Bootstrapping the Ultimate Claims of Cumulative Payments and Claims-incurred Data using Kalman-filter Theory

Jochen Heberle

University of Hamburg, Germany

October 2, 2012
Overview

1. Introduction
   - Basic idea
   - Problem
   - Model Assumptions
   - Observed data

2. Kalman-filter
   - Estimation formulas
   - u-step predictions

3. Example
   - Given data
   - Parameter initialisation
   - Kalman-filtered triangle
   - Bootstrapping
   - Kernel-density
   - Summary and risk-measures
Kalman-filter Example


dev. year

0

i

J

accident year

0

i

I

C_{i,j}

Kalman-filter

dev. year

0

i

J

estimated

dev. year

0

i

J

C_{i,j}^{Pa}

accident year

0

i

I

noise term

accident year

0

i

I

C_{i,j}^{ln}

unobservable

observable

estimated

\hat{C}_{i,j}^{Kal,S}

\hat{C}_{i,j}^{Kal,F}

\hat{C}_{i,j}^{Kal,P}
“Classical” claims reserving is often only able to predict first and second moments. For example:

- First moment: Predicted ultimate claims
  \[ \Rightarrow E(C_{i,J} \mid \text{“observed data”}) \]
- Second moment: Mean square error of prediction (MSEP)
  \[ \Rightarrow E((\hat{C}_{i,J} - C_{i,J})^2 \mid \text{“observed data”}) \]

We would like to have the whole distribution
(for example of the cumulative claims reserve)
\[ \Rightarrow \text{Bootstrapping!} \]

Note: When bootstrapping this problem we only get a “bootstrap-distribution” and not a theoretical distribution.
The observed paid-/incurred data can be displayed as follows:

\[ C_{i,j}^{\text{Pa,In}} = \begin{pmatrix} C_{i,j}^{\text{Pa}} \\ C_{i,j}^{\text{In}} \end{pmatrix} = \begin{pmatrix} h_j^{\text{Pa}} \\ h_j^{\text{In}} \end{pmatrix} C_{i,j} + \begin{pmatrix} w_{i,j}^{\text{Pa}} \\ w_{i,j}^{\text{In}} \end{pmatrix} = h_j C_{i,j} + w_{i,j}, \]

The unobservable claims payments \( C_{i,j} \) are given through:

\[ C_{i,j+1} = g_j C_{i,j} + v_{i,j} \]

\( (w_{i,j}) \) and \( (v_{i,j}) \) are uncorrelated white-noise-processes with:

\[
E [w_{i,j} w_{k,l}'] = \begin{cases} T_{i,j} & i = k, j = l \\ 0 & \text{otherwise} \end{cases}
\text{ and } E [v_{i,j} v_{k,l}'] = \begin{cases} q_{i,j} & i = k, j = l \\ 0 & \text{otherwise} \end{cases}
\]

Cumulative claims \( C_{i,j} \) of different accident years \( i = 0, \ldots, l \) are independent.
The observed data given in the paid- and incurred-triangle are given through:

\[
\mathcal{D}_{K}^{Pa} = \{ C_{i,j}^{Pa} \mid i + j \leq K \}
\]

\[
\mathcal{D}_{K}^{In} = \{ C_{i,j}^{In} \mid i + j \leq K \}
\]

Figure: Observed paid- and incurred-triangle.

Combining \(\mathcal{D}_{K}^{Pa}\) and \(\mathcal{D}_{K}^{In}\) leads to:

\[
\mathcal{D}_{K} = \mathcal{D}_{K}^{Pa} \cup \mathcal{D}_{K}^{In}
\]
Estimation formulas:

\[
\hat{C}_{i,j|B_{i,k}}^{\text{Kal,S}} = \hat{C}_{i,j|B_{i,k-1}}^{\text{Kal,S}} + \left(\hat{\sigma}_{i,j|B_{i,k}}^{\text{Kal,S}}\right)^2 h_k^{'} \Delta_{i,k}^{-1} \left(C_{i,k}^{\text{Pa,In}} - h_k^{'} \hat{C}_{i,k}^{\text{Kal,P}}\right)
\]

\[
\hat{C}_{i,l-i}^{\text{Kal,F}} = \hat{C}_{i,l-i}^{\text{Kal,P}} + \left(\hat{\sigma}_{i,l-i}^{\text{Kal,P}}\right)^2 h_{l-i}^{'} \Delta_{i,l-i}^{-1} \left(C_{i,l-i}^{\text{Pa,In}} - h_{l-i}^{'} \hat{C}_{i,l-i}^{\text{Kal,P}}\right)
\]

\[
\hat{C}_{i,j+1}^{\text{Kal,P}} = g_j^{'} \hat{C}_{i,j}^{\text{Kal,P}} + \gamma_{i,j}^{'} \Delta_{i,j}^{-1} \left(C_{i,j}^{\text{Pa,In}} - h_j^{'} \hat{C}_{i,j}^{\text{Kal,P}}\right)
\]

