Are the Upper Tails of Predictive Distributions of Outstanding Liabilities Underestimated when using Bootstrapping?

Peter England PhD
Martin Cairns FIA

GIRO Conference, 07 October 2009
Edinburgh

Background

- The ROC/GIRO Reserving Uncertainty Working Party reports 2007/2008 suggested that the upper tail of predictive distributions of outstanding liabilities may be underestimated.
  - This was considered in Section 9 and Appendix B of the 2007 report and a "key area" of the 2008 report.
- The reports have never been published in a peer-reviewed journal. Despite this, the results have generated some interest and are gaining credibility, for example:
  - *"As shown by the 'General Insurance Reserve Oversight Committee', commonly used existing methods are inadequate to cover the full range of reserving variability."* [1]
  - The FSA has also shown interest and is asking companies to comment on the potential underestimation where bootstrapping has been used[2].
- This presentation evaluates the evidence and sheds new light on the issues[3].

(1) Source: Brooks *et al* (2009). Actuarial aspects of internal models for Solvency II (Appendix A, 17.3.1). Presented to the Institute of Actuaries 23 Feb 2009.

(2) It is unclear whether the FSA actually *believes* the report, or is just using it as a way to ask insightful questions.

(3) We only consider Section 9 and Appendix B of the 2007 report and the "key areas" of the 2008 report.

Key ROC/GIRO Working Party Questions

Assessing bootstrapping

When assessing the performance of stochastic reserving methods, the working party considered the following questions:

- How well do stochastic reserving methods "predict" the upper tails when the claims generating process agrees with the underlying model?
- How robust is the underlying stochastic reserving method when the claims generating process is different?

Obviously both questions are interesting, although we are only interested in question 1 in this presentation applied to bootstrapping, since it would indicate a potential flaw in the methodology.

[Question 2 is also interesting in practice, since we will never know for sure that the model is appropriate.]

ROC/GIRO Working Party Method

1. Simulate 30,000 data sets where the future is known, consistent with the model assumptions
2. For each of the 30,000 data sets, consider only the “triangle” element (holding back the ‘true’ outstanding claims)
3. For each of the 30,000 triangles apply the stochastic method[4], and record the percentile at which the ‘true’ value lies (amongst other statistics)
4. If the test is appropriate, then 1% of ‘true’ values should exceed the 99th percentile

(4) Where bootstrapping is used to obtain a predictive distribution, obtaining that percentile from the simulated results is straightforward. The WP used 1000 bootstrap iterations for the 2007 report, and tested 2000 iterations in the 2008 report.

Where a predictive distribution is not available, additional assumptions have to be made, for example, assuming the reserves are lognormally distributed – in that case the results will be dependent on the suitability (or otherwise) of that additional assumption.

ROC/GIRO WP Findings
Over-dispersed Poisson model

- The GIRO WP considered a special case of the over-dispersed Poisson model (constant scale parameter, chain ladder structure)
- The 30,000 data sets were generated using "Algorithm B" (where Algorithm B supposedly satisfies the assumptions of the ODP model).
- Using bootstrapping to obtain a predictive distribution (using England, 2002), 2.6% of true values were found to lie above the 99th percentile[5], indicating potential under-estimation of the upper tail.

  Should we be surprised? Maybe. Maybe not. In the discussion of England & Verrall (2002), England says:

  "Mr Murphy, Mr Sharma and Ms Cresswell mentioned extremes and taking care over the extremes of the predictive distribution. I agree totally that the model assumptions should be scrutinised carefully if the main interest is in the extremes."

  But are the working party results correct?

(5) ROC/GIRO WP Report 2007/Section B2.4.3 and 2008/Table 2-3

ROC/GIRO WP Findings
Mack's model

- The GIRO WP did not bootstrap Mack's model (even though it is possible to bootstrap in a way that is consistent with Mack's model[6])
- 10,000 data sets were generated (repeatedly) using "Algorithm A", (where Algorithm A supposedly "perfectly" satisfies the assumptions of Mack's model).
- Applying Mack's method to obtain a mean and standard deviation of the forecast, then assuming the forecast is lognormally distributed, around 10% of true values were found to lie above the 99th percentile[7], indicating potential under-estimation of the upper tail.
- Obviously this is quite surprising. In fact, it is so surprising that the WP decided to try it again:

  "To provide a final comprehensive check, another member of the WP … carried out a completely independent simulation exercise… For the independent exercise, Mack's method and Algorithm A were implemented…"

  "… Having obtained essentially the same results in two quite separate and independent simulation exercises, we are confident that these results are genuine."

  "It is beyond the scope of this paper to definitively explain where Mack's method goes wrong…"

  But are the working party results correct?

(6) England & Verrall (2006)

(7) ROC/GIRO WP Report 2007 Table B-4

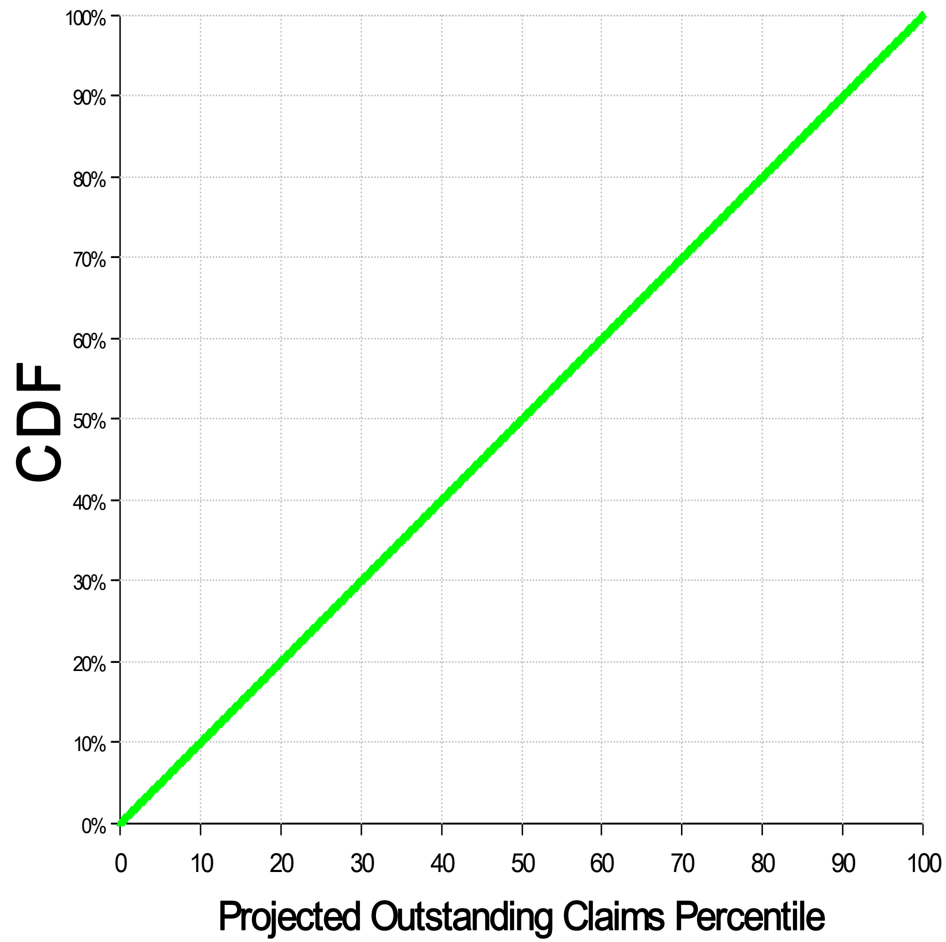
EMB Research

Given the working party findings, we had some key questions:

- Can we replicate the WP findings under their parameters?
- Can we suggest why the WP findings may have arisen?
- If so, can we suggest alternative methodologies that eliminate or ameliorate the issues?
- Is the test itself valid? That is, under the test procedure, do we expect, for example, 1% of ‘true’ values to lie above the 99th percentile of the forecast?

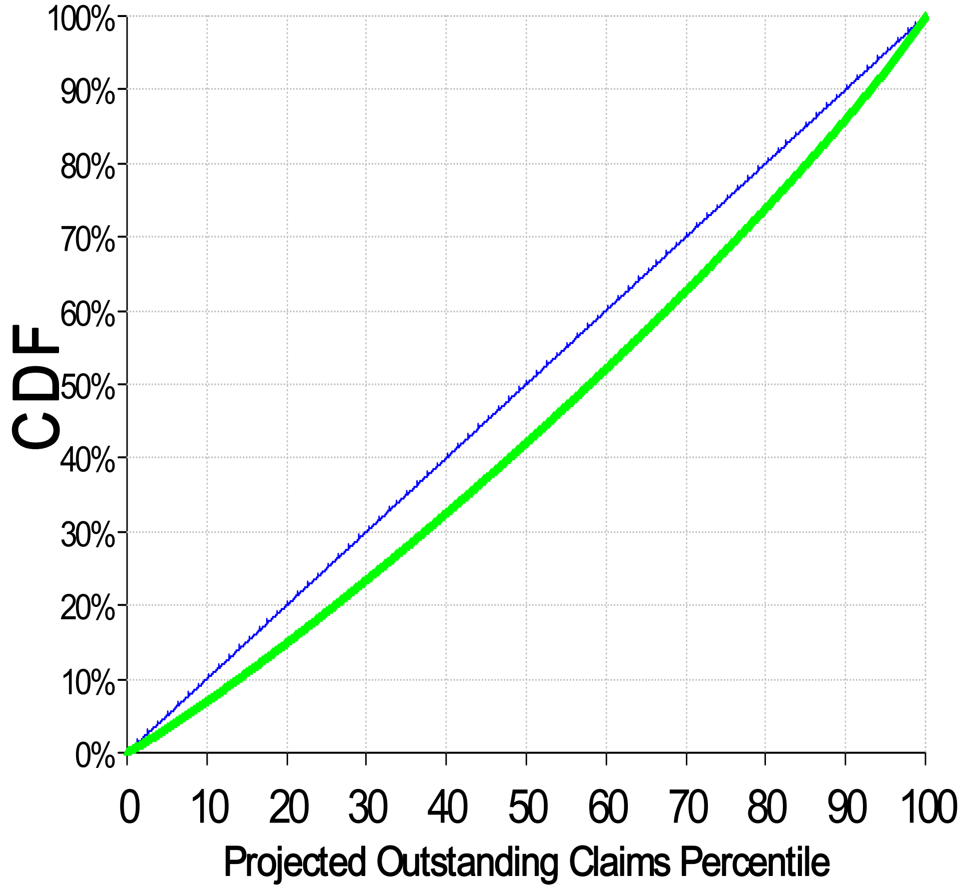
Some background on what we'd *expect* to see

- Let's assume the test is valid for the moment.
- The test looks at the proportion of simulated data sets for which the (bootstrapped) projected x^{th} percentile is exceeded by the (simulated) true future outstanding
- Plotting this for all percentiles, with the x^{th} percentile on the x-axis, and the (equivalent) proportion of simulations where the "true" outstanding is less than the projected x^{th} percentile should give a uniform distribution, $Y=X$
- Effectively this is plotting the CDF of the distribution function Pr("true outstanding"< x^{th} percentile) for $0<x<100\%$
- But how does the $Y=X$ line change as we violate the uniformity assumption?



