantities of this type. There are also RBNS claims (Reported But Not Settled), for claims that are known but not (completely) paid. Other acronyms are IBNFR, IBNER and RBNFS, where the F is for Fully, the E for Enough. Large claims known to the insurer are often handled on a case-by-case basis.

Table 10.1 A run-off triangle with payments by development year (horizontally) and year of origin (vertically)

Year of origin	Development year							
	1	2	3	4	5	6	7	8
2011	101	153	52	17	14	3	4	1
2012	99	121	76	32	10	3	1	
2013	110	182	80	20	21	2		
2014	160	197	82	38	19			
2015	161	254	85	46				
2016	185	201	86					
2017	178	261						
2018	168							

When modeling these situations, one generally starts from a so-called *run-off triangle*, containing loss figures, for example cumulated payments, for each combination of policy year and development year. It is compiled in the following way:

1. We start in 2011 with a portfolio of insurance contracts. Let us assume that the total claims to be paid are fully known on January 1, 2019, eight years after the end of this year of origin;
2. The claims occurring in the year 2011 have to be paid from the premiums collected in 2011;
3. These payments have been made in the year 2011 itself, but also in the years 2012–2018;
4. In the same way, for the claims pertaining to the year of origin 2012, one has the claims which are known in the years 2012–2018, and it is unknown what has to be paid in 2019;
5. For the year 2016, the known claims are the ones paid in the period 2016–2018, but there are also unknown ones that will come up in the years 2019 and after;
6. For the claims concerning the premiums paid in 2018, on December 31, 2018 only the payments made in 2018 are known, but we can expect that more payments will have to be made in and after 2009. We may expect that the claims develop in a pattern similar to the one of the claims in 2011–2018.

The development pattern can schematically be depicted as in Table 10.1. The numbers in the triangle are the known total payments, grouped by year of origin i (by row) and development year j (by column). The row corresponding to year 2013 contains the six numbers known on December 31, 2018. The third element in this row, for example, denotes the claims incurred in 2013, but paid for in the third year of development 2015. In the triangle of Table 10.1, we look at new contracts only. This situation may occur when a new type of policy was issued for the first time in 2011. The business written in this year on average has had only half a year to produce claims in 2011, which is why the numbers in the first column are somewhat lower than those in the second. The numbers on the diagonal with $i + j - 1 = k$ denote the payments that were made in calendar year k. There are many ways to group

these same numbers into a triangle, but the one given in Table 10.1 is the customary one. On the basis of the claim figures in Table 10.1, which could be claim numbers but also more general losses, we want to make predictions about claims that will be paid, or filed, in future calendar years. These future years are to be found in the bottom-right part of Table 10.1. The goal of the actuarial IBNR techniques is to predict these figures, so as to complete the triangle into a square. The total of the figures found in the lower right triangle is the total of the claims that will have to be paid in the future from the premiums that were collected in the period 2011–2018. This total is precisely the reserve to be kept.

We assume that the development pattern lasts eight years. In many branches, notably in liability, claims may still be filed after a time longer than eight years. In that case, we have to make predictions about development years after the eighth, of which our run-off triangle provides no data. We not only have to extend the triangle to a square, but to a rectangle containing more development years. The usual practice is to assume that the development procedure is stopped after a number of years, and to apply a correction factor for the payments made after the development period considered.

The future payments are estimated following well-established actuarial practice. Sometimes one central estimator is given, but also sometimes a whole range of possibilities is considered, containing both the estimated values and, conceivably, the actual results. Not just estimating the mean, but also getting an idea of the variance of the results is important. Methods to determine the reserves have been developed that each meet specific requirements, have different model assumptions, and produce different estimates. In practice, sometimes the method that is the most likely to produce the 'best' estimator is used to determine the estimate of the expected claims, while the results of other methods are used as a means to judge the variation of the stochastic result, which is of course a rather unscientific approach.

To complete the triangle in Table 10.1, we can give various methods, each reflecting the influence of a number of exogenous factors. In the direction of the year of origin, variation in the size of the portfolio will have an influence on the claim figures. On the other hand, for the factor development year (horizontally), changes in the claim handling procedure as well as in the speed of finalization of the claims will produce a change. The figures on the diagonals correspond to payments in a particular calendar year. Such figures will change due to monetary inflation, but also by changing jurisprudence or increasing claim proneness. As an example, in liability insurance for the medical profession the risk increases each year, and if the amounts awarded by judges get larger and larger, this is visible along the diagonals. In other words, the separation models, which have as factors the year of development and the calendar year, would be the best choice to describe the evolution of portfolios like these.

Obviously, one should try to get as accurate a picture as possible about the stochastic mechanism that produced the claims, test this model if possible, and estimate the parameters of this model optimally to construct good predictors for the unknown observations. Very important is how the variance of claim figures is related to the mean value. This variance can be more or less constant, it can be proportional to the

mean, proportional to the square of the mean (meaning that the coefficient of variation is a constant), or have some other relation with it. See the following section, as well as the chapters on Generalized Linear Models.

Just as with many rating techniques, see the previous chapter, in the actuarial literature quite often a heuristic method to complete an IBNR triangle was described first, and a sound statistical foundation was provided only later. In Section 10.2, we describe briefly the two most often used techniques to make IBNR forecasts: the chain ladder method and the Bornhuetter-Ferguson technique. There is a very basic generalized linear model (GLM) for which the ML-estimators can be computed by the well-known chain ladder method. On the other hand it is possible to give a model that involves a less rigid statistical structure and in which the calculations of the chain ladder method produce an optimal estimate in the sense of mean squared error. In Section 10.3 we give a general GLM, special cases of which can be shown to boil down to familiar methods of IBNR estimation such as the arithmetic and the geometric separation methods, as well as the chain ladder method. A numerical illustration is provided in Section 10.4, where various sets of covariates are used in GLMs to complete the triangle in Table 10.1. How to use R to do the calculations is described in Section 10.5. In Section 10.6, an analytical estimate of the prediction error of the chain ladder method is studied, as well as a bootstrap method. They were proposed by England and Verrall (1999) and England (2002). In this way, a standard error of prediction and an approximate predictive distribution for the random future losses are produced. In Section 10.7, we give another example, in which the parameters relating to the accident year are replaced by the known portfolio size, expressed in its number of policies or its premium income. A method related to Bornhuetter-Ferguson arises.

10.2 Two time-honored IBNR methods

The two methods most frequently used in practice are the chain ladder (CL) method and the Bornhuetter-Ferguson method. We give a short description here; for the R implementation, we refer to later sections.

10.2.1 Chain ladder

The idea behind the chain ladder method is that in any development year, about the same total percentage of the claims from each year of origin will be settled. In other words, in the run-off triangle, the columns are proportional. To see how in the chain ladder method predictions are computed for the unobserved part of a run-off rectangle, look at Table 10.2. Note that in most texts, the run-off figures given are cumulated by rows. This is a relic of the time when calculations had to be done by hand. In this text, we avoid this custom.

Table 10.2 Completing a run-off rectangle with CL predictions

	1	2	3	4	5
01	$\mathscr{A}$	$\mathscr{A}$	$\mathscr{A}$	$\mathscr{B}$	•
02	$\mathscr{A}$	$\mathscr{A}$	$\mathscr{A}$	$\mathscr{B}$	
03	$\mathscr{C}$	$\mathscr{C}$	$\mathscr{C}$	$\star$	
04	$\mathscr{D}$	$\mathscr{D}$	$\widehat{\mathscr{D}}$	$\star\star$	
05	•				

Consider the $(3,4)$ element in Table 10.2, denoted by $\star$ and representing payments regarding policy year 03 in their 4th development year. This is a claim figure for calendar year 06, which is the first future calendar year, and just beyond the edge of the observed figures. Because of the assumed proportionality, the ratio of the elements $\star : \mathscr{C}$ will be about equal to the ratio $\mathscr{B} : \mathscr{A}$. Therefore, a prediction $\widehat{X}_{34}$ of this element $\star$ is

$$\widehat{X}_{34} = \mathscr{C}_{\Sigma} \times \frac{\mathscr{B}_{\Sigma}}{\mathscr{A}_{\Sigma}}. \tag{10.1}$$

Here $\mathscr{B}_{\Sigma}$, for example, denotes the total of the $\mathscr{B}$-elements in Table 10.2, which are observed values. Prediction $\widehat{\mathscr{D}}$ is computed in exactly the same way, multiplying the total of the incremental payments to the left of it by the total above it, and dividing by the total of losses of earlier policy years and development years. The prediction $\star\star$ for $\widehat{X}_{44}$ (policy year 04, calendar year 07, so one year further in the future) can be computed by using the same 'development factor' $\mathscr{B}_{\Sigma}/\mathscr{A}_{\Sigma}$:

$$\widehat{X}_{44} = \mathscr{D}_{\Sigma} \times \frac{\mathscr{B}_{\Sigma}}{\mathscr{A}_{\Sigma}}, \tag{10.2}$$

where the sum $\mathscr{D}_{\Sigma}$ includes $\widehat{\mathscr{D}}$, which is not an actual observation but a prediction constructed as above. By using the fact that $\star = \mathscr{C}_{\Sigma} \times \mathscr{B}_{\Sigma}/\mathscr{A}_{\Sigma}$, it is easy to see that exactly the same prediction is obtained by taking

$$\widehat{X}_{44} = \frac{\mathscr{D}_{\Sigma} \times (\mathscr{B}_{\Sigma} + \star)}{\mathscr{A}_{\Sigma} + \mathscr{C}_{\Sigma}}, \tag{10.3}$$

hence by following the same procedure as for an observation in the next calendar year. In this way, starting with row 2 and proceeding from left to right, the entire lower triangle can be filled with predictions.

Remark 10.2.1 (Mirror property of the chain ladder method)
Note that this procedure produces the same estimates to complete the square if we exchange the roles of development year and year of origin, hence take the mirror image of the triangle around the NW–SE diagonal. ∇

Remark 10.2.2 (Marginal totals property of the chain ladder method)
One way to describe the chain ladder method is as follows: find numbers $\widehat{\alpha}_i, \widehat{\beta}_j$, $i, j = 1, \ldots, t$ such that the products $\widehat{\alpha}_i \widehat{\beta}_j$ (fitted values) for 'observed' combinations

(i,j) with $i+j-1 \leq t$ have the same column sums and row sums as the actual observations:

$$\sum_{j(i)} \widehat{\alpha}_i \widehat{\beta}_j = \sum_{j(i)} X_{ij} \quad \text{for all } i(j). \tag{10.4}$$

Then predict future values for (i,j) with $i+j-1 > t$ by $\widehat{\alpha}_i \widehat{\beta}_j$. How to find these numbers $\widehat{\alpha}_i, \widehat{\beta}_j$ will be described later on.

