Stochastic Ensemble Loss Reserving

Benjamin Avanzi\textsuperscript{a} Yanfeng Li\textsuperscript{b} Bernard Wong\textsuperscript{b} Alan Xian\textsuperscript{c}

\textsuperscript{a}The University of Melbourne, \textsuperscript{b}The University of New South Wales, \textsuperscript{c}Taylor Fry Actuarial and Analytics Consultancy

21 June, 2022
Background

Motivation

Ensemble Modelling framework
   Choice of component models
   Combination of models into final predictions

Illustration
   Example data
   Comparison of the predictive performance of the ensembles
   Predictive performance on incremental payments level
   Predictive performance on aggregate reserve level

Summary
Background

Motivation

Ensemble Modelling framework
  Choice of component models
  Combination of models into final predictions

Illustration
  Example data
  Comparison of the predictive performance of the ensembles
  Predictive performance on incremental payments level
  Predictive performance on aggregate reserve level

Summary
Background on the development of stochastic loss reserving models

Traditional loss reserving models:

- Mack’s Chain-Ladder: Mack (1993)
- Generalized Linear Models:
  - Over-Dispersed Poisson: Renshaw and Verrall (1998)
- Parametric Curves:
  - Hoerl Curve and Wright Curve: Wright (1990)

Machine learning loss reserving models:

- Smoothing Splines: England and Verrall (2001)
- Generalized Additive Models for Location Scale and Shape (GAMLSS):
  Spedicato, Clemente and Schewe (2014)
Stochastic Ensemble Loss Reserving

Motivation

Background

Motivation

Ensemble Modelling framework
- Choice of component models
- Combination of models into final predictions

Illustration
- Example data
- Comparison of the predictive performance of the ensembles
- Predictive performance on incremental payments level
- Predictive performance on aggregate reserve level

Summary
Motivation for model combination

- Loss reserving generally focuses on identifying a single model that can generate superior predictive performance.
- However, different loss reserving models specialise in capturing different aspects of loss data.
- Existed model combination strategies in loss reserving:
  - Are usually based on ad-hoc, manual rules (Friedland, 2010): could be subjective and hard to take account into the full distribution of claims.
  - Can cause complications in cases where one wants information on the random distribution of claims: as Shapland (2022) suggests, future development of model combination strategies should “reflect actuaries’ judgements about the entire distribution, not just a central estimate.”
We aim to propose an objective model combination strategy tailored to general insurance loss reserving data, with the following attributes:

- The combination weights are driven by the data and minimum manual adjustment is required
- The triangular shape of aggregate loss reserving data is taken into account
- The unique characteristics of loss reserving data (e.g. impacts and features related to different Accident and Development period combinations) are considered when building the ensemble
Background

Motivation

**Ensemble Modelling framework**
- Choice of component models
- Combination of models into final predictions

Illustration
- Example data
- Comparison of the predictive performance of the ensembles
- Predictive performance on incremental payments level
- Predictive performance on aggregate reserve level

Summary
Background

Motivation

**Ensemble Modelling framework**

Choice of component models

Combination of models into final predictions

Illustration

Example data

Comparison of the predictive performance of the ensembles

Predictive performance on incremental payments level

Predictive performance on aggregate reserve level

Summary
How to select models?

The following objectives are considered to select component models being included in the ensemble:

1. The component models should be able to fit mechanically without requiring much user adjustment, so as to avoid the potential subjective bias in model fitting, and to save time spent on manually adjusting each component model.

2. The component models should have different strengths and limitations so as to complement each other in the ensemble.

3. The component models should be easily identifiable and interpretable, and hence are restricted to traditional stochastic loss reserving models and statistical learning models with relatively simple structures.