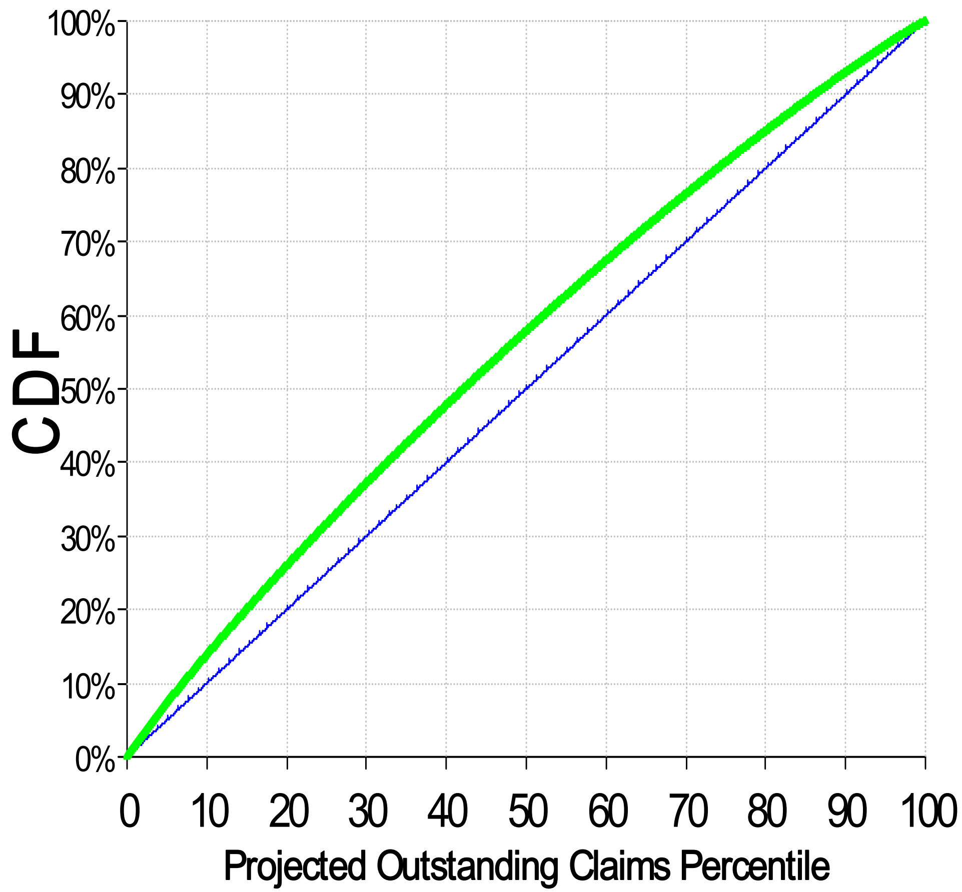
Some background on what we'd *expect* to see

- According to the test the CDF of the proportion of "true" outstandings below the projected percentile should be uniform, *Y=X* (shown in blue)
- The green line shows the impact of changing from the true distribution (assumed to be Normal(8,1) for this simple example, to one with a biased mean below the true mean, with the correct standard deviation, (assumed to be Normal(7.8,1) for this simple example)



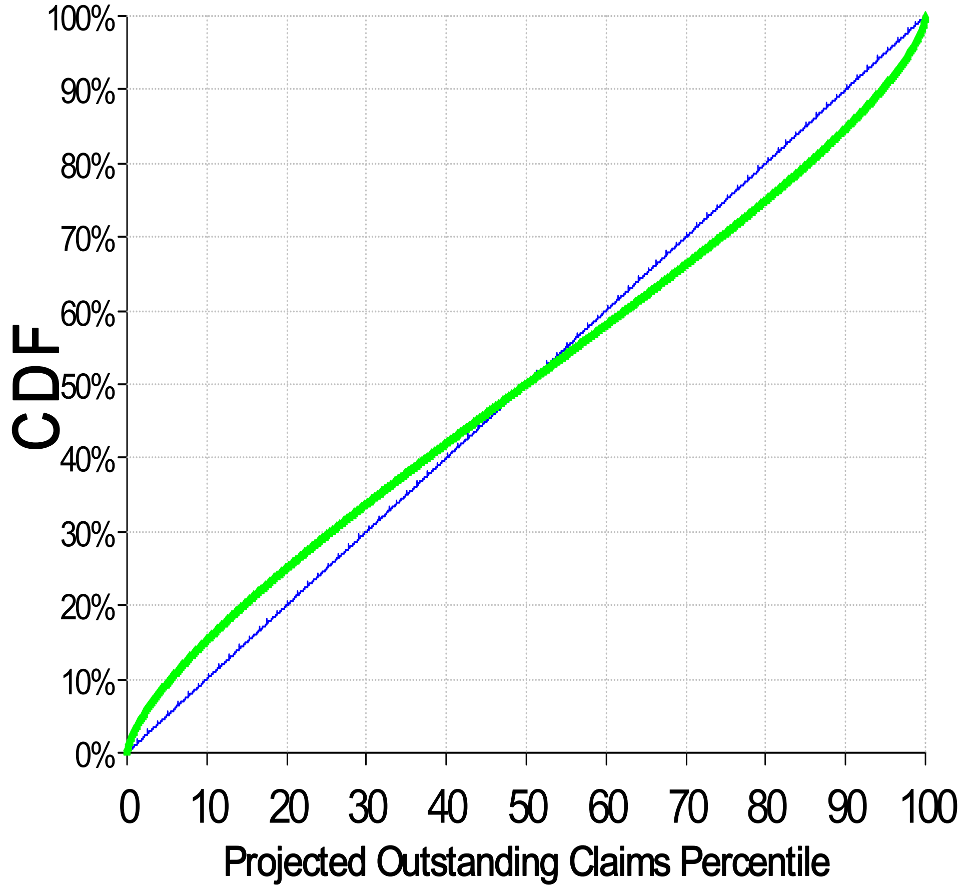
Some background on what we'd *expect* to see

- According to the test the CDF of the proportion of "true" outstandings below the projected percentile should be uniform, *Y*=*X* (shown in blue)
- The green line shows the impact of changing from the true distribution (assumed to be Normal(8,1) for this simple example, to one with a biased mean above the true mean, with the correct standard deviation, (assumed to be Normal(8.2,1) for this simple example)



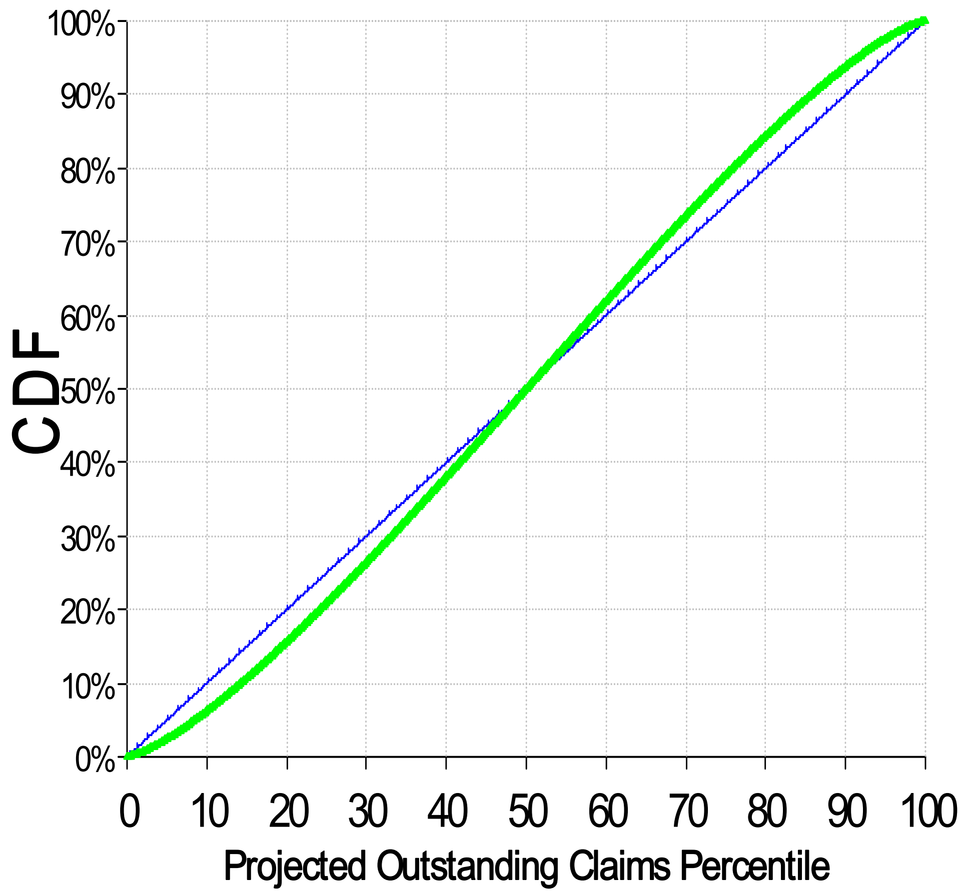
Some background on what we'd *expect* to see

- According to the test the CDF of the proportion of "true" outstandings below the projected percentile should be uniform, *Y*=*X* (shown in blue)
- The green line shows the impact of changing from the true distribution (assumed to be Normal(8,1) for this simple example), to one with the correct mean, but a biased standard deviation below the true value, (assumed to be Normal(8,0.8) for this simple example)