We will illustrate why this procedure leads to the same forecasts as the chain ladder method by looking at Table 10.2. First observe that $\mathscr{B}_\Sigma$ and $\mathscr{C}_\Sigma$ are already column and row sums, but also the sums of claim figures $\mathscr{A}_\Sigma$ needed can be computed from these quantities. For instance in our example, $\mathscr{A}_\Sigma = R_1 + R_2 - (C_5 + C_4)$ when R_i and C_j denote the ith row sum and the jth column sum.

Next, observe that if we replace the past losses X_{ij} by their fitted values $\widehat{\alpha}_i \widehat{\beta}_j$, the row and column sums remain unchanged, and therefore also the quantities like $\mathscr{A}_\Sigma$. When the chain ladder algorithm described above is applied to the new triangle, the numbers $\widehat{\alpha}_i \widehat{\beta}_j$ result as future predictions. ∇

The basic principle of the chain ladder method admits many variants. One may wonder if there is indeed proportionality between the columns. Undoubtedly, this is determined by effects that operate along the axis describing the year of origin of the claims. By the chain ladder method, only the run-off pattern can be captured, given that all other factors that have an influence on the proportion of claims settled remain unchanged over time.

The chain ladder method is merely an algorithm, a deterministic method. But there are also stochastic models for the generating process underlying the run-off triangle in which these same calculations lead to an optimal prediction in some sense. See, for example, Section 10.3.1.

10.2.2 Bornhuetter-Ferguson

One of the difficulties with using the chain ladder method is that reserve forecasts can be quite unstable. In Table 10.2, a change of $p\%$ in $\mathscr{C}_\Sigma$ due to sampling variability will generate the same change in all forecasts for this row. So applying this method to a volatile claims experience will produce volatile forecasts. This volatility will show itself by changes in the reserve estimate each year, when a new diagonal of observations is added to the triangle. The Bornhuetter-Ferguson (1972) method provides a procedure for stabilizing such estimates.

Suppose that one has some prior expectation as to the ultimate losses to emerge from each accident period i, specifically, that $\mathrm{E}[X_{i1} + \cdots + X_{it}] = M_i$ for some known quantity M_i. This quantity is often referred to as the *schedule* or *budget ultimate losses*. Notably one may have a prior view of the loss ratio M_i/P_i, where P_i is the premium income with accident year i. Combining these prior estimates with the

Table 10.3 Random variables in a run-off triangle

Year of origin	Development year 1	$\cdots$	$t-n+1$	$\cdots$	t
1	X_{11}	$\cdots$	$X_{1,t-n+1}$	$\cdots$	X_{1t}
$\vdots$	$\vdots$		$\vdots$		$\vdots$
n	X_{n1}	$\cdots$	$X_{n,t-n+1}$	$\cdots$	X_{nt}
$\vdots$	$\vdots$		$\vdots$		$\vdots$
t	X_{t1}	$\cdots$	$X_{t,t-n+1}$	$\cdots$	X_{tt}

development factors of the chain ladder method, one may form an estimate of the entire schedule of loss development. See Section 10.7 for more details.

It can be shown that the Bornhuetter-Ferguson method can be interpreted as a Bayesian method. The forecasts have the form of a credibility estimator.

10.3 A GLM that encompasses various IBNR methods

Several often used and traditional actuarial methods to complete an IBNR triangle can be described by one Generalized Linear Model. In Table 10.3, the random variables X_{ij} for $i, j = 1,2,\ldots,t$ denote the claim figure for year of origin i and year of development j, meaning that the claims were paid in calendar year $i+j-1$. For (i, j) combinations with $i+j-1 \leq t$, X_{ij} has already been observed, otherwise it is a future observation. As well as claims actually paid, these figures may also be used to denote quantities such as loss ratios. We take a multiplicative model, with a parameter for each row i, each column j and each diagonal $k = i+j-1$, as follows:

$$X_{ij} \approx \alpha_i \cdot \beta_j \cdot \gamma_k. \tag{10.5}$$

The deviation of the observation on the left hand side from its mean value on the right hand side is attributed to chance. As one sees, if we assume further that the random variables X_{ij} are independent and restrict their distribution to be in the exponential dispersion family, (10.5) is a Generalized Linear Model in the sense of Chapter 9. Year of origin i, year of development j and calendar year $k = 1+j-1$ act as explanatory variables for the observation X_{ij}. The expected value of X_{ij} is the exponent of the linear form $\log\alpha_i + \log\beta_j + \log\gamma_k$, so there is a logarithmic link. Note that the covariates are all dummies representing group membership for rows, columns and diagonals in Table 10.3. We will determine maximum likelihood estimates of the parameters α_i, β_j and γ_k, under various assumptions for the probability distribution of the X_{ij}. It will turn out that in this simple way, we can generate quite a few widely used IBNR techniques.

Having found estimates of the parameters, it is easy to extend the triangle to a square, simply by taking

$$\widehat{X}_{ij} := \widehat{\alpha}_i \cdot \widehat{\beta}_j \cdot \widehat{\gamma}_k. \tag{10.6}$$

A problem is that we have no data on the values of the γ_k for future calendar years k with $k > t$. The problem can be solved, for example, by assuming that the γ_k have a geometric pattern, with $\gamma_k \propto \gamma^k$ for some real number γ.

10.3.1 Chain ladder method as a GLM

The first method that can be derived from model (10.5) is the *chain ladder* method of Section 10.2.1. Suppose that we restrict model (10.5) to:

$$X_{ij} \sim \text{Poisson}(\alpha_i \beta_j) \text{ independent; } \gamma_k \equiv 1. \tag{10.7}$$

If the parameters $\alpha_i > 0$ and $\beta_j > 0$ are to be estimated by maximum likelihood, we have in fact a multiplicative GLM with Poisson errors and a log-link, because the observations X_{ij}, $i, j = 1, \ldots, t$; $i + j \leq t$ are independent Poisson random variables with a logarithmic model for the means; explanatory variables are the factors row number and column number.

By Property 9.3.9 it follows that the marginal totals of the triangle, hence the row sums R_i and the column sums C_j of the observed figures X_{ij}, must be equal to the predictions $\sum_j \widehat{\alpha}_i \widehat{\beta}_j$ and $\sum_i \widehat{\alpha}_i \widehat{\beta}_j$ for these quantities; see (10.4). So it follows from Remark 10.2.2 that the optimal estimates of the parameters α_i and β_j produced by this GLM are equal to the parameter estimates found by the chain ladder method.

One of the parameters is superfluous, since if we replace all α_i and β_j by $\delta\alpha_i$ and β_j/δ we get the same expected values. To resolve this ambiguity, we impose an additional restriction on the parameters. We could use a 'corner restriction', requiring for example $\alpha_1 = 1$, but a more natural restriction to ensure identifiability of the parameters is to require $\beta_1 + \cdots + \beta_t = 1$. This allows the β_j to be interpreted as the fraction of claims settled in development year j, and α_i as the 'volume' of year of origin i: it is the total of the payments made.

Maximizing the likelihood with model (10.7) can be done by an appropriate call of R's function `glm`; see Section 10.5. But by the triangular shape of the data, the system of marginal totals equations admits the following recursive solution method, originally devised by Verbeek (1972) for the case of the arithmetic separation method below.

Algorithm 10.3.1 (Verbeek's algorithm for chain ladder)
Look at Table 10.4. The row and column totals are those of the past X_{ij} in Table 10.3. To solve the marginal totals equations (10.4), we can proceed as follows.

1. From the first row sum equality $\widehat{\alpha}_1(\widehat{\beta}_1 + \cdots + \widehat{\beta}_t) = R_1$ it follows that $\widehat{\alpha}_1 = R_1$. Then from $\widehat{\alpha}_1 \widehat{\beta}_t = C_t$ we find $\widehat{\beta}_t = C_t/R_1$.

Table 10.4 The marginal totals equations in a run-off triangle

Year of origin	Development year 1	⋯	$t-n+1$	⋯	t	Row total
1	$\alpha_1\beta_1$		$\alpha_1\beta_{t-n+1}$		$\alpha_1\beta_t$	R_1
⋮						⋮
n	$\alpha_n\beta_1$		$\alpha_n\beta_{t-n+1}$			R_n
⋮						⋮
t	$\alpha_t\beta_1$					R_t
Column total	C_1	⋯	C_{t-n+1}	⋯	C_t	

2. Assume that, for a certain $n < t$, we have found estimates $\widehat{\beta}_{t-n+2},\ldots,\widehat{\beta}_t$ and $\widehat{\alpha}_1,\ldots,\widehat{\alpha}_{n-1}$. Then look at the following two marginal totals equations:

$$\begin{aligned}\widehat{\alpha}_n(\widehat{\beta}_1+\cdots+\widehat{\beta}_{t-n+1}) &= R_n;\\ (\widehat{\alpha}_1+\cdots+\widehat{\alpha}_n)\widehat{\beta}_{t-n+1} &= C_{t-n+1}.\end{aligned} \tag{10.8}$$

   By the fact that we take $\widehat{\beta}_1+\cdots+\widehat{\beta}_t = 1$, the first of these equations directly produces a value for $\widehat{\alpha}_n$, and then we can compute $\widehat{\beta}_{t-n+1}$ from the second one.
3. Repeat step 2 for $n = 2,\ldots,t$. ∇

10.3.2 Arithmetic and geometric separation methods

In the separation models, one assumes that in each year of development a fixed percentage is settled, and that there are additional effects that operate in the diagonal direction (from top-left to bottom-right) in the run-off triangle. So this model describes best the situation that there is inflation in the claim figures, or when the risk increases by other causes. This increase is characterized by an index factor for each calendar year, which is a constant for the observations parallel to the diagonal. One supposes that in Table 10.4, the random variables X_{ij} are average loss figures, where the total loss is divided by the number of claims, for year of origin i and development year j.

