Credibility theory features of **actuar**

Christophe Dutang
ISFA, Université Claude Bernard Lyon 1

Vincent Goulet
École d'actuariat, Université Laval

Mathieu Pigeon
École d'actuariat, Université Laval

1 Introduction

Credibility models are actuarial tools to distribute premiums fairly among an heterogeneous group of policyholders (henceforth called *entities*). More generally, they can be seen as prediction methods applicable in any setting where repeated measures are made for subjects with different risk levels.

The credibility theory facilities of **actuar** consist of one data set and two main functions:

1. matrix `hachemeister` containing the famous data set of Hachemeister (1975);
2. function `simpf` to simulate data from compound hierarchical models;
3. function `cm` to fit hierarchical and regression credibility models.

2 Hachemeister data set

The data set of Hachemeister (1975) consists of private passenger bodily injury insurance average claim amounts, and the corresponding number of claims, for five U.S. states over 12 quarters between July 1970 and June 1973. The data set is included in the package in the form of a matrix with 5 rows and 25 columns. The first column contains a state index, columns 2–13 contain the claim averages and columns 14–25 contain the claim numbers:

```
> data(hachemeister)
> hachemeister
```

```
     state ratio.1 ratio.2 ratio.3 ratio.4 ratio.5 ratio.6
[1,]     1    1738    1642    1794    2051    2079    2234
[2,]     2    1364    1408    1597    1444    1342    1675
[3,]     3    1759    1685    1479    1763    1674    2103
[4,]     4    1223    1146    1010    1257    1426    1532
[5,]     5    1456    1499    1609    1741    1482    1572
     ratio.7 ratio.8 ratio.9 ratio.10 ratio.11 ratio.12
[1,]    2032    2035    2115     2262     2267     2517
[2,]    1470    1448    1464     1831     1612     1471
[3,]    1502    1622    1828     2155     2233     2059
[4,]    1953    1123    1343     1243     1762     1306
[5,]    1606    1735    1607     1573     1613     1690
     weight.1 weight.2 weight.3 weight.4 weight.5 weight.6
[1,]     7861     9251     8706     8575     7917     8263
[2,]     1622     1742     1523     1515     1622     1602
[3,]     1147     1357     1329     1204      998     1077
[4,]      407      396      348      341      315      328
[5,]     2902     3172     3046     3068     2693     2910
     weight.7 weight.8 weight.9 weight.10 weight.11 weight.12
[1,]     9456     8003     7365      7832      7849      9077
[2,]     1964     1515     1527      1748      1654      1861
[3,]     1277     1218      896      1003      1108      1121
[4,]      352      331      287       384       321       342
[5,]     3275     2697     2663      3017      3242      3425
```

3 Portfolio simulation

Function `simpf` simulates portfolios of data following compound models of the form

$$S = C_1 + \cdots + C_N, \tag{1}$$

where $C_1, C_2, \dots$ are mutually independent and identically distributed random variables each independent from N. Both the frequency and the severity components can have a hierarchical structure. The main characteristic of hierarchical models is to have the probability law at some level in the classification structure be conditional on the outcome in previous levels. For example, consider the following compound hierarchical model:

$$S_{ijt} = C_{ijt1} + \cdots + C_{ijtN_{ijt}}, \tag{2}$$

for $i = 1, \dots, I$, $j = 1, \dots, J_i$, $t = 1, \dots, n_{ij}$ and with

$$\begin{aligned} N_{ijt}|\Lambda_{ij}, \Phi_i &\sim \text{Poisson}(w_{ijt}\Lambda_{ij}) & C_{ijtu}|\Theta_{ij}, \Psi_i &\sim \text{Lognormal}(\Theta_{ij}, 1) \\ \Lambda_{ij}|\Phi_i &\sim \text{Gamma}(\Phi_i, 1) & \Theta_{ij}|\Psi_i &\sim N(\Psi_i, 1) \\ \Phi_i &\sim \text{Exponential}(2) & \Psi_i &\sim N(2, 0.1). \end{aligned} \tag{3}$$

The random variables Φ_i, Λ_{ij}, Ψ_i and Θ_{ij} are generally seen as risk parameters in the actuarial literature. The w_{ijt}s are known weights, or volumes. Using as convention to number the data level 0, the above is a two-level hierarchical model.