In short, we want the ensemble to be automatically generated (1), rich (2) and constituted of known and identifiable components (3).
Our selection of component models in the ensemble

<table>
<thead>
<tr>
<th>Model Class</th>
<th>Model Structures</th>
<th>Fitted Distributions</th>
<th>Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLM Based Model</td>
<td>cross-classified</td>
<td>ODP, Log-Normal, Gamma</td>
<td>accident and development effects</td>
</tr>
<tr>
<td></td>
<td>cross-classified+Calendar term</td>
<td>ODP, Log-Normal, Gamma</td>
<td>calendar period effects</td>
</tr>
<tr>
<td></td>
<td>Hoerl Curve</td>
<td>ODP, Log-Normal, Gamma</td>
<td>shape of development patterns</td>
</tr>
<tr>
<td></td>
<td>PPCI</td>
<td>ODP</td>
<td>claims notification effects</td>
</tr>
<tr>
<td></td>
<td>PPCF</td>
<td>ODP</td>
<td>claims finalisation effects</td>
</tr>
<tr>
<td></td>
<td>ZALN GLM and ZAGA GLM</td>
<td>ZALN, ZAGA</td>
<td>Occurrence of zero incremental claims</td>
</tr>
<tr>
<td>Smoothing Splines</td>
<td>Smoothing Splines</td>
<td>Normal, Log-Normal, Gamma</td>
<td>shape of development patterns</td>
</tr>
<tr>
<td>GAMLSS</td>
<td>GAMLSS</td>
<td>Log-Normal, Gamma</td>
<td>varying dispersion</td>
</tr>
</tbody>
</table>

Table 1: Summary of component models
Background

Motivation

**Ensemble Modelling framework**
- Choice of component models
- Combination of models into final predictions

Illustration
- Example data
- Comparison of the predictive performance of the ensembles
- Predictive performance on incremental payments level
- Predictive performance on aggregate reserve level

Summary
Ensemble construction framework

Aggregate Loss Data

Development Periods

Accident Periods

Data Partition

Training component models on training set ($D_{Train}$)

Level 1 Predictions

Training component models using the entire upper triangle ($D_{In}$)

Observations

Predicted Density

$\{y_{ij}, i, j \in D_{val}\}$

$\{\hat{f}_1(y_{ij}),\cdots, \hat{f}_M(y_{ij}), i, j \in D_{val}\}$

Linear Pool Ensemble

Estimating Combination Weights

Final Predictions
Defining model combination criterion

A *model combination criterion* decides how much weight is allocated to each component model. We consider *strictly proper scoring rules* as combination criteria:

- A *scoring rule* assigns a score to a forecasting model that reflects how close it is to the data-generating model (Gneiting and Raftery, 2007)

- Define $S(P, x)$ as the score received by the forecasting model $P$, and $S(Q, x)$ as the score assigned to the true model $Q$: a scoring rule is defined to be strictly proper if $S(Q, x) \geq S(P, x)$ always holds, with the equality holds if and only if $P = Q$

- A strictly proper scoring rule ensures the true distribution always receive the best score (Gneiting and Raftery, 2007; Gneiting and Katzfuss, 2014): encourages the forecasters to quote the true distribution
We focus on the Log Score, a widely applied strictly proper scoring rule in probabilistic forecasting literature (Gneiting and Raftery, 2007):

- Has solid theoretical foundation: Log-Likelihood, information theory (Gneiting and Raftery, 2007), Kullback-Leibler divergence (KLIC) distance (Hall and Mitchell, 2007)
- Has a closed-form expression for all distribution: computationally inexpensive
- The average Log Score attained by a model over a sample can be specified as:

\[
\text{LogS} = \frac{1}{|D|} \sum_{i,j \in D} \ln(\hat{f}(y_{i,j})) \tag{1}
\]

Here $\hat{f}(y_{i,j})$ is the predicted density at the observation $y_{ij}$, and $D$ is the set of observations that are of interest to modellers.
Basic ensembling choices

Best Model in the Validation set (BMV):

- Select the model with the highest Log Score averaged across all the cells in the validation set;
- Note that this corresponds to the traditional “model selection” strategy.

Equally weighted ensemble (EW):

- A common benchmark ensemble strategy in forecasting literature (Gneiting and Katzfuss, 2014) that assigns equal weights to all component models.
Standard ML ensembling approach: The Standard Linear Pool (SLP)

- This is the *standard machine learning ensembling approach* for combining distributional forecasts.
- The final prediction is a linear combination of predictions generated by component models (Gneiting, Ranjan et al., 2013).
- For a Standard Linear Pool, the optimal combination weights are estimated from the validation data by maximising the mean Log-Score received by the ensemble:

\[
\max_w \frac{1}{|D_{val}|} \sum_{y_{ij} \in D_{val}} \ln \left( \sum_{m=1}^{M} w_m \hat{f}_m(y_{i,j}) \right) \\
\text{s.t. } \sum_{m=1}^{M} w_m = 1, w_m \geq 0
\]
Accident periods Dependent Linear Pools (ADLP)