Arithmetic separation method The arithmetic separation method was described in Verbeek (1972), who applied the model to forecast the number of stop-loss claims reported. As time goes by, due to inflation more claims will exceed the retention, and this effect must be included in the model. In both the arithmetic and the geometric separation method the claim figures X_{ij} are explained by two aspects of time, just as for chain ladder and Bornhuetter-Ferguson. But in this case there is a calendar year effect γ_k, where $k = i+j-1$, and a development year effect β_j. So inflation and

run-off pattern are the determinants for the claim figures now. For the *arithmetic separation method* we assume

$$X_{ij} \sim \text{Poisson}(\beta_j \gamma_k) \text{ independent; } \alpha_i \equiv 1. \tag{10.9}$$

Again, β_j and γ_k are estimated by maximum likelihood. Since this is again a GLM (Poisson with the canonical log-link), because of Property 9.3.9 the marginal totals property must hold here as well. In model (10.9) these marginal totals are the column sums and the sums over the diagonals, with $i + j - 1 = k$.

The parameter estimates in the arithmetic separation method can be obtained by a variant of Method 10.3.1 (Verbeek) for the chain ladder method computations. We have $\mathrm{E}[X_{ij}] = \beta_j \gamma_{i+j-1}$. Again, the parameters β_j, $j = 1, \ldots, t$ describe the proportions settled in development year j. Assuming that the claims are all settled after t development years, we have $\beta_1 + \cdots + \beta_t = 1$. Using the marginal totals equations, see Table 10.4, we can determine directly the optimal factor $\widehat{\gamma}_t$, reflecting base level times inflation, as the sum of the observations on the long diagonal $\sum_i X_{i,t+1-i}$. Since β_t occurs in the final column only, we have $\widehat{\beta}_t = \widehat{X}_{1t}/\widehat{\gamma}_t$. With this, we can compute $\widehat{\gamma}_{t-1}$, and then $\widehat{\beta}_{t-1}$, and so on. Just as with the chain ladder method, the estimates thus constructed satisfy the marginal totals equations, and hence are maximum likelihood estimates because of Property 9.3.9.

To fill out the remaining part of the square, we also need values for the parameters $\gamma_{t+1}, \ldots, \gamma_{2t}$, to be multiplied by the corresponding $\widehat{\beta}_j$ estimate. We find values for these parameters by extending the sequence $\widehat{\gamma}_1, \ldots, \widehat{\gamma}_t$ in some way. This can be done with many techniques, for example loglinear extrapolation.

Geometric separation method The geometric separation method involves maximum likelihood estimation of the parameters in the following statistical model:

$$\log(X_{ij}) \sim \mathrm{N}\left(\log(\beta_j \gamma_k), \sigma^2\right) \text{ independent; } \alpha_i \equiv 1. \tag{10.10}$$

Here σ^2 is an unknown variance. We get an ordinary regression model with $\mathrm{E}[\log X_{ij}] = \log \beta_j + \log \gamma_{i+j-1}$. Its parameters can be estimated in the usual way, but they can also be estimated recursively in the way described above, starting from $\prod_j \beta_j = 1$.

Note that the values $\beta_j \gamma_{i+j-1}$ in this lognormal model are *not* the expected values of X_{ij}. In fact, they are only the medians; we have

$$\Pr[X_{ij} \leq \beta_j \gamma_{i+j-1}] = \frac{1}{2} \quad \text{but} \quad \mathrm{E}[X_{ij}] = \mathrm{e}^{\sigma^2/2} \beta_j \gamma_{i+j-1}. \tag{10.11}$$

10.3.3 De Vijlder's least squares method

In the least squares method of De Vylder (1978), he assumes that $\gamma_k \equiv 1$ holds, while α_i and β_j are determined by minimizing the sum of squares $\sum_{i,j} (X_{ij} - \alpha_i \beta_j)^2$,

Table 10.5 Run-off data used in De Vylder (1978)

Year of origin	Development year 1	2	3	4	5	6
1						4627
2					15140	13343
3				43465	19018	12476
4			116531	42390	23505	14371
5		346807	118035	43784	12750	12284
6	308580	407117	132247	37086	27744	
7	358211	426329	157415	68219		
8	327996	436774	147154			
9	377369	561699				
10	333827					

taken over the set (i, j) for which observations are available. But this is tantamount to determining α_i and β_j by maximum likelihood in the following model:

$$X_{ij} \sim \mathrm{N}(\alpha_i \beta_j, \sigma^2) \text{ independent; } \gamma_k \equiv 1. \tag{10.12}$$

Just as with chain ladder, we assume that the mean payments for a particular year of origin/year of development combination result from two effects. First, a parameter characterizing the year of origin, proportional to the size of the portfolio in that year. Second, a parameter determining which proportion of the claims is settled through the period that claims develop. The parameters are estimated by least squares.

In practice it quite often happens that not all data in an IBNR-triangle are actually available. In De Vylder (1978) a 10×10 IBNR-triangle is studied missing all the observations from calendar years $1, \ldots, 5$, as well as those for development years $7, \ldots, 10$. What these numbers represent is not relevant. See Table 10.5.

This paper is the first to mention that in IBNR problems, time operates in three different ways: by policy year i reflecting growth of the portfolio, by development year j reflecting the run-off pattern of the claims, and by calendar year $k = i + j - 1$ reflecting inflation and changes in jurisprudence. De Vijlder proposed to use a multiplicative model $\alpha_i \beta_j \gamma_k$ for the data X_{ij}, and to choose those parameter values α_i, β_j and γ_k that minimize the least squares distance, therefore solving:

$$\min_{\alpha_i, \beta_j, \gamma_k} \sum_{i,j} w_{ij} (X_{ij} - \alpha_i \beta_j \gamma_{i+j-1})^2. \tag{10.13}$$

We multiply by weights $w_{ij} = 1$ if y_{ij} is an actual observation, $w_{ij} = 0$ otherwise, so the sum can be taken over all (i, j) combinations. De Vijlder proceeds by taking the inflation component fixed, hence $\gamma_k = \gamma^k$ for some real γ, and proves that doing this, one might actually have left out inflation of the model altogether, thus taking $\gamma_k \equiv 1$. See Exercise 10.3.4. Next he describes the method of successive substitution to solve the reduced problem, and gives the results for this method. Note that his paper was written in pre-PC times; he used a programmable hand-held calculator.

10.4 Illustration of some IBNR methods

Obviously, introducing parameters for the three time aspects year of origin, year of development and calendar year sometimes leads to overparameterization. Many of these parameters could be dropped, that is, taken equal to 1 in a multiplicative model. Others might be required to be equal, for example by grouping classes having different values for some factor together. Admitting classes to be grouped leads to many models being considered simultaneously, and it is sometimes hard to construct proper significance tests in these situations. Also, a classification of which the classes are ordered, such as age class or bonus-malus step, might lead to parameters giving a fixed increase per class, except perhaps at the boundaries or for some other special class. In a loglinear model, replacing arbitrary parameter values, associated with factor levels (classes), by a geometric progression in these parameters is easily achieved by replacing the dummified factor by the actual levels again, or in GLIM parlance, treating this variable as a variate instead of as a factor. Replacing arbitrary values α_i, with $\alpha_1 = 1$, by α^{i-1} for some real α means that we assume the portfolio to grow, or shrink, by a fixed percentage each year. Doing the same to the parameters β_j means that the proportion settled decreases by a fixed fraction with each development year. Quite often, the first development year will be different from the others, for example because only three quarters are counted as the first year. In that case, one does best to allow a separate parameter for the first year, taking parameters $\beta_1, \beta^2, \beta^3, \ldots$ for some real numbers β_1 and β. Instead of with the original t parameters $\beta_1, \ldots, \beta_t$, one works with only two parameters. By introducing a new dummy explanatory variable to indicate whether the calendar year $k = i + j - 1$ with observation X_{ij} is before or after k_0, and letting it contribute a factor 1 or δ to the mean, respectively, one gets a model for which in one year, the inflation differs from the standard fixed inflation of the other years. Other functional forms for the β_j parameters include the Hoerl-curve, where $\beta_j = \exp(\gamma j + \delta \log j)$ for some real numbers γ and δ. These can be used for all rows in common, or for each row separately (interaction).

In the previous chapter, we introduced the (scaled) deviance as a 'distance' between the data and the estimates. It is determined from the difference of the maximally attainable likelihood and the one of a particular model. Using this, one may test if it is worthwhile to complicate a model by introducing more parameters. For a nested model, of which the parameter set can be constructed by imposing linear restrictions on the parameters of the original model, it is possible to judge if the distance between data and estimates is 'significantly' larger. It proves that this difference in distance, under the null-hypothesis that the eliminated parameters are superfluous, is approximately χ^2 distributed, when suitably scaled. In similar fashion, the 'goodness of fit' of non-nested models can be compared by using the Akaike information criterion, see Remark 9.4.4.

Some regression software leaves it to the user to resolve the problems arising from introducing parameters with covariates that are linearly dependent of the others, the so-called 'dummy trap' (multicollinearity). The `glm` function in `R` is more user-friendly in this respect. For example if one takes all three effects in (10.5) geo-

Table 10.6 Parameter set, degrees of freedom (= number of observations less number of estimated parameters), and deviance for several models applied to the data of Table 10.1.

Model	Parameters used	Df	Deviance	
I	$\mu, \alpha_i, \beta_j, \gamma_k$	15	25.7	
II	μ, α_i, β_j	21	38.0	
III	μ, β_j, γ_k	21	36.8	
IV	$\mu, \beta_j, \gamma^{k-1}$	27	59.9	
V	$\mu, \alpha^{i-1}, \beta_j$	27	59.9	
VI	$\mu, \alpha_i, \gamma^{k-1}$	27	504.	
VII	$\mu, \alpha_i, \beta^{j-1}$	27	504.	
VIII	$\mu, \alpha_i, \beta_1, \beta^{j-1}$	26	46.0	♡
IX	$\mu, \alpha^{i-1}, \beta_1, \beta^{j-1}$	32	67.9	
X	$\mu, \alpha^{i-1}, \beta^{j-1}$	33	582.	
XI	μ	35	2656	

metric, with as fitted values

$$\widehat{X}_{ij} = \widehat{\mu}\,\widehat{\alpha}^{i-1}\widehat{\beta}^{j-1}\widehat{\gamma}^{i+j-2}, \tag{10.14}$$

R does not stop but simply proceeds by taking the last of these three parameters to be equal to 1; see Exercise 10.4.2. Notice that by introducing $\widehat{\mu}$ in (10.14), all three parameter estimates can have the form $\widehat{\alpha}^{i-1}$, $\widehat{\beta}^{j-1}$ and $\widehat{\gamma}^{i+j-2}$. In the same way, we can take $\alpha_1 = \beta_1 = \gamma_1 = 1$ in (10.5). The parameter $\mu = \mathrm{E}[X_{11}]$ is the level in the first year of origin and development year 1.