Function `simpf` is presented in the credibility theory section because it was originally written in this context, but it has much wider applications. For instance, it is used by `aggregateDist` for the simulation of the aggregate claim amount random variable.

Goulet and Pouliot (2007) describe in detail the model specification method used in `simpf`. For the sake of completeness, we briefly outline this method here.

A hierarchical model is completely specified by the number of nodes at each level (I, $J_1, \dots, J_I$ and $n_{11}, \dots, n_{IJ}$, above) and by the probability laws at each level. The number of nodes is passed to `simpf` by means of a named list where each element is a vector of the number of nodes at a given level. Vectors are recycled when the number of nodes is the same throughout a level. Probability models are expressed in a semi-symbolic fashion using an object of mode `"expression"`. Each element of the object must be named — with names matching those of the number of nodes list — and should be a complete call to an existing random number generation function, with the number of variates omitted. Hierarchical models are achieved by replacing one or more parameters of a distribution at a given level by any combination of the names of the levels above. If no mixing is to take place at a level, the model for this level can be `NULL`.

Function `simpf` also supports usage of weights in models. These usually modify the frequency parameters to take into account the "size" of an entity. Weights are used in simulation wherever the name `weights` appears in a model.

Hence, function `simpf` has four main arguments: 1) `nodes` for the number of nodes list; 2) `model.freq` for the frequency model; 3) `model.sev` for the severity model; 4) `weights` for the vector of weights in lexicographic order, that is all weights of entity 1, then all weights of entity 2, and so on.

For example, assuming that $I = 2$, $J_1 = 4$, $J_2 = 3$, $n_{11} = \cdots = n_{14} = 4$ and $n_{21} = n_{22} = n_{23} = 5$ in model (3) above, and that weights are simply simulated from a uniform distribution on $(0.5, 2.5)$, then simulation of a data set with `simpf` is achieved with:

```
> wijt <- runif(31, 0.5, 2.5)
> nodes <- list(cohort = 2, contract = c(4, 3), year = c(4,
+     4, 4, 4, 5, 5, 5))
> mf <- expression(cohort = rexp(2), contract = rgamma(cohort,
+     1), year = rpois(weights * contract))
> ms <- expression(cohort = rnorm(2, sqrt(0.1)),
+     contract = rnorm(cohort, 1), year = rlnorm(contract,
+         1))
> pf <- simpf(nodes = nodes, model.freq = mf, model.sev = ms,
```

```
+       weights = wijt)
```

The function returns the variates in a two-dimension list of class "simpf" containing all the individual claim amounts for each entity. Such an object can be seen as a three-dimension array with a third dimension of potentially varying length. The function also returns a matrix of integers giving the classification indexes of each entity in the portfolio (subscripts i and j in the notation above). Displaying the complete content of the object returned by simpf can be impractical. For this reason, the print method for this class only prints the simulation model and the number of claims in each node:

```
> pf

Portfolio of claim amounts

  Frequency model
    cohort   ~ rexp(2)
    contract ~ rgamma(cohort, 1)
    year     ~ rpois(weights * contract)
  Severity model
    cohort   ~ rnorm(2, sqrt(0.1))
    contract ~ rnorm(cohort, 1)
    year     ~ rlnorm(contract, 1)

  Number of claims per node:

     cohort contract year.1 year.2 year.3 year.4 year.5
[1,]      1        1      2      2      1      0     NA
[2,]      1        2      0      0      0      0     NA
[3,]      1        3      0      3      0      2     NA
[4,]      1        4      0      1      1      1     NA
[5,]      2        1      0      1      1      1      2
[6,]      2        2      0      0      0      0      0
[7,]      2        3      3      4      2      2      0
```

The package defines methods for four generic functions to easily access key quantities of the simulated portfolio.