- The combination scheme under SLP implies each model receives the same weight in all accident periods.
- But Loss reserving models tend to have different predictive performances in different accident periods (Taylor, 2000):
  - e.g. PPCF tends to perform better in mature accident periods by considering the small number of outstanding claims in those periods, whereas PPCI and Chain Ladder usually have better performance in immature periods.
- To let the weights vary by accident periods, a possible solution is to partition the validation data by accident periods to train the combination weights:
  - e.g. partition data from accident period 2 to accident period 28 to the first subset and the data from later accident periods to the second subset.
However, if only accident periods are considered, the weights in the second subset cannot capture the full development period effects:

| Development Periods | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
|---------------------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|                     | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| Training Set        |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| Validation Set 1    |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| Test Set 1          |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| Validation Set 2    |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| Test Set 2          |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
Solution: to fully capture the accident period and development period effects, we propose an Accident period Dependent Linear Pools (ADLP) framework by using overlapping partitions:

![Figure 1: ADLP Validation Subset 1](image1)

![Figure 2: ADLP Validation Subset 2](image2)

Note: The bottom set results will hence correspond to the SLP results.
The above ADLP scheme can be extended to any number of subsets.

More formally, the validation set is firstly divided into $K$ region, denoted as $D_{val}^1,...,D_{val}^K$, sorted by accident periods.

The weights allocated to the $k^{th}$ subset are estimated on $D_{val}^{k^*}$, where

$$D_{val}^{k^*} = D_{val}^1 \cup ... \cup D_{val}^k$$

$$\max_{w^k} \frac{1}{|D_{val}^{k^*}|} \sum_{y_{ij} \in D_{val}^{k^*}} \ln(\sum_{m=1}^{M} w_m^k \hat{f}_m(y_{i,j}))$$

s.t. $\sum_{m=1}^{M} w_m^k = 1, w_m^k \geq 0$
Optimisation Algorithm

- We implement the Minorization-Maximization strategy, being a popular strategy for Log Score optimisation, to derive the combination weights.
- Instead of directly maximizing the Log Score, a surrogate objective function is maximised (Conflitti et al., 2015).
- We initialise the weights as $\frac{1}{M}$, which corresponds to an equally weighted ensemble.
- The weights are then updated iteratively until convergence by maximising the surrogate function:

$$
(w_m^k)_{i+1} = (w_m^k)_i \sum_{y_{ij} \in D_{val}^{k*}} \frac{\hat{f}_m(y_{i,j})}{\sum_{l=1}^{M}{\hat{f}_l(y_{i,j})(w_l^k)_i}}
$$

(4)

- The non-negativity constraint and sum-to-unity constraint on weights are automatically satisfied in each iteration.
Background

Motivation

Ensemble Modelling framework
  Choice of component models
  Combination of models into final predictions

Illustration
  Example data
  Comparison of the predictive performance of the ensembles
  Predictive performance on incremental payments level
  Predictive performance on aggregate reserve level

Summary
Background

Motivation

Ensemble Modelling framework
- Choice of component models
- Combination of models into final predictions

Illustration
- Example data
  - Comparison of the predictive performance of the ensembles
  - Predictive performance on incremental payments level
  - Predictive performance on aggregate reserve level

Summary
Data for the illustration

Our models are illustrated using the SynthETIC simulator (Avanzi, Taylor, Wang and Wong, 2021), with the following parameters:

- Loss triangle size: 40 x 40
- 100 data sets are simulated to test the average performance of each model
- Key characteristics of simulated data:
  - Claim payments are long-tailed
  - High volatility of incremental claims
  - Payment size depends on claim closure

These key assumptions aim to simulate the complicated issues that actuaries can encounter in practice when modelling outstanding claims.
Background

Motivation

Ensemble Modelling framework
  Choice of component models
  Combination of models into final predictions

Illustration
  Example data
  Comparison of the predictive performance of the ensembles
  Predictive performance on incremental payments level
  Predictive performance on aggregate reserve level

Summary
How to compare ensembling strategies?