10.4.1 Modeling the claim numbers in Table 10.1

We fitted a number of models to explain the claim figures in Table 10.1. They were actually claim numbers; the averages of the payments are shown in Table 10.8. To judge which model best fits the data, we estimated a few models for (10.5), all assuming the observations to be Poisson$(\alpha_i\beta_j\gamma_{i+j-1})$. See Table 10.6. By imposing (loglinear) restrictions like $\beta_j = \beta^{j-1}$ or $\gamma_k \equiv 1$, we reproduce the various models discussed earlier. The reader may verify that in model I, one may choose $\gamma_8 = 1$ without loss of generality. This means that model I has only 6 more parameters to be estimated than model II. Notice that for model I with $\mathrm{E}[X_{ij}] = \mu\alpha_i\beta_j\gamma_{i+j-1}$, there are $3(t-1)$ parameters to be estimated from $t(t+1)/2$ observations, hence model I only makes sense if $t \geq 4$.

It can be shown that we get the same estimates using either of the models $\mathrm{E}[X_{ij}] = \mu\alpha_i\beta^{j-1}$ and $\mathrm{E}[X_{ij}] = \mu\alpha_i\gamma^{i+j-1} = (\mu\gamma)(\alpha_i\gamma^{i-1})(\beta_j\gamma^{j-1})$. Completing the triangle of Table 10.1 into a square by using model VIII produces Table 10.7. The column 'Total' contains the row sums of the estimated future payments, hence exactly the amount to be reserved regarding each year of origin. The figures in the top-left part

Table 10.7 The claim figures of Table 10.1 estimated by model VIII. The last column gives the totals for all the future predicted payments.

Year of origin	Development year 1	2	3	4	5	6	7	8	Total
2000	102.3	140.1	510.4	25.2	10.7	4.5	1.9	0.8|	0.0
2001	101.6	1310.2	510.1	25.0	10.6	4.5	1.9|	0.8	0.8
2002	124.0	1610.9	72.1	30.6	13.0	5.5|	2.3	1.0	3.3
2003	150.2	205.8	87.3	37.0	15.7|	6.7	2.8	1.2	10.7
2004	170.7	233.9	910.2	42.1|	17.8	7.6	3.2	1.4	30.0
2005	1510.9	2110.1	92.9|	310.4	16.7	7.1	3.0	1.3	67.5
2006	185.2	253.8|	107.6	45.7	110.4	8.2	3.5	1.5	185.8
2007	168.0|	230.2	97.6	41.4	17.6	7.4	3.2	1.3	398.7

are estimates of the already observed values, the ones in the bottom-right part are predictions for future payments.

All other models are nested in model I, since its set of parameters contains all other ones as a subset. The estimates for model I best fit the data. About the deviances and the corresponding numbers of degrees of freedom, the following can be said. The chain ladder model II is not rejected statistically against the fullest model I on a 95% level, since it contains six parameters fewer, and the χ^2 critical value is 12.6 while the difference in scaled deviance is only 12.3. The arithmetic separation model III fits the data somewhat better than model II. Model IV with an arbitrary run-off pattern β_j and a constant inflation γ is equivalent to model V, which has a constant rate of growth for the portfolio. In Exercise 10.3.3, the reader is asked to explain why these two models are identical. Model IV, which is nested in III and has six parameters fewer, predicts significantly worse. In the same way, V is worse than II. Models VI and VII again are identical. Their fit is bad. Model VIII, with a geometric development pattern except for the first year, seems to be the winner: with five parameters fewer, its fit is not significantly worse than model II in which it is nested. It fits better than model VII in which the first column is not treated separately. Comparing VIII with IX, we see that a constant rate of growth in the portfolio must be rejected in favor of an arbitrary growth pattern. In model X, there is a constant rate of growth as well as a geometric development pattern. The fit is bad, mainly because the first column is so different.

From model XI, having only a constant term, we see that the 'percentage of explained deviance' of model VIII is more than 98%. But even model IX, which contains only a constant term and three other parameters, already explains 97.4% of the deviation.

The estimated model VIII gives the following predictions:

$$\begin{aligned} &\text{VIII:} \quad \widehat{X}_{ij} = 102.3 \times \alpha_i \times 3.20^{j\neq 1} \times 0.42^{j-1}, \\ &\text{with} \quad \vec{\alpha}' = (1, 0.99, 1.21, 1.47, 1.67, 1.56, 1.81, 1.64). \end{aligned} \tag{10.15}$$

Table 10.8 Average payments corresponding to the numbers of payments in Table 10.1.

Year of origin	Development year 1	2	3	4	5	6	7	8
2000	62	146	117	175	203	212	406	318
2001	133	122	96	379	455	441	429	
2002	148	232	120	481	312	390		
2003	119	185	223	171	162			
2004	93	109	87	190				
2005	33	129	176					
2006	237	179						
2007	191							

Here $j \neq 1$ should be read as a Boolean expression, with value 1 if true, 0 if false (in this case, for the special column with $j = 1$). Model IX leads to:

$$\text{IX:} \quad \widehat{X}_{ij} = 101.1 \times 1.10^{i-1} \times 3.34^{j\neq 1} \times 0.42^{j-1}. \tag{10.16}$$

10.4.2 Modeling claim sizes

The Poisson distribution with year of origin as well as year of development as explanatory variables, that is, the chain ladder method, is appropriate to model the number of claims. Apart from the numbers of claims given in Table 10.1, we also know the average claim size; it can be found in Table 10.8. For these claim sizes, the portfolio size, characterized by the factors α_i, is irrelevant. The inflation, hence the calendar year, is an important factor, and so is the development year, since only large claims tend to lead to delay in settlement. So for this situation, the separation models are more suitable. We have estimated the average claim sizes under the assumption that they arose from a gamma distribution with a constant coefficient of variation, with a multiplicative model.

The results for the various models are displayed in Table 10.9. As one sees, the nesting structure in the models is $7 \subset 6 \subset 4/5 \subset 3 \subset 2 \subset 1$; models 4 and 5 are both between 6 and 3, but they are not nested in one another. We have scaled the deviances in such a way that the fullest model 1 has a scaled deviance equal to the number of degrees of freedom, hence 15. This way, we can test the significance of the model refinements by comparing the gain in scaled deviance to the critical value of the χ^2 distribution with as a parameter the number of extra parameters estimated. A statistically significant step in both chains is the step from model 7 to 6. Taking the development parameters β_j arbitrary as in model 5, instead of geometric β^{j-1} as in model 6, does not significantly improve the fit. Refining model 6 to model 4 by introducing a parameter for inflation γ^{k-1} also does not lead to a significant improvement. Refining model 4 to model 3, nor model 3 to model 2, improves the fit significantly, but model 1 is significantly better than model 2. Still, we prefer

Table 10.9 Parameters, degrees of freedom and deviance for various models applied to the average claim sizes of Table 10.8.

Model	Parameters used	Df	Deviance	
1	$\mu, \alpha_i, \beta_j, \gamma_k$	15	15	(♡)
2	μ, β_j, γ_k	21	30.2	
3	$\mu, \beta_j, \gamma^{k-1}$	27	36.8	
4	$\mu, \beta^{j-1}, \gamma^{k-1}$	33	39.5	
5	μ, β_j	28	38.7	
6	μ, β^{j-1}	34	41.2	♡
7	μ	35	47.2	

the simple model 6, if only because model 6 is not dominated by model 1. This is because at the cost of 19 extra parameters, the gain in scaled deviance is only 26.2. So the best estimates are obtained from model 6. It gives an initial level of 129 in the first year of development, increasing to $129 \times 1.17^7 = 397$ in the eighth year. Notice that if the fit is not greatly improved by taking the coefficients γ_{i+j-1} arbitrary instead of geometric or constant, it is better either to ignore inflation or to use a fixed level, possibly with a break in the trend somewhere, just to avoid the problem of having to find extrapolated values of $\gamma_{t+1}, \ldots, \gamma_{2t}$.

By combining estimated average claim sizes by year of origin and year of development with the estimated claim numbers, see Table 10.7, we get the total amounts to be reserved. These are given in the rightmost column of Table 10.10. The corresponding model is found by combining both multiplicative models 6 and IX, see (10.16); it leads to the following estimated total payments:

$$6 \times \text{IX}: \quad \widehat{X}_{ij} = 13041 \times 1.10^{i-1} \times 3.34^{j \neq 1} \times 0.46^{j-1}. \tag{10.17}$$

This model can also be used if, as is usual in practice, one is not content with a square of observed and predicted values, but also wants estimates concerning these years of origin for development years after the one that has last been observed, hence a rectangle of predicted values. The total estimated payments for year of origin i are equal to $\sum_{j=1}^{\infty} \widehat{X}_{ij}$. Obviously, these are finite only if the coefficient for each development year in models 6 and IX combined is less than 1 in (10.17).

Remark 10.4.1 (Variance of the estimated IBNR totals)
To obtain a prediction interval for the estimates in practice, finding an estimate the variance of the IBNR totals is vital. If the model chosen is the correct one and the parameter estimates are unbiased, this variance reflects parameter uncertainty as well as volatility of the process. If we assume that in Table 10.7 the model is correct and the parameter estimates coincide with the actual values, the estimated row totals are estimates of Poisson random variables. As these random variables have a variance equal to this mean, and the yearly totals are independent, the total estimated process variance is equal to the total estimated mean, hence $0.8 + \cdots + 398.7 = 696.8 = 26.4^2$. If there is overdispersion present in the model, the variance must be multiplied by the estimated overdispersion factor. The actual vari-

Table 10.10 Observed and predicted total claims corresponding to the Tables 10.1 and 10.7. Under Total paid are the total payments made so far, under Total est., the estimated remaining payments.