1. By default, the method of aggregate returns the values of aggregate claim amounts S_{ijt} in a regular matrix (subscripts i and j in the rows, subscript t in the columns). The method has a by argument to get statistics for other groupings and a FUN argument to get statistics other than the sum:

   ```
   > aggregate(pf)
   ```

```
     cohort contract year.1 year.2 year.3 year.4 year.5
[1,]      1        1  31.37  7.521 11.383  0.000     NA
[2,]      1        2   0.00  0.000  0.000  0.000     NA
[3,]      1        3   0.00 72.706  0.000 23.981     NA
[4,]      1        4   0.00 98.130 50.622 55.705     NA
[5,]      2        1   0.00 11.793  2.253  2.397  10.48
[6,]      2        2   0.00  0.000  0.000  0.000   0.00
[7,]      2        3  44.81 88.737 57.593 14.589   0.00
```

```
> aggregate(pf, by = c("cohort", "year"), FUN = mean)
```

```
     cohort year.1 year.2 year.3 year.4 year.5
[1,]      1  15.69  29.73  31.00 26.562     NA
[2,]      2  14.94  20.11  19.95  5.662  5.238
```

2. The method of `frequency` returns the number of claims N_{ijt}. It is a wrapper for `aggregate` with the default `sum` statistic replaced by `length`. Hence, arguments `by` and `FUN` remain available:

```
> frequency(pf)
```

```
     cohort contract year.1 year.2 year.3 year.4 year.5
[1,]      1        1      2      2      1      0     NA
[2,]      1        2      0      0      0      0     NA
[3,]      1        3      0      3      0      2     NA
[4,]      1        4      0      1      1      1     NA
[5,]      2        1      0      1      1      1      2
[6,]      2        2      0      0      0      0      0
[7,]      2        3      3      4      2      2      0
```

```
> frequency(pf, by = "cohort")
```

```
     cohort freq
[1,]      1   17
[2,]      2   16
```

3. The method of `severity` (a generic function introduced by the package) returns the individual claim amounts C_{ijtu} in a matrix similar to those above, but with a number of columns equal to the maximum number of observations per entity,

$$\max_{i,j} \sum_{t=1}^{n_{ij}} N_{ijt}.$$

Thus, the original period of observation (subscript t) and the identifier of the severity within the period (subscript u) are lost and each variate now constitute a "period" of observation. For this reason, the method provides an argument `splitcol` in case one would like to extract separately the individual severities of one or more periods:

```
> severity(pf)

$first
     cohort contract claim.1 claim.2 claim.3 claim.4 claim.5
[1,]      1        1   7.974  23.401   3.153   4.368  11.383
[2,]      1        2      NA      NA      NA      NA      NA
[3,]      1        3   3.817  41.979  26.910   4.903  19.078
[4,]      1        4  98.130  50.622  55.705      NA      NA
[5,]      2        1  11.793   2.253   2.397   9.472   1.004
[6,]      2        2      NA      NA      NA      NA      NA
[7,]      2        3  14.322  11.522  18.966  33.108  15.532
     claim.6 claim.7 claim.8 claim.9 claim.10 claim.11
[1,]      NA      NA      NA      NA       NA       NA
[2,]      NA      NA      NA      NA       NA       NA
[3,]      NA      NA      NA      NA       NA       NA
[4,]      NA      NA      NA      NA       NA       NA
[5,]      NA      NA      NA      NA       NA       NA
[6,]      NA      NA      NA      NA       NA       NA
[7,]   14.99   25.11   40.15   17.44    4.426    10.16

$last
NULL

> severity(pf, splitcol = 1)

$first
     cohort contract claim.1 claim.2 claim.3 claim.4 claim.5
[1,]      1        1   3.153   4.368  11.383      NA      NA
[2,]      1        2      NA      NA      NA      NA      NA
[3,]      1        3   3.817  41.979  26.910   4.903  19.078
[4,]      1        4  98.130  50.622  55.705      NA      NA
[5,]      2        1  11.793   2.253   2.397   9.472   1.004
[6,]      2        2      NA      NA      NA      NA      NA
[7,]      2        3  33.108  15.532  14.990  25.107  40.150
     claim.6 claim.7 claim.8
[1,]      NA      NA      NA
[2,]      NA      NA      NA
[3,]      NA      NA      NA
[4,]      NA      NA      NA
[5,]      NA      NA      NA
[6,]      NA      NA      NA
[7,]   17.44   4.426   10.16

$last
     cohort contract claim.1 claim.2 claim.3
[1,]      1        1   7.974   23.40      NA
```

```
[2,]      1         2      NA      NA      NA
[3,]      1         3      NA      NA      NA
[4,]      1         4      NA      NA      NA
[5,]      2         1      NA      NA      NA
[6,]      2         2      NA      NA      NA
[7,]      2         3  14.322   11.52   18.97
```