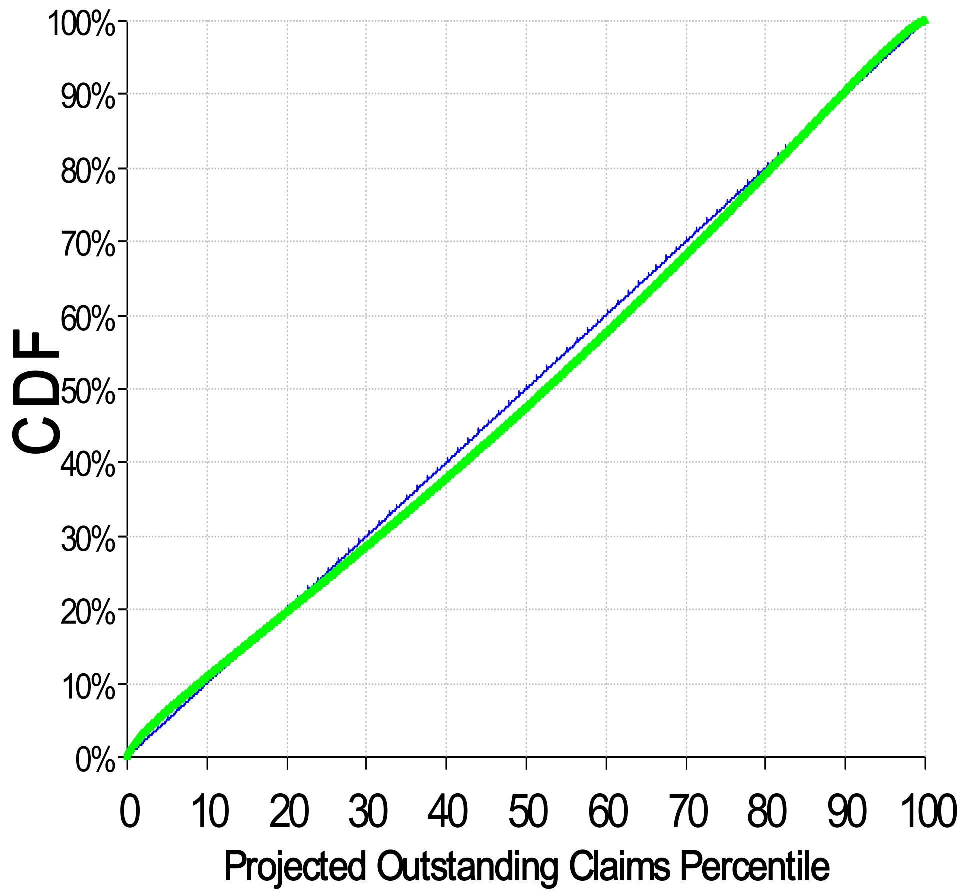
Some background on what we'd *expect* to see

- According to the test the CDF of the proportion of "true" outstandings below the projected percentile should be uniform, *Y=X* (shown in blue)
- The green line shows the impact of changing from the true distribution (assumed to be Normal(8,1) for this simple example), to one with the correct mean, but a biased standard deviation above the true value, (assumed to be Normal(8,1.2) for this simple example)



Some background on what we'd *expect* to see

- According to the test the CDF of the proportion of "true" outstandings below the projected percentile should be uniform, *Y=X* (shown in blue)
- The green line shows the impact of changing from the true distribution (assumed to be Normal(8,1) for this simple example), to one with the correct mean, and the correct standard deviation, but too much skewness, (assumed to be a Lognormal distribution with a mean of 8 and a standard deviation of 1 for this simple example)



Bootstrapping the ODP model

ROC/GIRO WP Parameters

Over-dispersed Poisson model
ROC/GIRO WP 'Algorithm B'

1. *The ultimate number of claims in an origin year is generated by random sampling from a Poisson distribution (same parameters for each origin year, but independent sampling)*
2. *Each claim is assumed to be settled by a single payment, and the development year of the payment determined by independent random sampling from a multinomial distribution (same parameters for each origin year)*
3. *The amount of each individual claim payment is determined by independent random sampling from a lognormal distribution (same parameters in each cell of the triangle)*
4. *The amounts of claims settling in the upper left triangle of the run-off array are accumulated to create this run-off triangle, and all claim amounts … are accumulated to obtain the 'true' ultimate position for each origin year.*

Notes:

The method is designed to simulate incremental claim amounts where the variance in each cell is proportional to the mean. The constant of proportionality is called the scale parameter.

The parameters used were not shown in the 2007 report, but are available in the 2008 report.

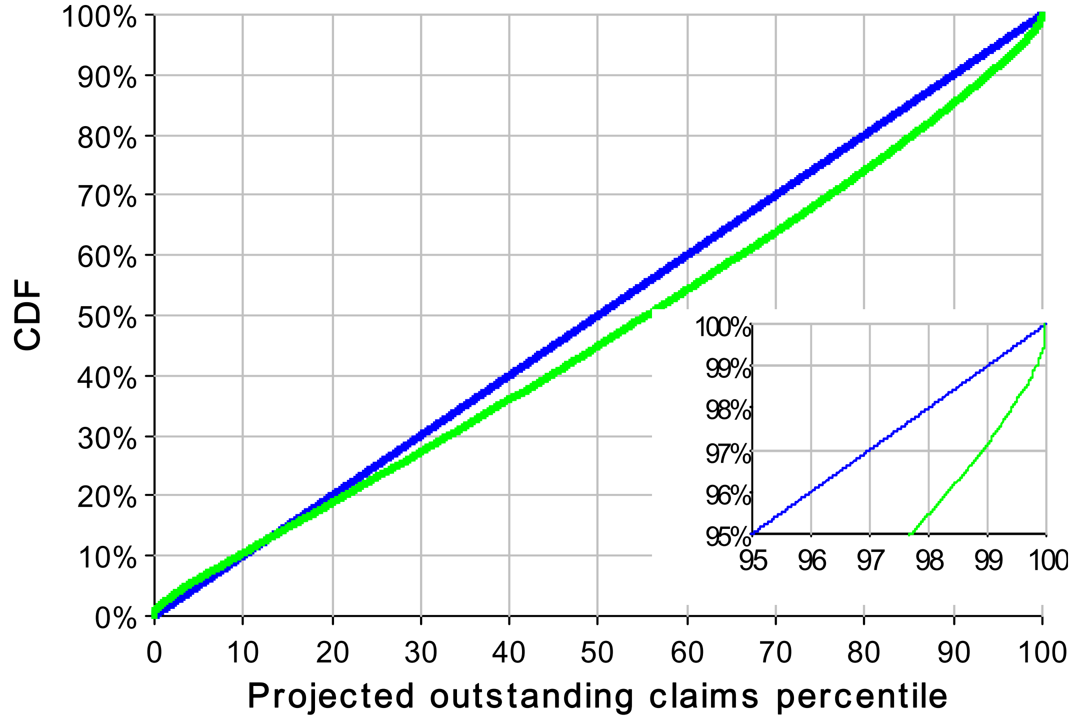
The ROC/GIRO WP only considered the special case of the ODP model with a constant scale parameter for all development years

Over-dispersed Poisson model
EMB Analysis

- EMB repeated the analysis:
  - Using Algorithm B and the WP parameters
  - Using a modified version of Algorithm B with non-constant scale parameters
  - Using an alternative data generation algorithm for the ODP with varying scale parameters, and non-parametric bootstrapping
  - Using an alternative data generation algorithm for the ODP with varying scale parameters, and parametric bootstrapping
- EMB used 30,000 generated data sets and 10,000 bootstrap iterations on each data set
  - The ROC/GIRO WP used 1000 bootstrap iterations in the 2007 report, and tested 2000 iterations in the 2008 report.

Over-dispersed Poisson model
EMB Analysis

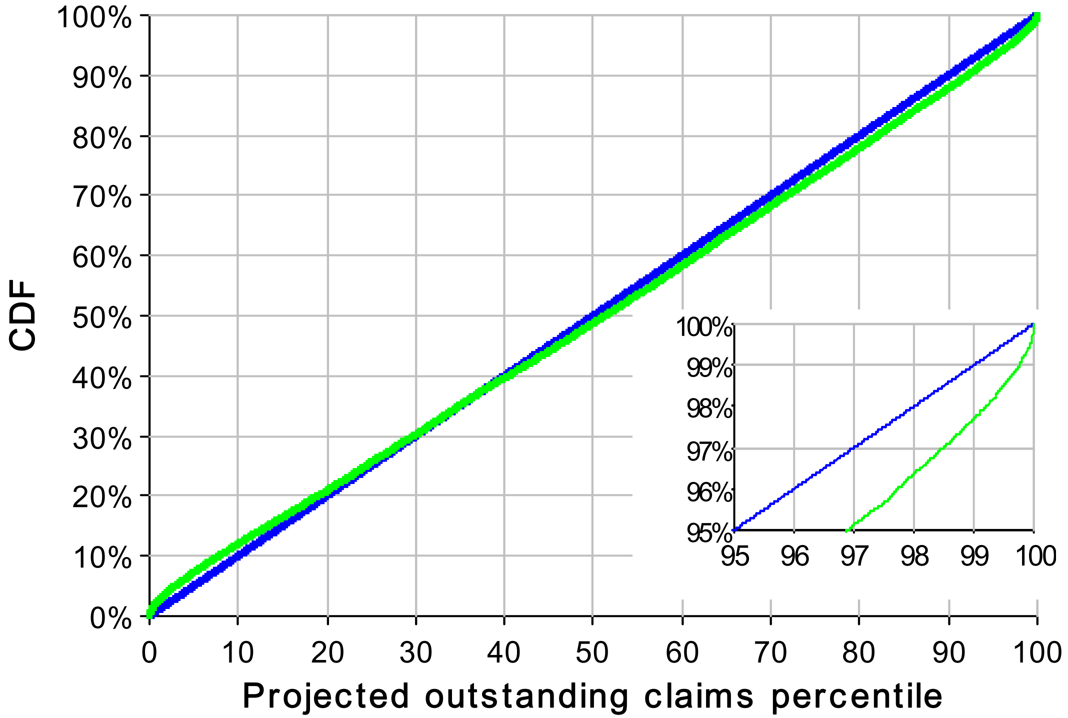
- Using Algorithm B and the WP parameters
  - We obtain similar results to the WP in the upper tails