- The values \(\hat{\sigma}_{i,j|B_{i,k}}^{\text{Kal,S}}, \hat{\sigma}_{i,j}^{\text{Kal,P}}, \gamma_{i,j}, \text{ and } \Delta_{i,j}\) are also given in the “estimation formulas”.
- \(B_{i,j}\) is a set of observations, i.e.:

\[
B_{i,j} = \{C_{i,k}^{\text{Pa}} \mid k = 0, \ldots, j\} \cup \{C_{i,k}^{\text{In}} \mid k = 0, \ldots, j\}
\]

- The values \(h_j\) and \(g_j\) must be given in advance, i.e. estimated through the given data or determined through experts opinion.
With the above described estimators we get the following triangle-estimation:

<table>
<thead>
<tr>
<th>dev. year</th>
<th>0</th>
<th>j</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>accident year</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>i</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>l</td>
<td>B</td>
<td>C</td>
<td>(D)</td>
</tr>
</tbody>
</table>

The Kalman-prediction (see above) is only a 1-step prediction. Since we need more than 1-step predictions (C), there also exists a formula for $u$-step Kalman-predictions (D):

\[
\hat{C}_{i,l-i+u+1}^{\text{Kal,P}} = gl_{i-u} \hat{C}_{i,l-i+1}^{\text{Kal,P}} = \cdots = \hat{C}_{i,l-i+1}^{\text{Kal,P}} \prod_{m=1}^{u} g_{l-i+m}
\]
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1,216,632</td>
<td>1,347,072</td>
<td>1,786,877</td>
<td>2,281,606</td>
<td>2,656,224</td>
<td>2,909,307</td>
<td>3,283,388</td>
<td>3,587,549</td>
<td>3,754,403</td>
<td>3,921,258</td>
</tr>
<tr>
<td>1</td>
<td>798,924</td>
<td>1,051,912</td>
<td>1,215,785</td>
<td>1,349,939</td>
<td>1,655,312</td>
<td>1,926,210</td>
<td>2,132,833</td>
<td>2,287,311</td>
<td>2,567,056</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1,115,636</td>
<td>1,387,387</td>
<td>1,930,867</td>
<td>2,177,002</td>
<td>2,513,171</td>
<td>2,931,930</td>
<td>3,047,368</td>
<td>3,182,511</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1,052,161</td>
<td>1,321,206</td>
<td>1,700,132</td>
<td>1,971,303</td>
<td>2,298,349</td>
<td>2,645,113</td>
<td>3,003,425</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>808,864</td>
<td>1,029,523</td>
<td>1,229,626</td>
<td>1,590,338</td>
<td>1,842,662</td>
<td>2,150,351</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1,016,862</td>
<td>1,251,420</td>
<td>1,698,052</td>
<td>2,105,143</td>
<td>2,385,339</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>948,312</td>
<td>1,108,791</td>
<td>1,315,524</td>
<td>1,487,577</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>917,530</td>
<td>1,082,426</td>
<td>1,484,405</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1,001,238</td>
<td>1,376,124</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>841,930</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: Paid- and incurred-triangle.
Parameter initialisation:

\[ h = (1, 1)' \]
\[ g = 1.08 \]
\[ \hat{C}_{i,0}^{Kal} = (0.5, 0.5) \left( C_{i,0}^{Pa}, C_{i,0}^{ln} \right)' \]

- The “meaning” of \( h = (1, 1)' \) is that the weighting between paid- and incurred-data is equal.