4. The method of `weights` extracts the weights matrix from a simulated data set:

```
> weights(pf)

     cohort contract year.1 year.2 year.3 year.4 year.5
[1,]      1        1 0.8361  2.115 1.2699 1.1555     NA
[2,]      1        2 1.7042  1.709 0.7493 1.0892     NA
[3,]      1        3 1.6552  1.762 1.5240 1.5100     NA
[4,]      1        4 1.5681  1.614 2.2358 2.1594     NA
[5,]      2        1 0.7229  1.907 2.2950 1.0595 0.9564
[6,]      2        2 0.5307  0.758 0.6868 0.9738 2.0823
[7,]      2        3 1.6995  2.320 1.6208 2.0114 1.2583
```

In addition, all methods have a `classification` and a `prefix` argument. When the first is `FALSE`, the classification index columns are omitted from the result. The second argument overrides the default column name prefix; see the `simpf.summaries` help page for details.

Function `simpf` was used to simulate the data in Forgues et al. (2006).

4 Hierarchical credibility model

The linear model fitting function of R is named `lm`. Since credibility models are very close in many respects to linear models, and since the credibility model fitting function of **actuar** borrows much of its interface from `lm`, we named the credibility function `cm`.

Function `cm` acts as a unified interface for all credibility models supported by the package. Currently, these are the unidimensional models of Bühlmann (1969) and Bühlmann and Straub (1970), the hierarchical model of Jewell (1975) (of which the first two are special cases) and the regression model of Hachemeister (1975). The modular design of `cm` makes it easy to add new models if desired.

This subsection concentrates on usage of `cm` for hierarchical models.

There are some variations in the formulas of the hierarchical model in the literature. We compute the credibility premiums as given in Bühlmann and Jewell (1987) or Bühlmann and Gisler (2005). We support three types of estimators of the between variance structure parameters: the unbiased estimators of Bühlmann and Gisler (2005) (the default), the slightly different version of Ohlsson (2005) and the iterative pseudo-estimators as found in Goovaerts and Hoogstad (1987) or Goulet (1998). For instance, for a two-level hierarchical model like (3), the best linear prediction for year $n+1$ based on ratios $X_{ijt} = S_{ijt}/w_{ijt}$ is

$$\begin{aligned} \hat{\pi}_{ij} &= z_{ij} X_{ijw} + (1 - z_{ij})\hat{\pi}_i \\ \hat{\pi}_i &= z_i X_{izw} + (1 - z_i) m \end{aligned} \tag{4}$$

with

$$z_{ij} = \frac{w_{ij\Sigma}}{w_{ij\Sigma} + s^2/a}, \qquad X_{ijw} = \sum_{t=1}^{n_{ij}} \frac{w_{ijt}}{w_{ij\Sigma}} X_{ijt}$$

$$z_i = \frac{z_{i\Sigma}}{z_{i\Sigma} + a/b}, \qquad X_{izw} = \sum_{j=1}^{J_i} \frac{z_{ij}}{z_{i\Sigma}} X_{ijw}.$$