- We will want to compare the performance of all four possible weighting schemes presented above.
- To compare the performance of our proposed ensembles with the benchmark strategies, the following evaluation metrics are used:
  - Incremental claims level: out-of-sample Log Score, with the statistical significance of score differential assessed by the Diebold-Mariano test (Diebold and Mariano, 1995).
  - Aggregate reserve level: central reserve bias, reserve bias at the 75th quantile
Background

Motivation

Ensemble Modelling framework
   Choice of component models
   Combination of models into final predictions

Illustration
   Example data
   Comparison of the predictive performance of the ensembles
   Predictive performance on incremental payments level
   Predictive performance on aggregate reserve level

Summary
There is no substantial difference between the equally weighted (EW) ensemble and SLP:

- SLP does not consider the difference in models’ performance in different accident periods (especially in earlier accident periods)
The optimal ADLP has the best overall performance:

- It effectively captures the outstanding performance of several models in earlier accident periods.
- It outperforms BMV, Equally Weighted Ensemble and SLP in 100 simulated data sets, and the difference is statistically significant in most data sets by using the Diebold-Mariano test.
ADLP with various split points

We also investigate the impact of split points on the ensemble’s performance:

- All ADLP ensembles outperform SLP
- The optimal split point for ADLP ensembles is at accident period 15
Assessing the difference

The Log Score differential between the optimal ADLP and its competing strategies is assessed by the Diebold-Mariano test:

- The null hypothesis is rejected in the majority of simulated datasets
- The test results imply a decision in favour of the optimal ADLP
Background

Motivation

Ensemble Modelling framework
  Choice of component models
  Combination of models into final predictions

Illustration
  Example data
  Comparison of the predictive performance of the ensembles
  Predictive performance on incremental payments level

  Predictive performance on aggregate reserve level

Summary
Predictive performance on aggregate reserve level

All linear pools ensembles outperform BMV and EW.
Background

Motivation

Ensemble Modelling framework
- Choice of component models
- Combination of models into final predictions

Illustration
- Example data
- Comparison of the predictive performance of the ensembles
- Predictive performance on incremental payments level
- Predictive performance on aggregate reserve level

Summary
Summary

- Introduced an **objective, data-driven** method to optimally combine different models for reserving.

- Proposed a novel validation strategy that addresses the characteristics of general insurance loss reserving data:
  - the triangular shape of aggregate loss data;
  - impacts and features related to different Accident and Development period combinations.

- Illustrated the proposed ensemble framework using simulated data:
  - The proposed ensembles outperform both the model selection strategy and the equally weighted ensemble based on various evaluation metrics.
  - The investigation of split points can provide general guidance for practitioners to partition the loss data.
Appendix A: Details of Minorization-Maximization strategy

The surrogate objective function to be maximised is specified as (Conflitti et al., 2015):

\[ \phi_\lambda(w^k_m, a) = \frac{1}{|D^k_{val}|} \sum_{y_{ij} \in D^k_{val}} \sum_{m=1}^{M} \frac{\hat{f}_m(y_{i,j})a_m}{\sum_{l=1}^{M} \hat{f}_l(y_{i,j})a_l} \ln\left(\frac{w^k_m}{a_m} \sum_{l=1}^{M} \hat{f}_l(y_{i,j})a_l\right) - \lambda \left(\sum_{m=1}^{M} w^k_m\right) \]

(5)

The weights are then updated iteratively by maximizing the surrogate function defined in (5):

\[ (w^k_m)_{i+1} = \arg\max_{w} \phi_\lambda(w^k_m, (w^k_m)_i) \]

(6)

\[ a_m \] is an arbitrary weight, \( \lambda \) is a Lagrange Multiplier, and \( D^k_{val} = D^1_{val} \cup \ldots \cup D^k_{val} \). For the standard linear pool ensemble, we have \( w^1_m = \ldots = w^k_m = w_m \).
By setting $\frac{\partial \phi_\lambda(w^k_m, (w^k_m)_i)}{\partial w^k_m} = 0$, we have:

$$w^k_m = \frac{1}{\lambda |D^\ast_{val}|} \sum_{y_{ij} \in D^\ast_{val}} \frac{\hat{f}_m(y_{i,j})(w^k_m)_i}{\sum_{l=1}^M \hat{f}_l(y_{i,j})(w^k_l)_i}$$