2.86% chance of exceeding projected 99^{th} percentile

Over-dispersed Poisson model
EMB Analysis

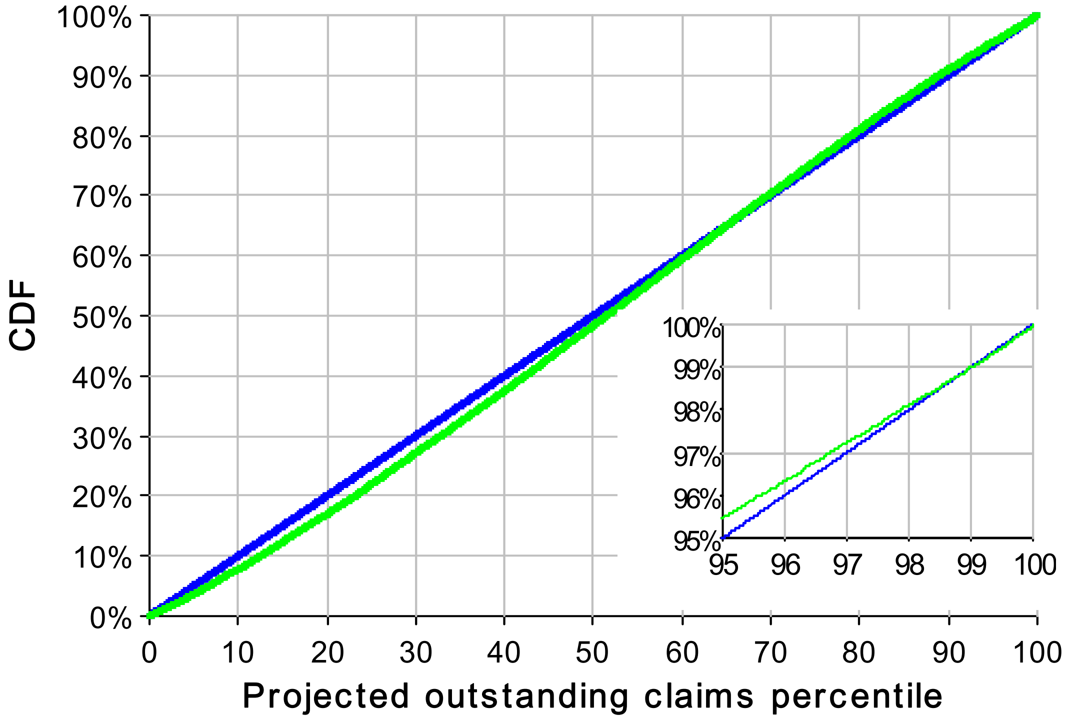
- Using a modified version of Algorithm B, with non-constant scale parameters
  - The lognormal parameters were selected such that the coefficient of variation of the sum of payments in each cell is constant for all cells, but the scale parameter depends on development period
  - The proportion exceeding the 99th percentile is now a little lower



2.32% chance of exceeding projected 99th percentile

Over-dispersed Poisson model
EMB Analysis

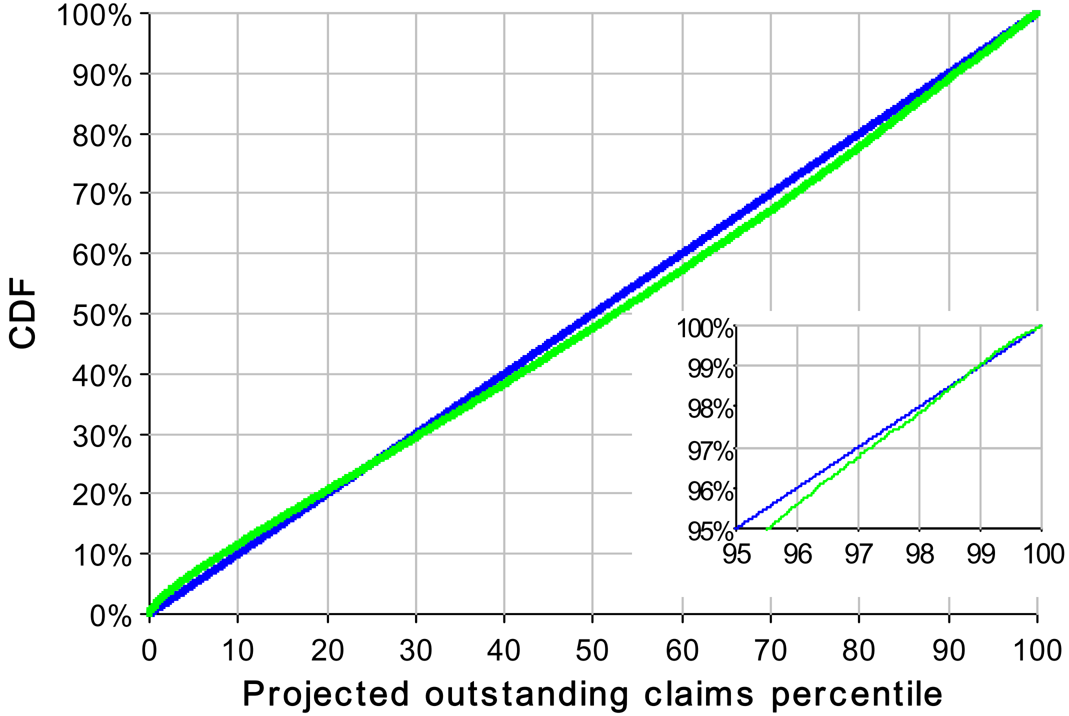
- Using an alternative algorithm for the ODP with varying scale parameters and non-parametric bootstrapping
  - Given the mean and standard deviation of total payments in each cell, simulate from a Gamma distribution with parameters selected to give the target mean and variance
  - This is a simple method and in the spirit of the ODP generalised linear model, which simply specifies the first two moments
  - [Note: we could simulate from a Poisson with expected value equal to the target mean divided by the scale parameter, then multiply the result by the scale parameter. However this gives a very ‘lumpy’ distribution.]



0.99% chance of exceeding projected 99th percentile

Over-dispersed Poisson model
EMB Analysis

- Using an alternative algorithm for the ODP with varying scale parameters and parametric bootstrapping
  - Given the mean and standard deviation of total payments in each cell, simulate from a Gamma distribution with parameters selected to give the target mean and variance
  - When applying the bootstrap procedure to each data set, estimate the residuals and scale parameters in the usual way, but instead of resampling the residuals and inverting to give pseudo data, generate the pseudo data directly from a parametric distribution, given the mean and variance characteristics. We used a Gamma distribution for this purpose.
  - Note that the scale parameters (and hence residuals) are still required when using the parametric bootstrap



0.96% chance of exceeding projected 99th percentile

Bootstrapping the ODP model

EMB Data generation method

Taylor & Ashe Data

Taylor & Ashe Data
Observed incremental values

	1	2	3	4	5	6	7	8	9	10
1	357,848	766,940	610,542	482,940	527,326	574,398	146,342	139,950	227,229	67,948
2	352,118	884,021	933,894	1,183,289	445,745	320,996	527,804	266,172	425,046	
3	290,507	1,001,799	926,219	1,016,654	750,816	146,923	495,992	280,405		
4	310,608	1,108,250	776,189	1,562,400	272,482	352,053	206,286			
5	443,160	693,190	991,983	769,488	504,851	470,639				
6	396,132	937,085	847,498	805,037	705,960					
7	440,832	847,631	1,131,398	1,063,269						
8	359,480	1,061,648	1,443,370							
9	376,686	986,608								
10	344,014									

Dev Factors	**3.49061**	**1.74733**	**1.45741**	**1.17385**	**1.10382**	**1.08627**	**1.05387**	**1.07656**	**1.01772**	**1.00000**

Taylor & Ashe Data
Fitted incremental values (chain ladder model)

	1	2	3	4	5	6	7	8	9	10	Reserve
1	270,061	672,617	704,494	753,438	417,350	292,571	268,344	182,035	272,606	67,948	0
2	376,125	936,779	981,176	1,049,342	581,260	407,474	373,732	253,527	379,669	94,634	94,634
3	372,325	927,316	971,264	1,038,741	575,388	403,358	369,957	250,966	375,833	93,678	469,511
4	366,724	913,365	956,652	1,023,114	566,731	397,290	364,391	247,190	370,179	92,268	709,638
5	336,287	837,559	877,254	938,200	519,695	364,316	334,148	226,674	339,456	84,611	984,889
6	353,798	881,172	922,933	987,053	546,756	383,287	351,548	238,477	357,132	89,016	1,419,459
7	391,842	975,923	1,022,175	1,093,189	605,548	424,501	389,349	264,121	395,534	98,588	2,177,641
8	469,648	1,169,707	1,225,143	1,310,258	725,788	508,792	466,660	316,566	474,073	118,164	3,920,301
9	390,561	972,733	1,018,834	1,089,616	603,569	423,113	388,076	263,257	394,241	98,266	4,278,972
10	344,014	856,804	897,410	959,756	531,636	372,687	341,826	231,882	347,255	86,555	4,625,811

Total 18,680,856

Taylor & Ashe Data

Scaled residuals : ODP with constant scale parameter

	1	2	3	4	5	6	7	8	9	10
1	0.737	0.501	-0.488	-1.359	0.742	2.272	-1.027	-0.430	-0.379	0.000
2	-0.171	-0.238	-0.208	0.570	-0.775	-0.591	1.099	0.110	0.321	
3	-0.585	0.337	-0.199	-0.094	1.008	-1.760	0.903	0.256		
4	-0.404	0.889	-0.804	2.325	-1.704	-0.313	-1.142			
5	0.804	-0.688	0.534	-0.759	-0.090	0.768				
6	0.310	0.260	-0.342	-0.799	0.939					
7	0.341	-0.566	0.471	-0.125						
8	-0.701	-0.436	0.860							
9	-0.097	0.061								
10	0.000									

	1	2	3	4	5	6	7	8	9	10
Scale^0.5	**229.3**	**229.3**	**229.3**	**229.3**	**229.3**	**229.3**	**229.3**	**229.3**	**229.3**	**229.3**