Year of origin	Development year 1	2	3	4	5	6	7	8	Total paid	Total est.
2000	6262	22338	6084	2975	2842	636	1624	318|	43079	0
2001	13167	14762	7296	12128	4550	1323	429|	361	53655	361
2002	16280	42224	9600	9620	6552	780|	800	398	85056	1198
2003	19040	36445	18286	6498	3078|	1772	881	438	83347	3092
2004	14973	27686	7395	8740|	3926	1952	971	483	58794	7331
2005	6105	25929	15136|	8696	4324	2150	1069	532	47170	16771
2006	42186	46719|	19262	9578	4762	2368	1178	586	88905	37733
2007	32088|	42665	21215	10549	5245	2608	1297	645	32088	84224

ance of course also includes the variation of the estimated mean, but this is harder to come by. Again assuming that all parameters have been correctly estimated and that the model is also correct, including the independence of claim sizes and claim numbers, the figures in Table 10.10 are predictions for compound Poisson random variables with mean $\lambda\mu_2$. The parameters λ of the numbers of claims can be obtained from Table 10.7, the second moments μ_2 of the gamma distributed payments can be derived from the estimated means in (10.15) together with the estimated dispersion parameter. In Section 10.6, we describe a bootstrap method to estimate the predictive distribution. Also we derive a delta method based approximation for the prediction error. ∇

Remark 10.4.2 ('The' stochastic model behind chain ladder)
We have shown that the chain ladder method is just one algorithm to estimate the parameters of a simple GLM with two factors (year of origin and development year), a log-link and a mean-variance relationship of Poisson type ($\sigma^2 \propto \mu$). Mack (1993) describes as 'the' stochastic model behind chain ladder a different set of distributional assumptions under which doing these calculations makes sense. Aiming for a distribution-free model, he cannot specify a likelihood to be maximized, so he sets out to find minimum MSE unbiased linear estimators instead. His model does not require independence, but only makes some assumptions about conditional means and variances, given the past development for each year of origin. They are such that the unconditional means and variances of the incremental observations are the same as in the GLM. ∇

10.5 Solving IBNR problems by R

Since we have shown that the chain ladder and many other methods are actually GLMs, R's built-in function `glm` can do the necessary calculations. The dependent variable consists of a vector containing the elements of the triangle of the observed past losses, for example the aggregate payments in the past. These losses are broken

down by year of origin of the policy and year of development of the claim filing process, which act as explanatory variables. In other applications, we need the calendar year. In this section we show by a simple example how to get the triangular data into a usable vector form, as well as how to construct the proper row and column numbers conveniently. Then we show which `glm`-call can be used to produce the chain ladder estimates, and also how to implement Verbeek's method 10.3.1 to produce these same estimates.

First, we fill a one-dimensional array `Xij` with, stored row-wise, the 15 incremental observations from the triangle of Exercise 10.3.3. We also store the corresponding row and column numbers in vectors `i` and `j`.

```
Xij <- c(232,106,35,16,2, 258,115,56,27, 221,82,4, 359,71, 349)
  i <- c(  1,  1, 1, 1,1,   2,  2, 2, 2,   3, 3,3,   4, 4,   5)
  j <- c(  1,  2, 3, 4,5,   1,  2, 3, 4,   1, 2,3,   1, 2,   1)
```

In general, if we denote the width of the triangle by `TT`, the length of the vector `Xij` is `TT*(TT+1)/2`. The row numbers constitute a vector of ones repeated `TT` times, then twos repeated `TT-1` times, and so on until just the single number `TT`. The column numbers are the sequence `1:TT`, concatenated with `1:(TT-1)`, then `1:(TT-2)` and so on until finally just `1`. So from any vector `Xij` containing a runoff triangle, we can find `TT`, `i` and `j` as follows.

```
TT <- trunc(sqrt(2*length(Xij)))
i <- rep(1:TT,TT:1); j <- sequence(TT:1)
```

Now to apply the chain ladder method to this triangle, and to extract the parameter estimates for the α_i and β_j in (10.7), we simply call:

```
CL <- glm(Xij~as.factor(i)+as.factor(j), family=poisson)
coefs <- exp(coef(CL)) ##exponents of parameter estimates
alpha.glm <- coefs[1] * c(1, coefs[2:TT])
beta.glm <- c(1, coefs[(TT+1):(2*TT-1)])
```

The resulting values of the coefficients α_i and β_j in the vector `coefs` are:

```
> coefs
  (Intercept) as.factor(i)2 as.factor(i)3 as.factor(i)4 as.factor(i)5
     250.1441        1.1722        0.8315        1.2738        1.3952
as.factor(j)2 as.factor(j)3 as.factor(j)4 as.factor(j)5
       0.3495        0.1264        0.0791        0.0080
```

To apply Verbeek's algorithm 10.3.1 to find these same parameter estimates $\widehat{\alpha}_i$ and $\widehat{\beta}_j$, we need the row and column sums of the triangle. These sums over all observations sharing a common value of `i` and `j`, respectively, can be found by using the function `tapply` as below. First, `alpha` and `beta` are initialized to vectors of length `TT`. In the loop, we compute `alpha[n]` using (10.8) and add it to the auxiliary variable `aa` storing the sum of the α's computed so far. The last line produces the matrix of predicted values $\widehat{\alpha}_i\widehat{\beta}_j$ for all i, j, as the outer matrix product $\widehat{\alpha}\widehat{\beta}'$. So Verbeek's method 10.3.1 can be implemented as:

```
Ri <- tapply(Xij, i, sum); Cj <- tapply(Xij, j, sum)
alpha <- beta <- numeric(TT)
aa <- alpha[1] <- Ri[1]
bb <- beta[TT] <-  Cj[TT] / Ri[1]
for (n in 2:TT) {
   aa <- aa + (alpha[n] <- Ri[n]/(1-bb))
   bb <- bb + (beta[TT-n+1] <- Cj[TT-n+1] / aa)}
pred <- alpha %*% t(beta)
```

Using Verbeek's algorithm 10.3.1 instead of a call of `glm` to compute parameter estimates is quicker because no iterative process is needed. This is definitely an issue when many bootstrap simulations are done such as in Section 10.6. Also, it is slightly more general since it can also be applied when some of the observations are negative. To get non-negative parameter estimates, all row and column sums must be non-negative, as well as all sums over rectangles such as $\mathscr{A}_\Sigma$ in (10.1). Note that $\mathscr{A}_\Sigma \geq 0$ is not implied by non-negative marginals alone; to see this, consider a 2×2 triangle with $C_1 = R_1 = 1$, $C_2 = R_2 = 2$. Negative numbers in an IBNR-triangle occur in case recuperations, or corrections to case estimates that proved to be too pessimistic, are processed as if they were negative payments in a future development year.

To find the cumulated loss figures, it is convenient to store the IBNR data as a matrix, not as a long array. One straightforward way to construct a `TT` by `TT` square matrix containing the IBNR losses at the proper places, and next to construct the row-wise cumulated loss figures from it, is by doing

```
Xij.mat.cum <- Xij.mat <- matrix(0, nrow=TT, ncol=TT)
for (k in 1:length(Xij)) Xij.mat[i[k],j[k]] <- Xij[k]
for (k in 1:TT) Xij.mat.cum[k,] <- cumsum(Xij.mat[k,])
```

For a matrix `Xij.mat`, the row and column numbers can be found as the matrices `row(Xij.mat)` and `col(Xij.mat)` respectively. From these, it is easy to find the calendar years in which a loss occurred. It occurs in the future if its calendar year is past `TT`. To reconstruct the original long vector containing the past observations from the matrix representation, we have to take the transpose of `Xij.mat` before extracting the past elements from it, because `R` stores the elements of arrays in the so-called *column major order*, that is, not by rows but by columns.

```
i.mat <- row(Xij.mat); j.mat <- col(Xij.mat);
future <- i.mat + j.mat - 1 > TT
t(Xij.mat)[!t(future)] ## equals the vector Xij
```

10.6 Variability of the IBNR estimate

The aim of IBNR analysis is to make a prediction for how much remains to be paid on claims from the past. Earlier, we showed how to compute a point estimate of the outstanding claims, using the GLM that underlies well-known chain ladder method.

Point estimates are useful, especially if they have nice asymptotic properties such as the ones resulting from generalized linear models. But often we want to know prediction intervals for the outstanding claims, or for example the 95% quantile. Not only is there a process variance, since future claims constitute a multiple of a Poisson random variable in the chain ladder model, but additionally, there is parameter uncertainty. We can give standard deviations for all estimated coefficients, but from these, we cannot easily compute the variance around the estimated mean of the total outstanding claims. In two papers England and Verrall (1999) and England (2002) describe a method to obtain estimates of the prediction error. Based on the delta method, see Example 9.1.1, they give an approximation that can be derived using quantities produced by a `glm` call. It involves variances and covariances of the linear predictors and fitted values. They also give a bootstrapping method to estimate the reserve standard errors (the estimation error component of the prediction error). In a subsequent paper, England (2002) proposes not just using the bootstrap estimates to compute a standard deviation, but to actually generate a pseudo-sample of outcomes of the whole future process, in this way obtaining a complete approximate predictive distribution. From this, characteristics such as mean, variance, skewness and medians, as well as other quantiles, are easily derived.