The estimator of s^2 is

$$\hat{s}^2 = \frac{1}{\sum_{i=1}^{I} \sum_{j=1}^{J_i} (n_{ij} - 1)} \sum_{i=1}^{I} \sum_{j=1}^{J_i} \sum_{t=1}^{n_{ij}} w_{ijt} (X_{ijt} - X_{ijw})^2. \tag{5}$$

The three types of estimators for parameters a and b are the following. First, let

$$A_i = \sum_{j=1}^{J_i} w_{ij\Sigma}(X_{ijw} - X_{iww})^2 - (J_i - 1)s^2 \qquad c_i = w_{i\Sigma\Sigma} - \sum_{j=1}^{J_i} \frac{w_{ij\Sigma}^2}{w_{i\Sigma\Sigma}}$$

$$B = \sum_{i=1}^{I} z_{i.}(X_{izw} - \bar{X}_{zzw})^2 - (I - 1)a \qquad d = z_{\Sigma\Sigma} - \sum_{i=1}^{I} \frac{z_{i\Sigma}^2}{z_{\Sigma\Sigma}},$$

with

$$\bar{X}_{zzw} = \sum_{i=1}^{I} \frac{z_{i\Sigma}}{z_{\Sigma\Sigma}} X_{izw}. \tag{6}$$

(Hence, $\mathrm{E}[A_i] = c_i a$ and $\mathrm{E}[B] = db$.) Then, the Bühlmann–Gisler estimators are

$$\hat{a} = \frac{1}{I} \sum_{i=1}^{I} \max\left(\frac{A_i}{c_i}, 0\right) \tag{7}$$

$$\hat{b} = \max\left(\frac{B}{d}, 0\right), \tag{8}$$

the Ohlsson estimators are

$$\hat{a}' = \frac{\sum_{i=1}^{I} A_i}{\sum_{i=1}^{I} c_i} \tag{9}$$

$$\hat{b}' = \frac{B}{d} \tag{10}$$

and the iterative (pseudo-)estimators are

$$\tilde{a} = \frac{1}{\sum_{i=1}^{I}(J_i - 1)} \sum_{i=1}^{I} \sum_{j=1}^{J_i} z_{ij}(X_{ijw} - X_{izw})^2 \tag{11}$$

$$\tilde{b} = \frac{1}{I-1} \sum_{i=1}^{I} z_i (X_{izw} - X_{zzw})^2, \tag{12}$$

where

$$X_{zzw} = \sum_{i=1}^{I} \frac{z_i}{z_\Sigma} X_{izw}. \tag{13}$$

Note the difference between the two weighted averages (6) and (13). See Goulet and Ouellet (2007) for further discussion on this topic.

Finally, the estimator of the collective mean m is $\hat{m} = X_{zzw}$.

The credibility modeling function `cm` assumes that data is available in the format most practical applications would use, namely a rectangular array (matrix or data frame) with entity observations in the rows and with one or more classification index columns (numeric or character). One will recognize the output format of `simpf` and its summary methods.

Then, function `cm` works much the same as `lm`. It takes in argument a formula of the form `~ terms` describing the hierarchical interactions in a data set; the data set containing the variables referenced in the formula; the names of the columns where the ratios and the weights are to be found in the data set. The latter should contain at least two nodes in each level and more than one period of experience for at least one entity. Missing values are represented by `NA`s. There can be entities with no experience (complete lines of `NA`s).