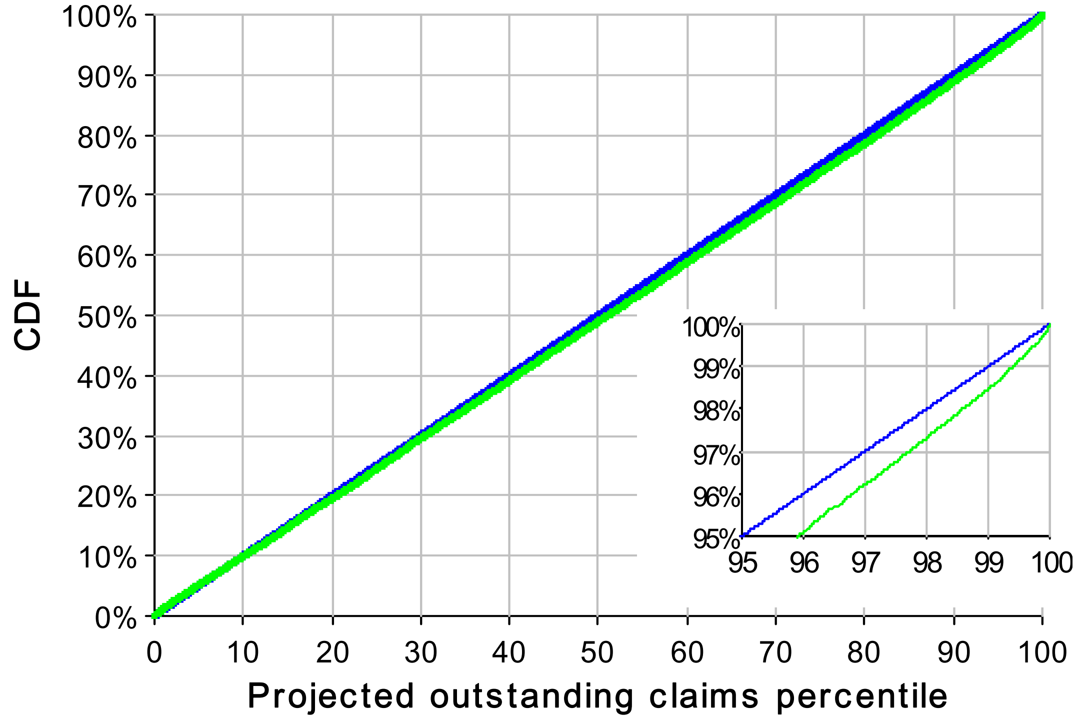
Taylor & Ashe Data

Scaled residuals : ODP with constant scale parameter

Note that the volatility is lower at the earlier and later development periods

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

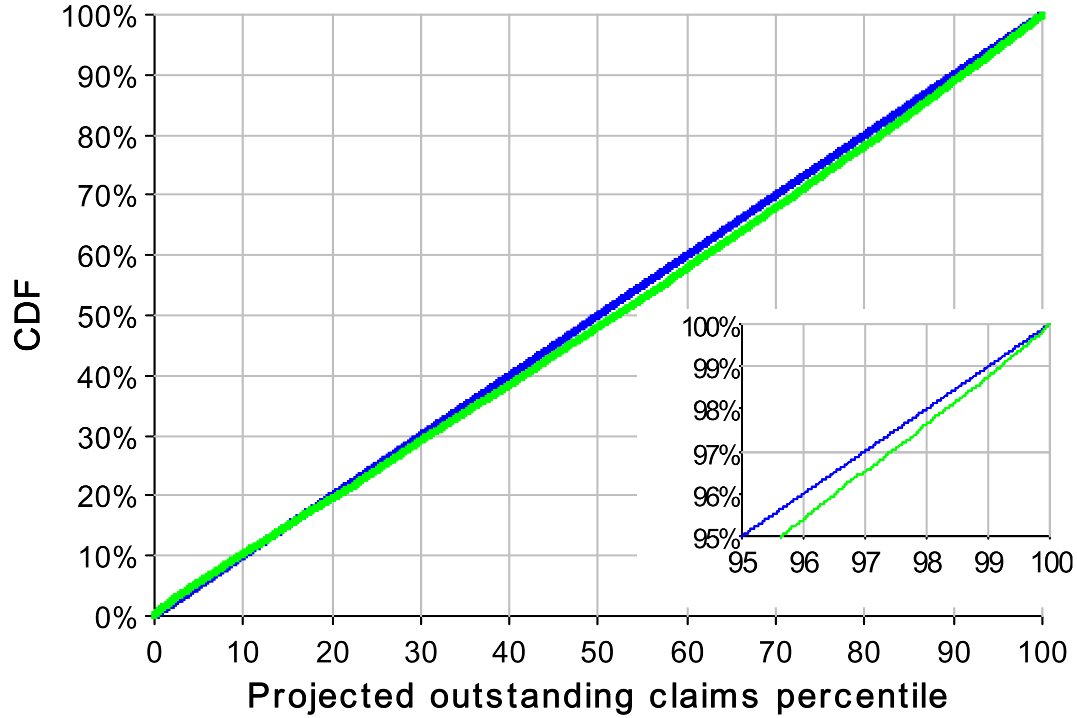
- Results using:
  - constant scale parameter
  - non-parametric bootstrapping



1.54% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

- Results using:
  - constant scale parameter
  - parametric bootstrapping



1.25% chance of exceeding projected 99th percentile

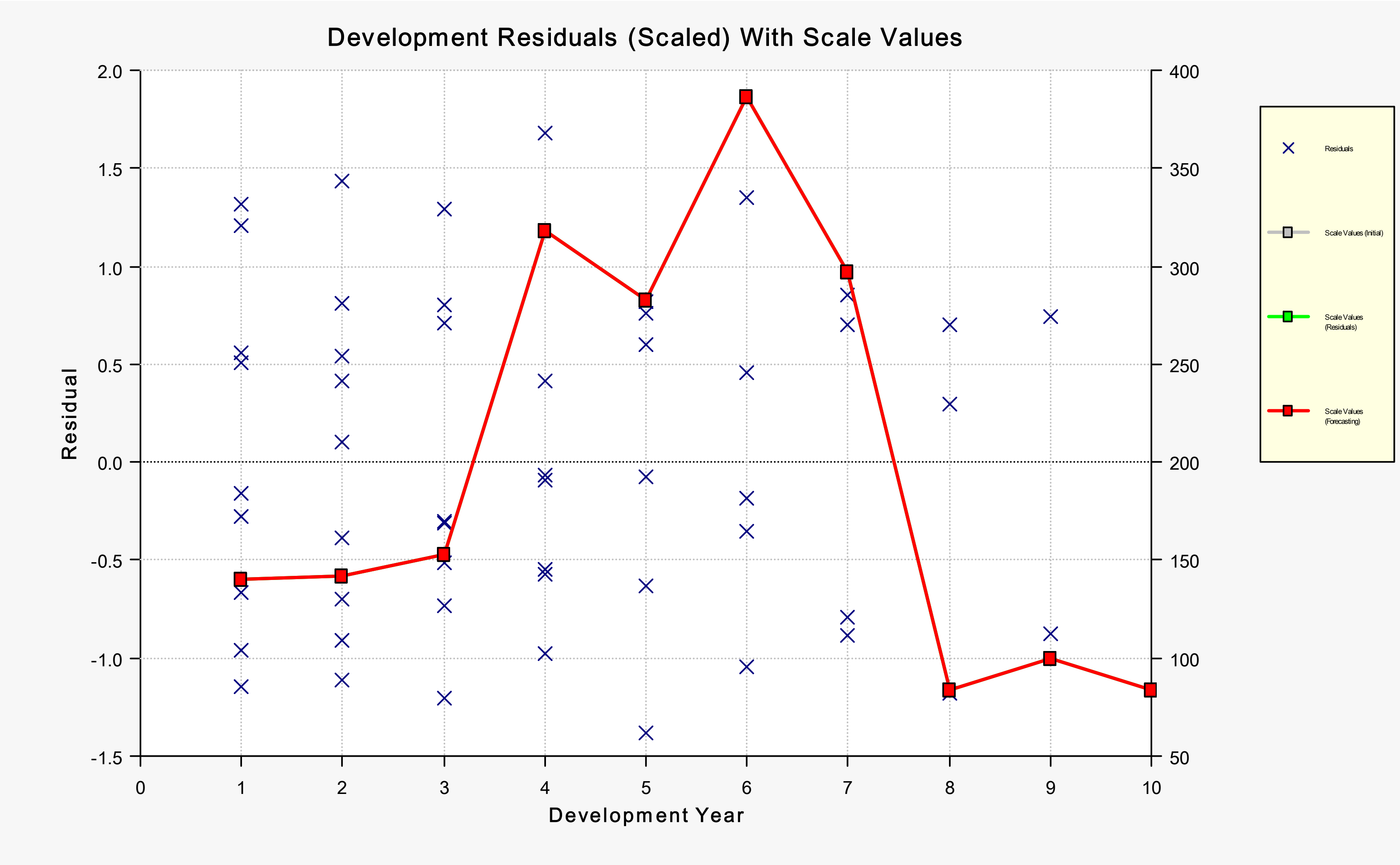
Taylor & Ashe Data

Scaled residuals : ODP with non-constant scale parameter

	1	2	3	4	5	6	7	8	9	10
1	1.207	0.808	-0.731	-0.980	0.602	1.348	-0.794	-1.176	-0.873	0.000
2	-0.280	-0.383	-0.312	0.411	-0.629	-0.350	0.849	0.299	0.740	
3	-0.958	0.544	-0.299	-0.068	0.818	-1.045	0.698	0.701		
4	-0.662	1.433	-1.206	1.676	-1.383	-0.186	-0.883			
5	1.317	-1.109	0.800	-0.548	-0.073	0.456				
6	0.509	0.419	-0.513	-0.576	0.762					
7	0.559	-0.913	0.706	-0.090						
8	-1.149	-0.702	1.288							
9	-0.159	0.099								
10	0.000									
Scale^0.5	**139.9**	**142.3**	**153.0**	**318.1**	**282.6**	**386.6**	**296.7**	**83.9**	**99.6**	**83.9**

Taylor & Ashe Data

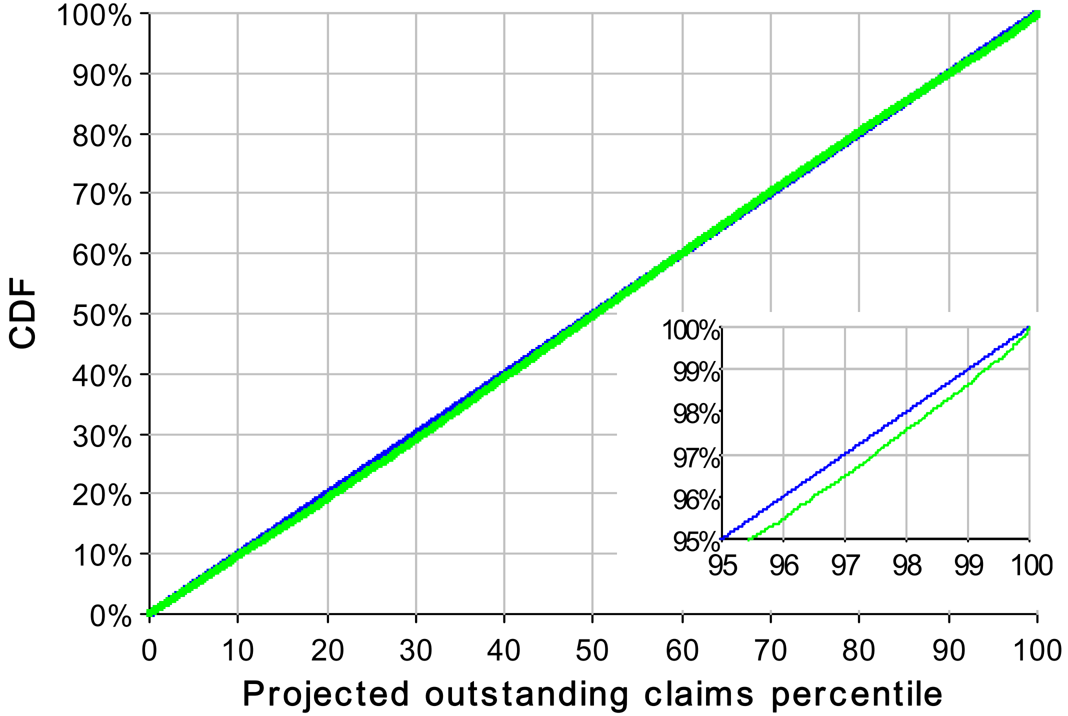
Scaled residuals: ODP with non-constant scale parameter