As an example, we use the triangle of Taylor & Ashe (1983). This dataset with 55 incremental losses is used in many texts on IBNR problems.

```
Xij <- scan(n=55)
357848  766940  610542  482940 527326 574398 146342 139950 227229 67948
352118  884021  933894 1183289 445745 320996 527804 266172 425046
290507 1001799  926219 1016654 750816 146923 495992 280405
310608 1108250  776189 1562400 272482 352053 206286
443160  693190  991983  769488 504851 470639
396132  937085  847498  805037 705960
440832  847631 1131398 1063269
359480 1061648 1443370
376686  986608
344014
```

Based on these original data, we compute estimates $\widehat{\alpha}_i, \widehat{\beta}_j$ in a chain ladder model. For that, we invoke the `glm`-function with Poisson errors and log-link, and as covariates row and column numbers i and j (treated as factors). See also the preceding section. Actually, we take a quasi-Poisson error structure, as if the observations X_{ij} were ϕ times independent Poisson(μ_{ij}/ϕ) random variables, $i, j = 1, \ldots, t$. Here $\mu_{ij} = \alpha_i \beta_j$ for some positive parameters $\alpha_1, \ldots, \alpha_t$ and $\beta_1, \ldots, \beta_t$ with $\beta_1 = 1$. Construct fitted values $\widehat{\alpha}_i \widehat{\beta}_j$, $i, j = 1, \ldots, t$ and compute the sum of the future fitted values, as follows:

```
n <- length(Xij); TT <- trunc(sqrt(2*n))
i <- rep(1:TT, TT:1); i <- as.factor(i) ## row nrs
j <- sequence(TT:1); j <- as.factor(j)  ## col nrs
Orig.CL <- glm(Xij~i+j, quasipoisson)
coefs <- exp(as.numeric(coef(Orig.CL)))
alpha <- c(1, coefs[2:TT]) * coefs[1]
beta <- c(1, coefs[(TT+1):(2*TT-1)])
Orig.fits <- alpha %*% t(beta)
future <- row(Orig.fits) + col(Orig.fits) - 1 > TT
Orig.reserve <- sum(Orig.fits[future]) ## 18680856
```

10.6.1 Bootstrapping

England & Verrall (1999) describes a method to create bootstrap estimates. They are obtained by sampling (with replacement) from the observed residuals in the past observations to obtain a large set of pseudo-data, and computing an IBNR-forecast from it. The standard deviation of the set of reserve estimates obtained this way provides a bootstrap estimate of the estimation error. We will give each step of the method of England and Verrall, both the theoretical considerations and the R-implementation.

The Pearson X^2 statistic is the sum of the squared Pearson residuals. In the same way, the deviance can be viewed as the sum of squared deviance residuals, so the deviance residual is the square root of the contribution of an observation to the deviance, with the appropriate sign. See Section 11.4. Though the deviance residual is the natural choice in GLM contexts, in this case we will use the Pearson residual, since it is easy to invert:

$$r_{\mathrm{P}} = \frac{x-\mu}{\sqrt{\mu}}, \qquad \text{therefore} \qquad x = r_{\mathrm{P}}\sqrt{\mu} + \mu. \tag{10.18}$$

To calculate the outcomes of the Pearson residuals $(X_{ij} - \widehat{\mu}_{ij})/\sqrt{\widehat{\mu}_{ij}}$, do

```
Prs.resid <- (Xij-fitted(Orig.CL))/sqrt(fitted(Orig.CL))
```

The Pearson residual in (10.18) is unscaled in the sense that it does not include the scale parameter ϕ. This is not needed for the bootstrap calculations but only when computing the process error. To estimate ϕ, England and Verrall use the Pearson scale parameter. It uses a denominator $n-p$ instead of n to reduce bias:

$$\phi_{\mathrm{P}} = \frac{\sum r_{\mathrm{P}}^2}{n-p}, \tag{10.19}$$

where the summation is over all $n = t(t+1)/2$ past observations, and $p = 2t-1$ is the number of parameters estimated. In R, do

```
p <- 2*TT-1; phi.P <- sum(Prs.resid^2)/(n-p)
```

We adjust the residuals for bias in the same way as the scale parameter:

$$r'_{\mathrm{P}} = \sqrt{\frac{n}{n-p}}\, r_{\mathrm{P}}. \tag{10.20}$$

This is achieved as follows:

```
Adj.Prs.resid <- Prs.resid * sqrt(n/(n-p))
```

To be able to reproduce our results, we initialize the random number generator so as to get a fixed stream of random numbers:

```
set.seed(6345789)
```

Now run the bootstrap loop many times, for example, 1000 times.

```
nBoot <- 1000; payments <- reserves <- numeric(nBoot)
for (boots in 1:nBoot){ ## Start of bootstrap-loop
```

1. Resample from the adjusted residuals, with replacement:

   ```
   Ps.Xij <- sample(Adj.Prs.resid, n, replace=TRUE)
   ```

2. Using this set of residuals and the estimated values of $\widehat{\mu}_{ij}$, create a new suitable pseudo-history:

   ```
   Ps.Xij <- Ps.Xij * sqrt(fitted(Orig.CL)) + fitted(Orig.CL)
   Ps.Xij <- pmax(Ps.Xij, 0) ## Set 'observations' < 0 to 0
   ```

   For convenience, we set negative observations to zero. For the Taylor & Ashe example, about 0.16 negative pseudo-observations were generated in each bootstrap simulation, and setting them to zero obviously induces a slight bias in the results; in other triangles, this effect might be more serious. Note that to obtain feasible estimates $\widehat{\alpha}_i, \widehat{\beta}_j$, it is not necessary that all entries in the run-off triangle are non-negative, see Verbeek's algorithm 10.3.1 as well as Section 10.5. But this is required in the `glm`-routine for the `poisson` and `quasipoisson` families.
3. From this history, obtain estimates $\widehat{\alpha}_i, \widehat{\beta}_j$ using a chain ladder model:

   ```
   Ps.CL <- glm(Ps.Xij~i+j, quasipoisson)
   coefs <- exp(as.numeric(coef(Ps.CL)))
   Ps.alpha <- c(1, coefs[2:TT]) * coefs[1]
   Ps.beta <- c(1, coefs[(TT+1):(2*TT-1)])
   ```

4. Compute the fitted values, and use the sum of the future part as an estimate of the reserve to be held.

   ```
   Ps.fits <- Ps.alpha %*% t(Ps.beta)
   Ps.reserve <- sum(Ps.fits[future])
   ```

5. Then, sample from the estimated process distribution. In this case, this can be done by generating a single Poisson$\left(\sum \widehat{\alpha}_i \widehat{\beta}_j / \widehat{\phi}\right)$ random variable, with the sum taken over the future, and multiplying it by $\widehat{\phi}$:

   ```
   Ps.totpayments <- phi.P * rpois(1, Ps.reserve/phi.P)
   ```

6. At the end of the loop, store the simulated total payments and the estimated reserve to be held.

   ```
   reserves[boots] <- Ps.reserve
   payments[boots] <- Ps.totpayments
   }  ## Curly bracket indicates end of bootstrap-loop
   ```

The bootstrap reserve prediction error is computed as

$$\mathrm{PE}_{\mathrm{bs}}(R) = \sqrt{\phi_{\mathrm{P}} R + (\mathrm{SE}_{\mathrm{bs}}(R))^2}, \tag{10.21}$$

where R is a total reserve estimate (may also be for one origin year only), and $\mathrm{SE}_{\mathrm{bs}}(R)$ the bootstrap standard error of the reserve estimate, based on residuals that are adjusted for degrees of freedom as in (10.20). The process variance $\phi_{\mathrm{P}} R$ is added to the estimation variance.

```
PEbs <- sqrt(phi.P*Orig.reserve + sd(reserves)^2) ## 2882413
sd(reserves)^2 / (phi.P * Orig.reserve)           ## 7.455098
```

It proves that the estimation variance in this case is about 7.5 times the process variance. From the simulated values, one may compute various useful statistics. Differences with those given in England (2002) arose because we set negative observations to zero, but are largely due to randomness.

```
payments <- payments/1e6 ## expressed in millions
quantile(payments, c(0.5,0.75,0.9,0.95,0.99))
##       50%      75%      90%      95%      99%
##  18.56828 20.67234 22.35558 23.61801 26.19600
mean(payments)           ## 18.75786
sd(payments)             ## 2.873488
100 * sd(payments) / mean(payments)  ## 15.31885 = c.v. in %
pp <- (payments-mean(payments))/sd(payments)
sum(pp^3)/(nBoot-1)      ## 0.2468513 estimates the skewness
sum(pp^4)/(nBoot-1) - 3  ## 0.2701999 estimates the kurtosis
```

Our results are illustrated in the histogram in Figure 10.1. To the bars, representing fractions rather than frequencies, we added density estimates (the dashed one is a kernel density estimate, the dotted one just a fitted normal density), like this:

```
hist(payments,breaks=21,prob=TRUE)
lines(density(payments), lty="dashed")
curve(dnorm(x, mean = mean(payments), sd = sd(payments)),
   lty="dotted", add=TRUE)
```

Remark 10.6.1 (Caveats using IBNR methods)
If one thing can be learned from this whole exercise and the histogram in Figure 10.1, it is that for the future payments, one should not just give a point estimate prediction like 18.680856 million (the outcome of the chain ladder reserve estimate based on the original data). In a thousand pseudo-replications of the process, payments (in millions) ranged from 7.4 to 29.4, with quartiles 16.7 and 20.7, and one in ten fell outside the bounds (14.3, 23.6). The 'best estimate' is so inaccurate that to write it in any other way than 19 ± 3 million grossly overstates the precision of the prediction; all digits after the first two are insignificant.

Although some IBNR-methods have ardent supporters, it is very hard to substantiate a claim that any method is superior to the one described here; a lifetime is probably not long enough to prove this in practice. Other things should be considered, such as easy access to software, the possibility to generate convincing and

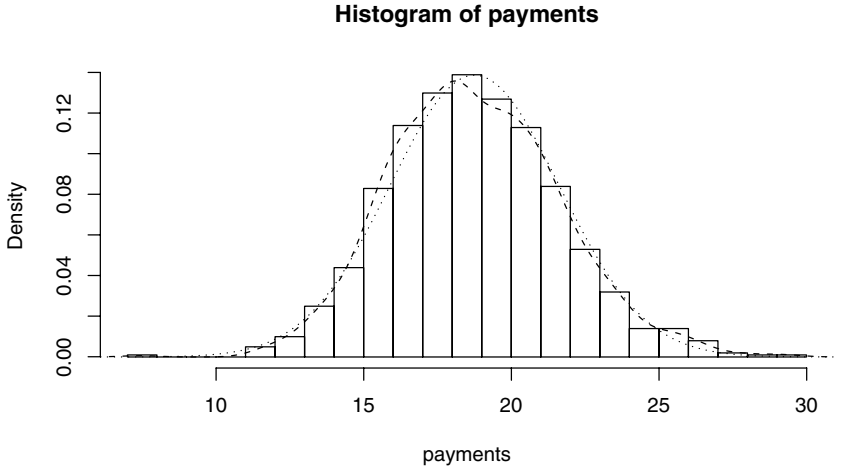

Fig. 10.1 Histogram of payments in millions, with estimated densities

revealing plots easily, control over what one is doing, and adaptability to new rules and situations. All this is not present when one uses a poorly documented black box method. As we have demonstrated here, `R` is a very suitable tool for the actuary, because it is powerful and closely follows the formulas of the theory. This makes checking, adapting and extending the code very easy. ∇

10.6.2 Analytical estimate of the prediction error

England and Verrall (1999) also provide a method to approximate the prediction error of an IBNR forecast. It does not need a bootstrap simulation run and uses only by-products stored in a `glm` object.