- The parameter \( g = 1.08 \) is estimated through the average of chain-ladder-factors computed on each development triangle.
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2,289,374</td>
<td>2,474,813</td>
<td>3,548,014</td>
<td>3,535,831</td>
<td>3,601,504</td>
<td>3,671,674</td>
<td>3,785,363</td>
<td>3,946,798</td>
<td>4,085,715</td>
<td>3,921,258</td>
</tr>
<tr>
<td>1</td>
<td>1,719,684</td>
<td>1,858,978</td>
<td>3,078,566</td>
<td>2,748,842</td>
<td>2,485,488</td>
<td>2,571,328</td>
<td>2,673,416</td>
<td>2,761,359</td>
<td>2,743,505</td>
<td>2,965,729</td>
</tr>
<tr>
<td>3</td>
<td>1,992,753</td>
<td>2,154,166</td>
<td>3,578,302</td>
<td>3,325,208</td>
<td>3,065,777</td>
<td>3,162,866</td>
<td>3,208,673</td>
<td>3,468,576</td>
<td>3,749,530</td>
<td>4,053,242</td>
</tr>
<tr>
<td>4</td>
<td>1,788,523</td>
<td>1,933,393</td>
<td>3,074,612</td>
<td>2,792,267</td>
<td>2,758,211</td>
<td>2,724,675</td>
<td>2,945,373</td>
<td>3,183,948</td>
<td>3,441,848</td>
<td>3,720,638</td>
</tr>
<tr>
<td>7</td>
<td>2,000,480</td>
<td>2,162,518</td>
<td>2,946,251</td>
<td>3,184,898</td>
<td>3,442,874</td>
<td>3,721,747</td>
<td>4,023,209</td>
<td>4,349,088</td>
<td>4,701,365</td>
<td>5,082,175</td>
</tr>
<tr>
<td>8</td>
<td>1,881,201</td>
<td>2,754,442</td>
<td>2,977,551</td>
<td>3,218,733</td>
<td>3,479,450</td>
<td>3,761,286</td>
<td>4,065,950</td>
<td>4,395,292</td>
<td>4,751,310</td>
<td>5,136,166</td>
</tr>
<tr>
<td>9</td>
<td>1,943,653</td>
<td>2,101,089</td>
<td>2,271,277</td>
<td>2,455,251</td>
<td>2,654,126</td>
<td>2,869,110</td>
<td>3,101,508</td>
<td>3,352,730</td>
<td>3,624,301</td>
<td>3,917,870</td>
</tr>
</tbody>
</table>

**Table:** Estimated cumulative claims $\hat{C}_{i,j}$. 

---

11 / 15
Bootstrapping-approach:

1. Calculate paid-/incurred-residuals:
   \[ r_{i,j}^{\text{Pa}} = C_{i,j}^{\text{Pa}} - h_j^{\text{Pa}} \hat{C}_{i,j}^{\text{Kal}} \]
   \[ r_{i,j}^{\text{In}} = C_{i,j}^{\text{In}} - h_j^{\text{In}} \hat{C}_{i,j}^{\text{Kal}} \]

2. The residuals are transformed to make them iid.

3. Sample from the two sets of residuals and re-build paid-/incurred-triangles (the sampling is done without replacement and synchronous to preserve a given correlation structure).

4. For each “new” paid-/incurred-triangle pair the unobservable “Kalman”-triangle is estimated.

5. Calculate for each bootstrapped Kalman-triangle the cumulative claims reserve, i.e.:
   \[ \tilde{R} = \sum_{i=1}^{l} \left( \tilde{C}_{i,l}^{\text{Kal}} - \hat{C}_{i,l-i}^{\text{Kal}} \right) \]
Figure: Density-plot of the bootstrapped aggregated claims reserves.
<table>
<thead>
<tr>
<th>Min.</th>
<th>1st Qu.</th>
<th>Median</th>
<th>Mean</th>
<th>3rd Qu.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.86</td>
<td>11.97</td>
<td>12.11</td>
<td>12.09</td>
<td>12.26</td>
<td>12.93</td>
</tr>
</tbody>
</table>

**Table:** Summary of 10,000 bootstrapped aggregated claims reserves ($\times 10^6$).

<table>
<thead>
<tr>
<th>$p$</th>
<th>0.90</th>
<th>0.95</th>
<th>0.99</th>
</tr>
</thead>
<tbody>
<tr>
<td>$VaR_p$</td>
<td>12,397,041</td>
<td>12,484,181</td>
<td>12,650,395</td>
</tr>
<tr>
<td>$ES_p$</td>
<td>12,408,385</td>
<td>12,489,257</td>
<td>12,651,168</td>
</tr>
<tr>
<td>$CTE_p$</td>
<td>12,510,478</td>
<td>12,585,705</td>
<td>12,727,641</td>
</tr>
</tbody>
</table>

**Table:** Value-at-Risk, Expected-Shortfall and Conditional-Tail-Expectation.
Thank you