In order to give an easily reproducible example, we group states 1 and 3 of the Hachemeister data set into one cohort and states 2, 4 and 5 into another. This shows that data does not have to be sorted by level. The fitted model using the iterative estimators is:

```
> X <- cbind(cohort = c(1, 2, 1, 2, 2), hachemeister)
> fit <- cm(~cohort + cohort:state, data = X, ratios = ratio.1:ratio.12,
+     weights = weight.1:weight.12, method = "iterative")
> fit

Call:
cm(formula = ~cohort + cohort:state, data = X, ratios = ratio.1:ratio.12,
    weights = weight.1:weight.12, method = "iterative")

Structure Parameters Estimators

  Collective premium: 1746
```

```
  Between cohort variance: 88981
  Within cohort/Between state variance: 10952
  Within state variance: 139120026
```

The function returns a fitted model object of class "cm" containing the estimators of the structure parameters. To compute the credibility premiums, one calls a method of `predict` for this class:

```
> predict(fit)

$cohort
[1] 1949 1543

$state
[1] 2048 1524 1875 1497 1585
```

One can also obtain a nicely formatted view of the most important results with a call to `summary`:

```
> summary(fit)

Call:
cm(formula = ~cohort + cohort:state, data = X, ratios = ratio.1:ratio.12,
    weights = weight.1:weight.12, method = "iterative")

Structure Parameters Estimators

  Collective premium: 1746

  Between cohort variance: 88981
  Within cohort/Between state variance: 10952
  Within state variance: 139120026

Detailed premiums

  Level: cohort
    cohort Indiv. mean Weight Cred. factor Cred. premium
    1      1967        1.407  0.9196       1949
    2      1528        1.596  0.9284       1543

  Level: state
    cohort state Indiv. mean Weight Cred. factor
    1      1     2061        100155 0.8874
    2      2     1511         19895 0.6103
    1      3     1806         13735 0.5195
    2      4     1353          4152 0.2463
    2      5     1600         36110 0.7398
```

```
    Cred. premium
    2048
    1524
    1875
    1497
    1585
```

The methods of `predict` and `summary` can both report for a subset of the levels by means of an argument `levels`. For example:

```
> summary(fit, levels = "cohort")

Call:
cm(formula = ~cohort + cohort:state, data = X, ratios = ratio.1:ratio.12,
    weights = weight.1:weight.12, method = "iterative")

Structure Parameters Estimators

  Collective premium: 1746

  Between cohort variance: 88981
  Within cohort variance: 10952

Detailed premiums

  Level: cohort
    cohort Indiv. mean Weight Cred. factor Cred. premium
    1      1967        1.407  0.9196       1949
    2      1528        1.596  0.9284       1543

> predict(fit, levels = "cohort")

$cohort
[1] 1949 1543
```

The results above differ from those of Goovaerts and Hoogstad (1987) for the same example because the formulas for the credibility premiums are different.

5 Bühlmann and Bühlmann–Straub models

As mentioned above, the Bühlmann and Bühlmann–Straub models are simply one-level hierarchical models. In this case, the Bühlmann–Gisler and Ohlsson estimators of the between variance parameters are both identical to the usual Bühlmann and Straub (1970) estimator

$$\hat{a} = \frac{w_{\Sigma\Sigma}}{w_{\Sigma\Sigma}^2 - \sum_{i=1}^{I} w_{i\Sigma}^2} \left(\sum_{i=1}^{I} w_{i\Sigma}(X_{iw} - X_{ww})^2 - (I-1)\hat{s}^2 \right), \qquad (14)$$

and the iterative estimator

$$\tilde{a} = \frac{1}{I-1} \sum_{i=1}^{I} z_i (X_{iw} - X_{zw})^2 \tag{15}$$

is better known as the Bichsel–Straub estimator.