The mean squared error of the (unbiased) prediction $\widehat{X}_{ij}$ for each future observation X_{ij} can approximately be decomposed into one part estimation variance and another part process variance:

$$\begin{aligned} \mathrm{E}[(X_{ij}-\widehat{X}_{ij})^2] &= \left(\mathrm{E}[X_{ij}]-\mathrm{E}[\widehat{X}_{ij}]\right)^2 + \mathrm{Var}[X_{ij}-\widehat{X}_{ij}] \\ &\approx 0 + \mathrm{Var}[X_{ij}] + \mathrm{Var}[\widehat{X}_{ij}], \quad i+j-1>t. \end{aligned} \tag{10.22}$$

The squared bias is small and can be neglected as long as the estimators $\widehat{X}_{ij}$, even if not unbiased, are consistent predictors of X_{ij}. The future loss X_{ij} and its forecast $\widehat{X}_{ij}$ computed from past losses are independent random variables, so the variance of their difference is just the sum of their variances.

The process variance in the Poisson case is given by $\mathrm{Var}[X_{ij}] = \phi\mu_{ij}$. For the estimation variance, note that $\mathrm{E}[X_{ij}] = \mu_{ij} = \exp(\eta_{ij})$. Using the delta method, see Example 9.1.1, we see

$$\mathrm{Var}[\widehat{X}_{ij}] \approx \left|\frac{\partial\mu_{ij}}{\partial\eta_{ij}}\right|^2 \mathrm{Var}[\widehat{\eta}_{ij}]. \tag{10.23}$$

Since $\partial\mu/\partial\eta = \mu$ in case of log-link, the last two relations lead to the following approximation for the mean squared error of the prediction of future payment X_{ij}:

$$\mathrm{E}[(X_{ij} - \widehat{X}_{ij})^2] \approx \widehat{\phi}\,\widehat{\mu}_{ij} + \widehat{\mu}_{ij}^2\,\widehat{\mathrm{Var}}[\widehat{\eta}_{ij}], \quad i+j-1 > t. \tag{10.24}$$

In similar fashion we can show that if $\widehat{X}_{ij}$ and $\widehat{X}_{kl}$ are different estimated future payments, then

$$\mathrm{Cov}[\widehat{X}_{ij}, \widehat{X}_{kl}] \approx \widehat{\mu}_{ij}\,\widehat{\mu}_{kl}\,\widehat{\mathrm{Cov}}[\widehat{\eta}_{ij}, \widehat{\eta}_{kl}]. \tag{10.25}$$

Computing the MSE of the prediction $\widehat{R} = \sum\widehat{X}_{ij}$ for future totals $R = \sum X_{ij}$ leads to an expression involving estimated (co-)variances of the various linear predictors. Taking sums over 'future' (i,j) and $(i,j) \neq (k,l)$, we get:

$$\begin{aligned}
&\mathrm{E}[(R-\widehat{R})^2] \\
&\quad = \sum \mathrm{E}[(X_{ij} - \widehat{X}_{ij})^2] + \sum \mathrm{E}[(X_{ij} - \widehat{X}_{ij})(X_{kl} - \widehat{X}_{kl})] \\
&\quad \approx \sum \mathrm{E}[(X_{ij} - \widehat{X}_{ij})^2] + \sum \mathrm{Cov}[X_{ij} - \widehat{X}_{ij}, X_{kl} - \widehat{X}_{kl}] \\
&\quad = \sum \mathrm{E}[(X_{ij} - \widehat{X}_{ij})^2] + \sum \mathrm{Cov}[\widehat{X}_{ij}, \widehat{X}_{kl}] \\
&\quad \approx \sum \widehat{\phi}\,\widehat{\mu}_{ij} + \sum \widehat{\mu}_{ij}^2\,\widehat{\mathrm{Var}}[\widehat{\eta}_{ij}] + \sum \widehat{\mu}_{ij}\,\widehat{\mu}_{kl}\,\widehat{\mathrm{Cov}}[\widehat{\eta}_{ij}, \widehat{\eta}_{kl}] \\
&\quad = \sum \widehat{\phi}\,\widehat{\mu}_{ij} + \widehat{\mu}\,'\,\widehat{\mathrm{Var}}[\widehat{\eta}]\;\widehat{\mu}.
\end{aligned} \tag{10.26}$$

Here $\widehat{\mu}$ and $\widehat{\eta}$ are vectors of length t^2 containing all $\widehat{\mu}_{ij}$ and $\widehat{\eta}_{ij}$. This is one way to implement this using R:

```
Xij.1 <- xtabs(Xij~i+j) ## full square matrix
ii <- row(Xij.1); jj <- col(Xij.1); Xij.1 <- as.vector(Xij.1)
future <- as.numeric(ii+jj-1 > TT)
ii <- as.factor(ii); jj <- as.factor(jj)  ## are now vectors
Full.CL <- glm(Xij.1~ii+jj, fam=quasipoisson, wei=1-future)
Sig <- vcov(Full.CL); X <- model.matrix(Full.CL)
Cov.eta <- X%*%Sig%*%t(X)
mu.hat <- fitted(Full.CL)*future
pe2 <- phi.P * sum(mu.hat) + t(mu.hat) %*% Cov.eta %*% mu.hat
cat("Total reserve =", sum(mu.hat), "p.e. =", sqrt(pe2), "\n")
## Total reserve = 18680856 p.e. = 2945659
```

The use of a cross-tabulation of `Xij` by rows `i` and columns `j` is a quicker way to get from a triangle to a square than the one presented in Section 10.5. This square is stored as a vector of length `TT*TT`. An estimate of the covariance matrix of the $\vec{\eta} = \mathbf{X}\vec{\beta}$ is $\mathbf{X}\widehat{\Sigma}\mathbf{X}'$, where $\widehat{\Sigma}$ is the estimated variance matrix of the parameters, and $\mathbf{X}$ is the regression matrix (design matrix, model matrix). In `fitted(Full.CL)`,

predictions for future values are automatically included. The approximate prediction error is 2 945 659, the bootstrap prediction error is 2 882 413. To get predictions for just row `r`, simply set the entries of `mu.hat` for elements outside row `r` equal to zero. See Exercise 10.6.4.

10.7 An IBNR-problem with known exposures

In this section, we use `R` to tackle an IBNR problem with given *exposures*, that is, the number of policies n_i for each year of origin is assumed known. The dataset consists of a run-off triangle for a period of eight years; the total number of claims is `Xij`. To give an estimate of the claims that have yet to be reported, we read the data and compute the row and column numbers just as in Section 10.5.

```
Xij <- scan(n=36)
156  37   6   5   3   2   1   0
154  42   8   5   6   3   0
178  63  14   5   3   1
198  56  13  11   2
206  49   9   5
250  85  28
252  44
221
TT <- trunc(sqrt(2*length(Xij)))
i <- rep(1:TT, TT:1); j <- sequence(TT:1)
ni <- c(28950,29754,31141,32443,34700,36268,37032,36637)
```

Looking at the data, one sees that in the last year of origin, only 221 claims emerged in the first development year, which is appreciably fewer than the losses of the previous years, while the exposure is about the same. This number has no influence on the estimates for the development factors in a chain ladder method (the $\widehat{\beta}$-values), but it is proportional to the estimates of future losses in the final year of origin, so it has considerable influence on the resulting total reserve. See Exercise 10.7.2. This is why it might be better to forecast the future losses using the Bornhuetter-Ferguson method, where it is assumed that the forecasts are not, as with chain ladder, proportional to the row sums, but with other quantities deemed appropriate by the actuary. In this case, this boils down to not multiplying the predicted losses by the $\widehat{\alpha}_i$ factors estimated by the chain ladder method, but by the exposures n_i. A good first guess by the actuary would be that the loss ratios remain about the same, meaning in this case that the number of claims in total for each year of origin is proportional to the exposure. This leads to $M_i = n_i\widehat{\alpha}_1/n_1$ as a prior mean for the row total of losses corresponding to year i. In the Bornhuetter-Ferguson method, the row parameters used are those of the chain ladder method. So implementing this method, using Verbeek's algorithm 10.3.1 to find estimated parameters $\widehat{\alpha}_i$ and $\widehat{\beta}_j$, goes as follows.

```
Ri <- tapply(Xij, i, sum); Cj <- tapply(Xij, j, sum)
alpha <- beta <- numeric(TT)
aa <- alpha[1] <- Ri[1]
```

```
bb <- beta[TT] <- Cj[TT] / Ri[1]
for (n in 2:TT) {
aa <- aa + (alpha[n] <- Ri[n]/(1-bb))
bb <- bb + (beta[TT-n+1] <- Cj[TT-n+1] / aa)}
Mi <- ni * alpha[1] / ni[1]
BF <- Mi %*% t(beta); CL <- alpha %*% t(beta)
future <- row(BF) + col(BF) - 1 > TT
rowSums(BF * future) ## 0.0  0.0  0.5  2.6  6.4 13.2 26.3 76.9
rowSums(CL * future) ## 0.0  0.0  0.6  3.1  7.0 19.2 32.1 90.0
```

The row sums of the future part of the square matrix $\vec{M}\widehat{\beta}'$ represent the estimated total numbers of losses by year of origin. So the reserve estimates for the chain ladder method turn out to be somewhat bigger than those for Bornhuetter-Ferguson. Contrary to what was expected earlier, this also holds for the final year.

One way to describe the chain ladder method is to construct vectors $\widehat{\alpha}$ and $\widehat{\beta}$ (with $\beta_\Sigma = 1$) in such a way that the deviance (9.29) between the data and the 'past' part (upper left triangle) of matrix $\widehat{\alpha}\widehat{\beta}'$ is minimized, and to use the 'future' part of this matrix to determine reserves. For Bornhuetter-Ferguson, use the future part of $\vec{M}\widehat{\beta}'$ instead. It can be shown, see Verrall (2004), that both these methods arise as extreme cases in a Bayesian framework with generalized linear models, with a loose prior for chain ladder and a tight one for Bornhuetter-Ferguson.