Note that the residuals are standardised better when using non-constant scale parameters

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

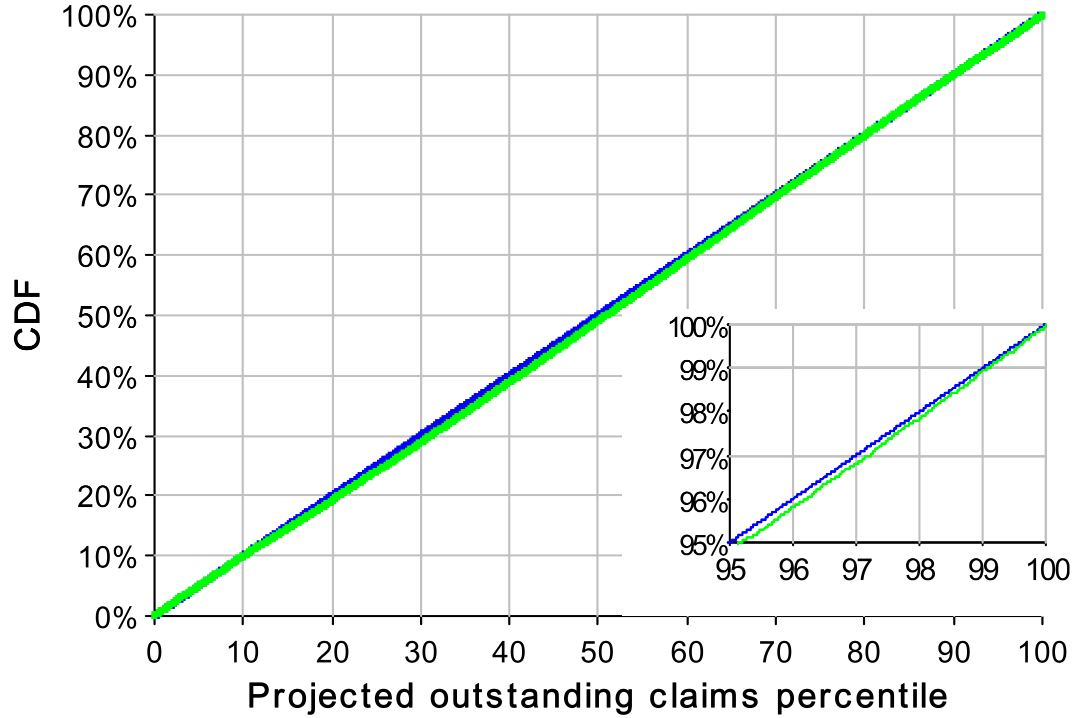
- Results using:
  - non-constant scale parameter
  - non-parametric bootstrapping



1.37% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

- Results using:
  - non-constant scale parameter
  - parametric bootstrapping



1.06% chance of exceeding projected 99th percentile

Taylor & Ashe Data
A note on the data generation method

It should be obvious that given the data generation method, the standard deviation of the simulated 'true' forecast should match the process error calculated analytically for the same model. This was checked:

ODP with constant scale

Year	Analytic	Simulated
2	70,554	70,553
3	157,153	157,098
4	193,204	193,291
5	227,610	227,703
6	273,250	272,132
7	338,448	338,832
8	454,107	455,043
9	474,426	474,696
10	493,279	492,863
Total	**976,199**	**987,979**

ODP with non-constant scale

Year	Analytic	Simulated
2	25,802	25,802
3	66,216	66,206
4	77,830	77,870
5	187,017	186,976
6	306,709	306,114
7	390,567	390,542
8	561,627	561,663
9	534,951	535,090
10	519,047	519,142
Total	**1,076,408**	**1,078,480**

Over-dispersed Poisson model
Summary of Findings

- Using the ROC/GIRO WP parameters and test procedure, the proportion of 'true' values above the 99th percentile are:
  - ODP constant scale, non-parametric, "Algorithm B": 2.86%
  - ODP non-constant scale, non-parametric, modified "Algorithm B": 2.32%
  - ODP non-constant scale, non-parametric, EMB data generation method: 0.99%
  - ODP non-constant scale, parametric, EMB data generation method: 0.96%

- Using the Taylor & Ashe data, EMB data generation method, the proportion of 'true' values above the 99th percentile are:
  - ODP constant scale, non-parametric: 1.54%
  - ODP constant scale, parametric: 1.25%
  - ODP non-constant scale, non-parametric: 1.37%
  - ODP non-constant scale, parametric: 1.06%

Over-dispersed Poisson model
Recommendations 1

- For the ODP model, we recommend using non-constant scale parameters
  - This is analogous to using varying *alpha* parameters in Mack's model
  - Some people have tried to avoid the heteroskedasticity issues by sampling residuals from different sections of the residuals triangle, but this is not ideal – a fundamental principle of bootstrapping is that the "observations" being re-sampled are *iid,* and standardising using non-constant scale parameters is a way of achieving this.
  - England & Verrall have tried to highlight this:

"This paper only considers the special case of a constant dispersion parameter ϕ. Evidence of heteroskedasticity would require extensions to the approach … The dispersion parameters would need to be included in the residual definition …" - England (2002)

"The restriction that the scale parameter is constant for all observations can be relaxed. It is common to allow the scale parameters to depend on development period …"

"We recommend using non-constant scale parameters (or at least checking the assumption that using a constant scale parameter is appropriate)…" - England & Verrall (2006)

Over-dispersed Poisson model
Recommendations 2

- We also recommend investigating parametric bootstrapping
  - In some ways, the common approach of using a non-parametric approach for bootstrapping (parameter uncertainty), then a parametric distribution at the forecasting stage (to include process uncertainty) is a little inconsistent.
  - Using non-parametric bootstrapping of residuals, the pseudo-data extremes are naturally limited by the extremes of the residuals (although the effect of this will reduce as triangle size increases)
  - Using an appropriate parametric approach, this issue is ameliorated
  - [For a given triangle however, there is no guarantee that a parametric approach will be more extreme]
  - For further information, see Björkwall *et al* 2008.

ROC/GIRO WP Findings
Over-dispersed Poisson model

- Another issue mentioned by the WP was the possibility of obtaining negative pseudo incremental values when using non-parametric bootstrapping (resampling residuals), which could in turn lead to negative pseudo cumulative values.
- This is a known issue with non-parametric bootstrapping. For example:

"Although the [non-parametric] bootstrap/ simulation procedure provides prediction errors that are consistent with their analytic counterparts, the predictive distribution produced in this way might have some undesirable properties. For example, for some origin year reserves, the minimum values of the predictive distribution could be negative." [8]

"It [non-parametric bootstrapping] is not without its difficulties, for example: a small number of sets of pseudo data may be incompatible with the underlying model…" [9]

$$\text{If } r_{ij} = \frac{Y_{ij} - \hat{m}_{ij}}{\sqrt{\hat{\phi}_j \hat{m}_{ij}}} \text{ then } Y_{ij}^B = r_{ij}^B \sqrt{\hat{\phi}_j \hat{m}_{ij}} + \hat{m}_{ij}$$

$$Y_{ij}^B < 0 \text{ if } r_{ij}^B < -\sqrt{\frac{\hat{m}_{ij}}{\hat{\phi}_j}}$$

$Y =$ incremental amounts

$m =$ expected incremental amounts

$\phi =$ scale parameter

$r =$ scaled Pearson residual

This issue disappears with parametric bootstrapping

(8) England (2002)

(9) England & Verrall (2006)

Bootstrapping Mack’s model

Mack's model (1993/4)

$$E\left[C_{j,k+1} \middle| C_{jk}\right] = f_k C_{jk}$$

$$Var\left[C_{j,k+1} \middle| C_{jk}\right] = \alpha_k^2 C_{jk}$$

$$C = \text{cumulative amounts}$$

$$f = \text{development factor}$$

$$\alpha^2 = \text{scale parameter}$$

- Note there are no parameters for the variance in the first development period, so is it really a model of the cumulative amounts?
- Note also that the cumulative value in the first development period for the most recent origin period (bottom left of a run-off triangle) contributes nothing to the parameter estimation. It is only required to set a level for forecasting
- In practice, the α_k parameters decrease rapidly as the cumulative values increase by development period

Mack's model
ROC/GIRO WP 'Algorithm A'

1. *Decide on values of the parameters f_k and α_k: the values we used are given in Table B-1*
2. *For each origin year j, generate a value for C_{j1} (representing the amount paid in the first development year). Mack's assumptions say nothing about how these values are generated so we are free to use any method. We used random sampling from a lognormal distribution: the same lognormal distribution for all years (mean = variance =1) but independent random sampling for each origin year.*
3. *For each origin year, generate C_{jk} (for k>1) recursively using Mack's assumptions. We generated $C_{j,2}$ from a shifted lognormal distribution that gives values greater than $C_{j,1}$, with mean equal to $f_1C_{j,1}$ and variance equal to $\alpha_1^2C_{j1}$… We continued recursively in this way … until we obtained a value for $C_{j,10}$, which is the 'true' ultimate figure for origin year j…*
4. *The triangle was then constructed by discarding the lower right part of the development array (the $C_{j,10}$ values were kept as the 'true' ultimates for comparison with estimates produced by applying Mack's method to the upper left triangle)*

Notes:

The items highlighted in red are heroic assumptions.