Based on the constraint $\sum_{m=1}^M w^k_m = 1$, we have:

$$\sum_{m=1}^M w^k_m = \frac{1}{\lambda |D^\ast_{val}|} \sum_{y_{ij} \in D^\ast_{val}} \frac{\sum_{m=1}^M \hat{f}_m(y_{i,j})(w^k_m)_i}{\sum_{l=1}^M \hat{f}_l(y_{i,j})(w^k_l)_i} = \frac{1}{\lambda |D^\ast_{val}|} \sum_{y_{ij} \in D^\ast_{val}} 1 = \frac{1}{\lambda} = 1.$$

Therefore, $\lambda = 1$, and the updated weights in each iteration become:

$$(w^k_m)_{i+1} = (w^k_m)_i \sum_{y_{ij} \in D^\ast_{val}} \frac{\hat{f}_m(y_{i,j})}{\sum_{l=1}^M \hat{f}_l(y_{i,j})(w^k_l)_i}.$$  \hfill (7)
Rationale behind the surrogate cost function

▶ Define our objective function (i.e. the Log Score) as:

$$\Phi_\lambda(w^k) = \frac{1}{|D_{val}^{k*}|} \sum_{y_{ij} \in D_{val}^{k*}} \ln(\sum_{m=1}^{M} w^k_m \hat{f}_m(y_{i,j})) - \lambda(\sum_{m=1}^{M} w^k_m)$$

▶ The surrogate cost function $\phi_\lambda$ defined in (5) has two desirable properties:

1. $\phi_\lambda(a, a) = \Phi_\lambda(a)$
2. $\phi_\lambda(w, a) \leq \Phi_\lambda(w)$ for any $w$ and $a$

▶ Based on the above properties, the Minorization-Maximization strategy yields a monotonic increase of the Log Score (i.e. $\Phi_\lambda$) in the iteration process:

$$\Phi_\lambda((w^k_m)_{i+1}) \geq \phi_\lambda((w^k_m)_{i+1}, (w^k_m)_i) \geq \phi_\lambda((w^k_m)_i, (w^k_m)_i) = \Phi_\lambda((w^k_m)_i)$$

▶ Due to the monotonic increase of the surrogate function, the weights are expected to converge to the maximizer of the Log Score (Conflitti et al., 2015), subject to the constraints $w^k_m \geq 0$, and $\sum_{m=1}^{M} w^k_m = 1$
Appendix B: ADLP with three subsets

- This paper focuses on ADLP with two subsets
- However, one can divide the data into more subsets to train the corresponding combination weights
- An example of three-subsets partition strategy would be:
  - Subset 1: Accident periods 2-15
  - Subset 2: Accident periods 2-29
  - Subset 3: Accident periods 2-40
There is no significant improvement in Log Score by introducing another subset:

- The Log Score differential in semi-mature accident periods is small.
- This phenomenon might be explained by the small difference in the performance of component models during semi-mature accident periods, limiting the diversification benefit.
Appendix C: Continuously Ranked Probability Score (CRPS)

Another popular proper score in evaluating distributional forecasting performance and model combination is the Continuously Ranked Probability Score (CRPS) (Gneiting and Ranjan, 2011):

\[
\text{CRPS}(F^*, y_k) = \int_{-\infty}^{\infty} \left( \sum_{m=1}^{M} w_m \cdot F_m(y_k) - I_{z \geq y_k} \right)^2 dz. \tag{8}
\]

A discretised version of (8) is often used for computational efficiency (Gneiting and Ranjan, 2011):

\[
\text{CRPS}(F^*, y_k) = \sum_{z=z_l}^{z_u} \left( \sum_{m=1}^{M} w_m \cdot F_m(y_k) - I_{z \geq y_k} \right)^2, \tag{9}
\]

where \(z_l\) and \(z_u\) are the lower and upper bounds for integration, respectively.

Note that CRPS is a negatively oriented score.
As per Hersbach (2000), the CRPS can be interpreted as a measurement of the difference between the predicted and observed cumulative distributions (i.e. the perfect forecaster).
Empirical results for CRPS

ADLP still outperforms SLP and BMV based on CRPS, though the optimal split point is slightly different from Log Score:

![Mean CRPS v.s. Different split points](chart.png)

![CRPS(under 100 simulations)](boxplot.png)
References


References II


References III


References IV