In the Bornhuetter-Ferguson method, it is assumed that the effect of the year of origin on the losses in the chain ladder method is captured by some external quantity resembling the portfolio growth, represented in our case by the externally given exposure vector $\vec{n}$. So, instead of $\widehat{\alpha}_i\widehat{\beta}_j$, with $\beta_\Sigma = 1$, the mean for cell (i, j) is estimated as $n_i\widehat{\beta}_j$ with $\widehat{\beta}$ found by the chain ladder method. Evidently, we may get a closer fit to the past data by choosing a different $\widehat{\beta}$. Still assuming the losses to have a (quasi-)Poisson distribution, we get a generalized linear model with about half the number of parameters of the chain ladder method. The fit to the observed data will be worse than with the chain ladder method (which uses optimal $\widehat{\alpha}$ as well as $\widehat{\beta}$), but we can easily judge by a deviance analysis if the fit is significantly worse.

To estimate the β-values using `glm`, note that the linear predictor $\eta_{ij} = \log n_i + \log\beta_j$ has a 'fixed' component $\log n_i$, which has to be included with a coefficient 1 for each observation in row i. This can be achieved by using the offset mechanism, see Section 9.5. The following R program achieves the fitting:

```
Expo <- ni[i] ## Exposures with each element of Xij
CLi <- glm(Xij~as.factor(i)+as.factor(j), poisson)
CLoff <- glm(Xij~offset(log(Expo))+as.factor(j), poisson)
```

The residual deviance with chain ladder is 34.2 on 21 degrees of freedom, as opposed to 62.0 on 28 with the offset model. Note that the data having an ordinary Poisson structure instead of an overdispersed one (with $\phi > 1$) is not far-fetched; the unscaled deviance equals the $\chi^2(21)$ critical value at level 96.5%. The $\chi^2(7)$ critical value at level 95% being 14.1, we conclude that the data exhibit more change with year of origin than just growth in proportion to the number of contracts. The exposure model is of course a restriction of the chain ladder model, with $\alpha_i = n_i/n_1$ fixed.

The reader might try other models, by including a calendar year effect, geometric or general, or a geometric effect of year of origin on top of the exposure.

10.8 Exercises

Section 10.1

1. In how many ways can the data in Table 10.1 be organized in a table, by year of origin, year of development and calendar year, vertically or horizontally, in increasing or decreasing order?

Section 10.2

1. Apply the chain ladder method to the given IBNR triangle with cumulated figures. What could be the reason why run-off triangles to be processed through the chain ladder method are usually given in a cumulated form?

Year of origin	Development year 1	2	3	4	5
1	232	338	373	389	391
2	258	373	429	456	
3	221	303	307		
4	359	430			
5	349				

Section 10.3

1. Prove (10.11). What is the mode of the random variables X_{ij} in model (10.10)?
2. Apply the arithmetic separation method to the data of Exercise 10.2.1. Determine the missing γ values by linear or by loglinear interpolation, whichever seems more appropriate.
3. Which distance between data and predicted values is minimized by the chain ladder method? Which by the separation methods?
4. Why is it not an improvement of the model to use a model $\alpha_i\beta_j\gamma^k$ rather than only $\alpha_i\beta_j$? Making use of `R` and (10.12), and also by the method of successive substitution, see Section 9.3, verify if the results in De Vylder (1978) have been computed and printed correctly (note that he did only a few iterations). Using the former method, is it a problem to estimate the model including the calendar year parameters? Hint: there is a complication because `R` asks for starting values if you use `glm(...,family=gaussian(link=log),...)`. Such starting values are supplied, for example, by specifying `mustart=mu.st` as a parameter, where `mu.st` contains the fitted values of a Poisson fit with log-link, or of a Gaussian fit with standard (identity) link.

Section 10.4

1. Verify that the same predictions (10.14) are obtained from the models $\mathrm{E}[X_{ij}] = \mu\alpha_i\beta^{j-1}$ and $\mathrm{E}[X_{ij}] = \mu\alpha_i\gamma^{i+j-2}$.
2. Argue why in model I, where for $i, j = 1, \ldots, t$, we have $\mathrm{E}[X_{ij}] = \mu\alpha_i\beta_j\gamma_{i+j-1}$, the parameter γ_t can be taken equal to 1 without loss of generality, meaning that for $t = 8$, model I has only six more parameters to be estimated than model II. Verify that with model I there are $3(t-1)$ parameters to be estimated from $t(t+1)/2$ observations, so model I makes sense only if $t > 3$.
3. Explain why models IV and V are equivalent.
4. For $i = j = 1, 3, 5, 7$, compute the values predicted by models (10.15) and (10.16), and compare these to the actual observations.
5. Verify (10.17). Use it to determine $\sum_{j=1}^{\infty} \widehat{X}_{ij}$.
6. Reproduce Table 10.10. Compare with a direct (quasi-)Poisson model instead of a two-stage model.

Section 10.5

1. Using for example `help("%*%")` to get inspiration, compare the results of the following calls that are candidates to produce the square of fitted values of a `glm`-call for the chain ladder method, and comment:

```
fitted(CL)
alpha[i]*beta[j]
alpha*beta
alpha%o%beta
alpha%*%beta
outer(alpha,beta)
alpha%*%t(beta)
```

   Find parameters `mu`, `alpha[1:TT]` and `beta[1:TT]`, with `alpha[1] = beta[1] = 1`, but such that they still lead to the same predictions `mu*alpha[i]*beta[j]`.
   Also, find equivalent parameter vectors such that the sum of the `beta` elements equals 1.
2. The calendar year corresponding to an observation can be computed simply as `k <- i+j-1`. Using an appropriate call of `glm`, apply the Arithmetic Separation method (10.9) to the data of Exercise 10.2.4. To generate fitted values for the lower triangle of the IBNR-data, plot the coefficients corresponding to the calendar years stored in an array `gamma.sep` by using the call `plot(log(gamma.sep))`, and extrapolate the γ_k-values geometrically. Then generate fitted values for the full IBNR-square.
3. Compute the parameter estimates for the Arithmetic Separation method through a method analogous to Verbeek's algorithm 10.3.1, involving sums over diagonals. Compare the resulting fits with those of the previous exercise.
4. The Hoerl-curve gives a functional form for the β_j parameters, having $\beta_j = \exp(\gamma j + \delta \log j)$ for some real numbers γ and δ. These can be used for all rows in common, or for each row separately (interaction). Apply this to the example in Table 10.1, and test for significance.

Section 10.6

1. Using the method described in this section, construct a predictive distribution of the IBNR-reserve to be held using a gamma error distribution instead of Poisson. Compare the resulting histograms, as well as the first three estimated cumulants of the resulting distributions. [Apply `glm` to the `Gamma` family with a log-link. Also, check England and Verrall (1999) for when $V(\mu) = \mu^2$ instead of $V(\mu) = \mu$ should be taken into account. Special care should be taken with the random drawing from the future claims. A gamma random variable has to be generated for each cell, rather than for all cells combined such as was possible in the quasi-Poisson case. The final results of the procedure should not deviate too much from the Poisson results, see England and Verrall (1999), Tables 1 and 2.]
2. Repeat the Poisson bootstrapping but now using Verbeek's algorithm 10.3.1 instead of calling `glm`. By calling `Sys.time`, verify that about 20 bootstrap runs can be processed per second using `glm`, and indicate how much faster Verbeek's algorithm 10.3.1 makes the procedure. Also investigate to how much bias the setting zero of negative observations leads, when compared with rejecting the full triangle in case any row or column sum proves to be negative.
3. England and Verrall (1999, formula (3.4)) also compute the contribution to the total prediction error of each separate year of origin i. Essentially, (3.4) equals (3.5) with the summations restricted to the corresponding row of the IBNR predicted triangle. This can be easily achieved by using the same code used for implementing formula (3.5), but now with the $\widehat{\mu}_{kj}$ replaced by zero for predictions of origin years with $k \neq i$.
   Reproduce the third column of Tables 1 and 2 in England and Verrall (1999), p. 288.

Section 10.7

1. What happens if both `offset(log(Expo))` and `as.factor(i)` are included as a model term in the example of this section?
2. Find out what happens with the reserves according to chain ladder and Bornhuetter-Ferguson if the number 221 in the last row is replaced by 180 or by 250.
3. Compute the past and future fitted totals in the chain ladder model for the data of this section; you should get the values 2121 and 152.0312. Why does the total of the fitted values for past observations equal `sum(Xij)`? Is this also the case when the factor year of origin in the `glm` call is replaced by the offset term?
4. The cause of the fit of the chain ladder model being rejected at the 5% level (the deviance is 34.2 on 21 df, the critical value `qchisq(0.95,df=21)` is 32.67057) might be that there is an effect in the data that is due to the calendar year. Fit a model that incorporates all three time effects (origin year, development year, calendar year), see also Section 10.3. Do an analysis-of-deviance on the two models: compare the difference in the deviances with an appropriate χ^2-critical value to judge if adding the extra set of covariates (calendar year as a factor) is worthwhile, in the sense that the fit is improved significantly.
   A problem with incorporating a calendar year effect is that for the lower right triangle, the calendar year effect cannot be estimated, since no data about that calendar year are available. One way to deal with this problem is to extrapolate the sequence of past calendar year parameters to the future, for example using linear extrapolation, by fitting a straight line through the points (i, γ_i), or geometrically.
5. In policy year 6, the terms of the policies were a little more consumer friendly, while in calendar year 7, the judges were somewhat more lenient. Inspecting the α_i and γ_k (for the calendar year) estimates confirms this; note that the notion that these years are different from the rest should *not* have been inspired by a peek at the estimation results. For that reason, treat these

two years separately from the rest. So try a model that is a restricted version of the chain ladder model in the sense that the effect of year of origin is captured by the known portfolio sizes except for year 6. Also try a model in which the calendar year parameters are restricted to γ_7 arbitrary, $\gamma_k \equiv 1$ otherwise. To estimate the former model, all that has to be done is to replace `as.factor(i)` in the chain ladder model by `(i==6)` as a model term. Note the double equality sign in a logical expression; the brackets are also necessary.