Table B-1									
Dev yr *k*	1	2	3	4	5	6	7	8	9
f(*k*)	4.289	2.064	1.502	1.268	1.15	1.085	1.048	1.027	1.015
alpha(*k*)	1	1	1	1	1	1	1	1	1

Note that the α parameters are constant, but in practice they decrease rapidly

Mack's model
EMB Analysis

- Instead of trying to explain where "Mack's method goes wrong", we focussed on the alternative explanation that Algorithm A is wrong
- Understanding Mack's model is not straightforward however
- It is essential to consider the estimation and forecasting components separately
- For estimation, it is actually a model of the ratios (see Mack 1999, England & Verrall 2002, and England & Verrall 2006)

$$E\left[F_{j,k} \middle| C_{jk}\right] = f_k$$

$$Var\left[F_{j,k} \middle| C_{jk}\right] = \frac{\alpha_k^2}{C_{jk}}$$

$$F_{j,k} = \frac{C_{j,k+1}}{C_{j.k}}$$

Mack's model re-formulated

$$E\left[F_{j,k} \middle| C_{jk}\right] = f_k$$

$$Var\left[F_{j,k} \middle| C_{jk}\right] = \frac{\alpha_k^2}{C_{jk}}$$

$$F_{j,k} = \frac{C_{j,k+1}}{C_{j.k}}$$

Writing $w_{jk} = C_{jk}$

$$\hat{f}_k = \frac{\sum_{j=1}^{n-k} w_{jk} F_{jk}}{\sum_{j=1}^{n-k} w_{jk}}$$

$$\hat{\alpha}_k^2 = \frac{1}{n-k-1} \sum_{j=1}^{n-k} w_{jk} \left(F_{jk} - \hat{f}_k\right)^2$$

- As a model of the ratios, there is no need for a variance parameter *α* for the cumulatives in the first development period
- As a GLM, Mack's estimators for *f* and *α* are obtained assuming the ratios *f* are normally distributed with weights *w* (see England & Verrall 2002/2006)
- In a GLM context, the weights are considered fixed and known

Mack's model
EMB data generation process (using the Taylor & Ashe data)

1. *Given the Taylor & Ashe data, obtain parameters f_k and α_k, and expected cumulative values $\hat{C}_{jk}$*
2. *Use the expected cumulative values as weights w_{jk} for j=1..n and development year k=1..n-j+1. The weights are then considered fixed and known.*
3. *For estimation: For each origin year j for j=1..n-1 and development year k=1..n-j, simulate ratios F_{jk} assuming:*

$$F_{jk} \sim Normal\left(f_k, \frac{\alpha_k}{\sqrt{w_{jk}}} \right)$$

4. *For forecasting: For each origin year, generate C_{jk} (for k>n-j+1) recursively using Mack's assumptions. We generated $C_{j,k+1}$ from a normal distribution, with mean equal to $f_k C_{j,k}$ and variance equal to $\alpha_k^2 C_{j,k}$. We continued recursively in this way, until we obtained a value for $C_{j,10}$, which is the 'true' ultimate figure for origin year j. Note that $C_{j,n-j+1}$ is deterministic, and $C_{10,1}$ is only ever used at the forecasting stage. The $C_{j,10}$ values were kept as the 'true' ultimates for comparison with estimates produced by bootstrapping Mack's method in step 5.*

5. *The triangle of ratios F_{jk} together with the known weights w_{jk} were then used in the bootstrapping procedure, where the volume weighted average ratios and alpha parameters were re-estimated using the known weights w_{jk} before bootstrapping using England & Verrall (2006). Forecasting after the bootstrap procedure was performed recursively as above with $C_{j,n-j+1}$ given by $w_{j,n-j+1}$. 10,000 iterations were used at this step. The percentile of the 'true' ultimate was recorded.*
6. *Repeat steps three to five 30,000 times.*

The analysis was also repeated using Gamma distributions for simulating the ratios and forecast cumulative values, parameterised to ensure the means and variances remain the same. This ensures that negative ratios and cumulative values can never be simulated.

Mack's model
Comments on the data generation process

- Given the simulated ratios, it is possible to generate a triangle of simulated cumulative amounts by starting with the latest cumulative diagonal, and recursively dividing backwards by the simulated ratios
  - Notice that this gives simulated amounts in the first development period, without additional parameters
  - Notice also that the incremental amounts in each cell are different for each simulation (except in the bottom left cell of the run-off triangle), but the sum of simulated incremental values in the triangle always equals the latest observed cumulative diagonal
- However, passing this simulated triangle forwards to the bootstrapping stage yields biased results, unless the assumed known weights are also passed forwards
  - Note the simulated ratios will be preserved when re-creating them from the simulated cumulative values, but the 'weights' will be different if they are based on the simulated cumulative values
  - As such, simulating cumulative values in this way adds nothing

Mack’s model
Comments on EMB data generation process

- Admittedly, separating estimation from forecasting and treating the weights as fixed and known seems strange, but this appears to be consistent with Mack’s method.
  - The variances of the ratios are consistent with Mack’s model.
- Forecasting recursively from a deterministic diagonal also seems strange at first sight, but notice that the standard deviation of the simulated forecast ‘true’ reserves matches the process error from Mack’s method, as we would expect (see the analogous results for the ODP data generation method)

Year	Analytic	Simulated Normal	Simulated Gamma
2	48,832	48,832	48,832
3	90,524	90,515	90,515
4	102,622	102,691	102,693
5	227,880	227,653	227,665
6	366,582	366,098	366,079
7	500,202	499,748	499,736
8	785,741	784,941	784,898
9	895,570	894,441	894,427
10	1,284,882	1,287,905	1,288,171
Total	**1,878,292**	**1,882,547**	**1,883,336**

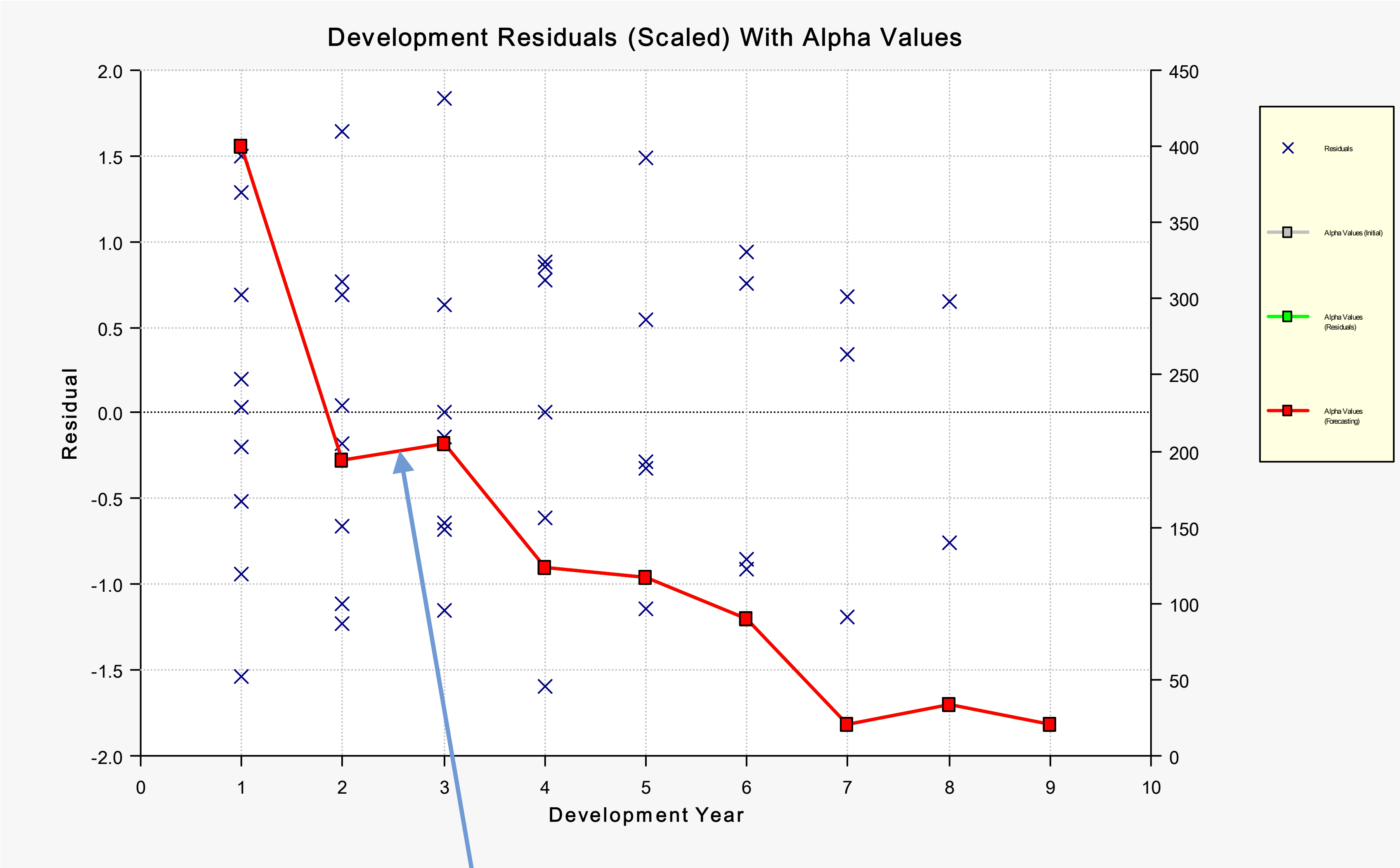
Taylor & Ashe Data
Scaled residuals: Mack's model

	1	2	3	4	5	6	7	8	9
1	-0.519	-1.117	-1.152	0.772	1.490	-0.850	-1.189	-0.759	0.000
2	0.030	0.047	0.632	-0.608	-0.322	0.939	0.346	0.651	
3	1.290	-0.179	0.006	0.850	-1.141	0.758	0.682		
4	1.500	-1.228	1.840	-1.594	-0.282	-0.907			
5	-1.540	0.689	-0.683	0.005	0.543				
6	-0.197	-0.664	-0.636	0.878					
7	-0.942	0.764	-0.137						
8	0.693	1.647							
9	0.197								
Mack's alpha	**400.4**	**194.3**	**204.9**	**123.2**	**117.2**	**90.5**	**21.1**	**33.9**	**21.1**