To fit the Bühlmann model using `cm`, one simply does not specify any weights:

```
> cm(~state, hachemeister, ratios = ratio.1:ratio.12)

Call:
cm(formula = ~state, data = hachemeister, ratios = ratio.1:ratio.12)

Structure Parameters Estimators

  Collective premium: 1671

  Between state variance: 72310
  Within state variance: 46040
```

In comparison, the results for the Bühlmann–Straub model using the Bichsel–Straub estimator are:

```
> cm(~state, hachemeister, ratios = ratio.1:ratio.12,
+     weights = weight.1:weight.12)

Call:
cm(formula = ~state, data = hachemeister, ratios = ratio.1:ratio.12,
    weights = weight.1:weight.12)

Structure Parameters Estimators

  Collective premium: 1684

  Between state variance: 89639
  Within state variance: 139120026
```

6 Regression model of Hachemeister

The regression model of Hachemeister (1975) is a generalization of the Bühlmann–Straub model. If data shows a systematic trend, the latter model will typically under- or over-estimate the true premium of an entity. The idea of Hachemeister was to fit to the data a regression model where the parameters are a credibility weighted average of an entity's regression parameters and the group's parameters.

In order to use cm to fit a credibility regression model to a data set, one has to specify a vector or matrix of regressors by means of argument xreg. For example, fitting the model

$$X_{it} = \beta_0 + \beta_1(12 - t) + \varepsilon_t, \quad t = 1, \dots, 12$$

to the original data set of Hachemeister is done with

```
> fit <- cm(~state, hachemeister, xreg = 12:1, ratios = ratio.1:ratio.12,
+     weights = weight.1:weight.12)
> fit

Call:
cm(formula = ~state, data = hachemeister, ratios = ratio.1:ratio.12,
    weights = weight.1:weight.12, xreg = 12:1)

Structure Parameters Estimators

  Collective premium: 1885 -32.05

  Between state variance:  145359 -6623.4
                            -6623   301.8
  Within state variance: 49870187
```

Computing the credibility premiums requires to give the "future" values of the regressors as in predict.lm, although with a simplified syntax for the one regressor case:

```
> predict(fit, newdata = 0)

[1] 2437 1651 2073 1507 1759
```

References

H. Bühlmann. Experience rating and credibility. *ASTIN Bulletin*, 5:157–165, 1969.

H. Bühlmann and A. Gisler. *A course in credibility theory and its applications*. Springer, 2005. ISBN 3-5402575-3-5.

H. Bühlmann and W. S. Jewell. Hierarchical credibility revisited. *Bulletin of the Swiss Association of Actuaries*, 87:35–54, 1987.

H. Bühlmann and E. Straub. Glaubgwürdigkeit für Schadensätze. *Bulletin of the Swiss Association of Actuaries*, 70:111–133, 1970.

A. Forgues, V. Goulet, and J. Lu. Credibility for severity revisited. *North American Actuarial Journal*, 10(1):49–62, 2006.

M. J. Goovaerts and W. J. Hoogstad. *Credibility theory*. Number 4 in Surveys of actuarial studies. Nationale-Nederlanden N.V., Netherlands, 1987.

V. Goulet. Principles and application of credibility theory. *Journal of Actuarial Practice*, 6:5–62, 1998.

V. Goulet and T. Ouellet. On estimation of variance components in hierarchical credibility. 2007. In preparation.

V. Goulet and L.-P. Pouliot. Simulation of hierarchical models in R. *Journal of Statistical Software*, 2007. Submitted for publication.

C. A. Hachemeister. Credibility for regression models with application to trend. In *Credibility, theory and applications*, Proceedings of the Berkeley actuarial research conference on credibility, pages 129–163, New York, 1975. Academic Press.

W. S. Jewell. The use of collateral data in credibility theory: a hierarchical model. *Giornale dell'Istituto Italiano degli Attuari*, 38:1–16, 1975.

E. Ohlsson. Simplified estimation of structure parameters in hierarchical credibility. URL http://www.actuaries.org/ASTIN/Colloquia/Zurich/Ohlsson.pdf. Presented at the Zurich ASTIN Colloquium, 2005.