Note that the α parameters decrease rapidly

Taylor & Ashe Data

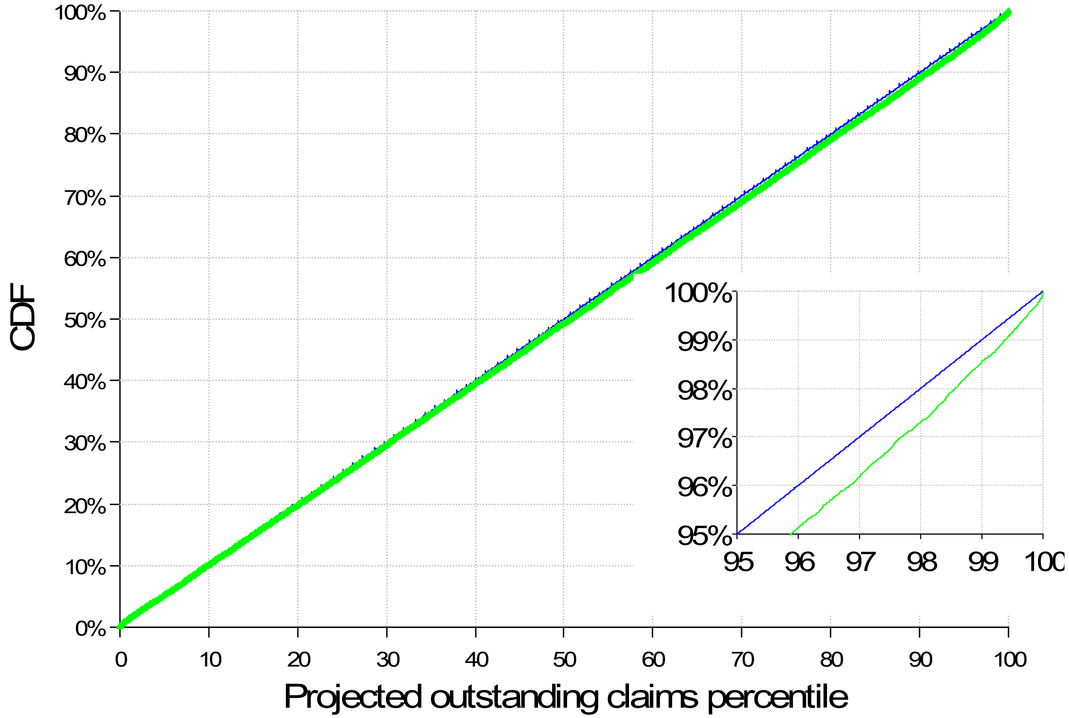
Scaled residuals: Mack's model



Note that the α parameters decrease rapidly

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

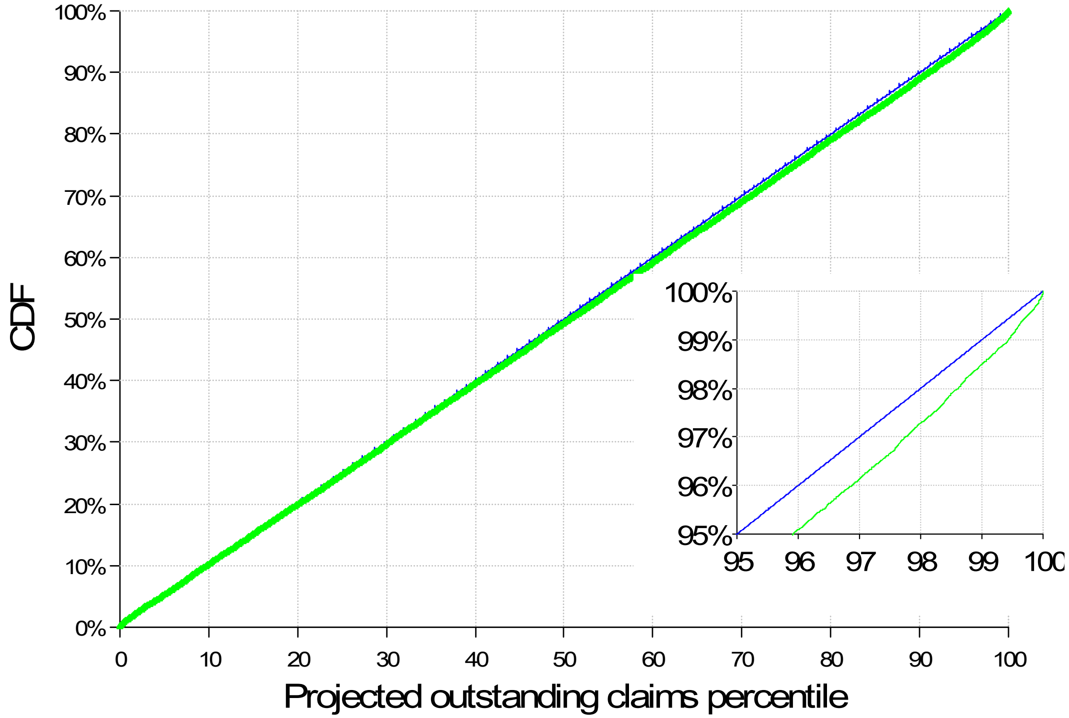
- Results using:
  - Normal forecast distributions
  - Non-parametric bootstrap



1.46% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

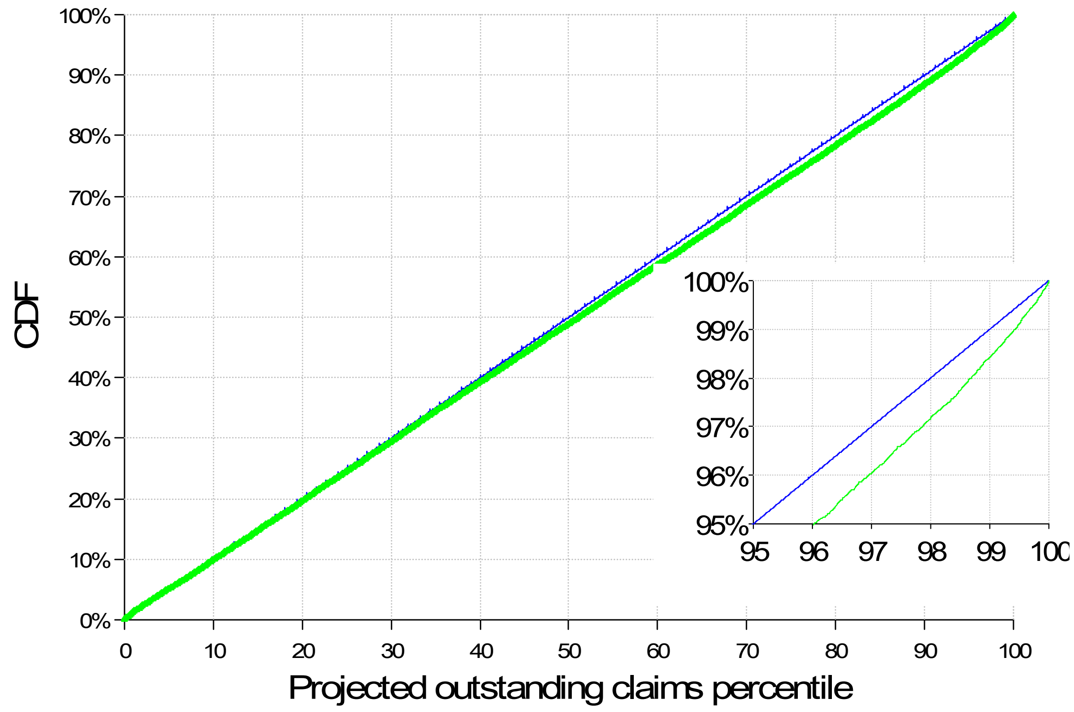
- Results using:
  - Normal forecast distributions
  - Parametric bootstrap



1.50% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

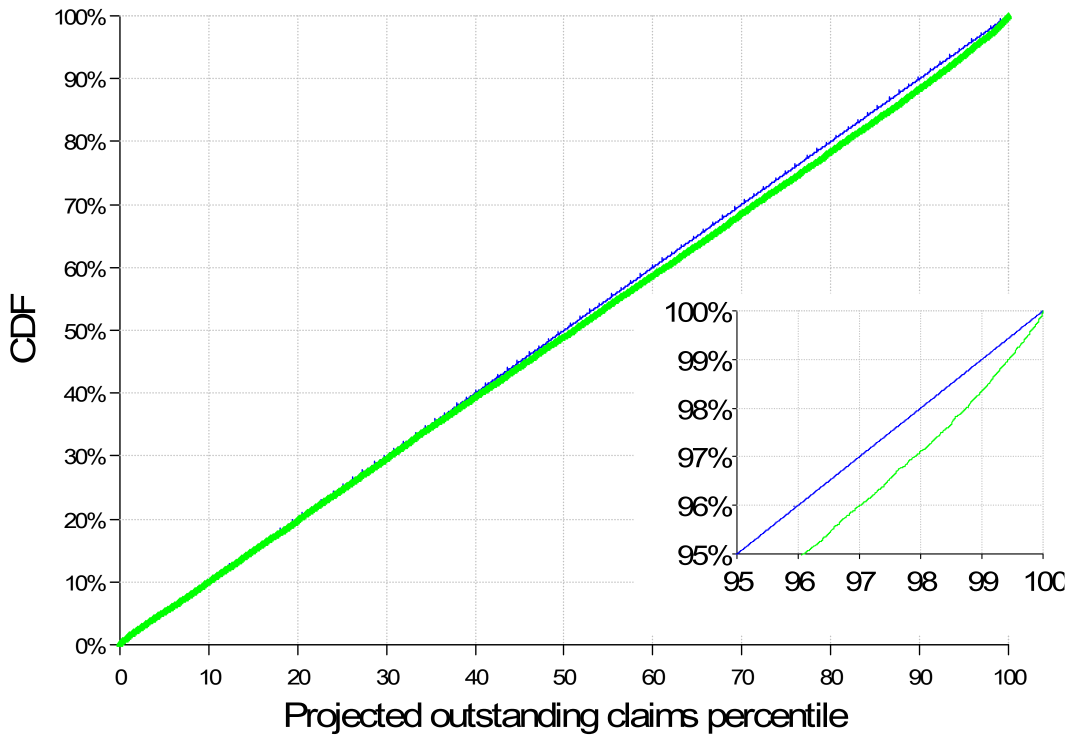
- Results using:
  - Gamma forecast distributions
  - Non-parametric bootstrap



1.57% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

- Results using:
  - Gamma forecast distributions
  - Parametric bootstrap



1.65% chance of exceeding projected 99th percentile

Bootstrapping Mack's model
Comments

- The ROC/GIRO WP did not bootstrap Mack's model to obtain the percentiles of the 'true' values
  - For example, the WP estimated the standard deviation of the forecast for each simulated data set, and then assumed the reserves are lognormally distributed
- As such, we cannot compare our results directly with the WP results
- Nevertheless we believe that 'Algorithm A' as used by the ROC/GIRO WP is not consistent with Mack's model
  - This would explain why all the findings based on Algorithm A do not seem to be intuitive
  - Unfortunately, this means that ALL results based on Algorithm A in the ROC/GIRO WP reports in 2007 and 2008 cannot be relied upon
  - See also Dan Murphy's presentation at the CLRS 3 weeks ago
    - *"Where's the Beef? Does the Mack Method produce an undernourished range of possible outcomes?"*
- We believe that our alternative data generation method is closer to Mack's model, but further improvements are possible

Bootstrapping Mack's model
Recommendations

- The common practice of estimating the standard deviation of outstanding liabilities by origin period and in total using Mack's formula, then assuming the reserves are lognormally distributed, was useful as an approximation in 1993, but is now inadequate
  - Unless the dependencies between origin years are taken into account, the standard deviation of the total across all years will be wrong.
  - In fact, the problem is more complicated, since it is the distribution of cash flows in each origin/development period combination that is important (for discounting, say). Assuming these are lognormally distributed (with parameters obtained from Mack's formulae) will give incorrect standard deviations at the origin period level, and overall total level, unless the dependencies between all cash-flows are taken into account. This is not straightforward.
  - As the cash-flows are combined, there are good reasons to expect the sum to be increasingly normally distributed (not lognormally distributed), even when the distributions of cash-flows are themselves skewed, unless the dependencies between cash-flows are extremely strong.
- These issues are avoided completely when bootstrapping Mack's model (or using MCMC methods), since distributions of cash-flows are obtained automatically such that the standard deviation of the totals (by origin period and overall) matches Mack's formulae
  - The distributions of cash-flows can be used in a variety of ways, for example, for discounting, or for investigating the 1-year view of reserving risk.

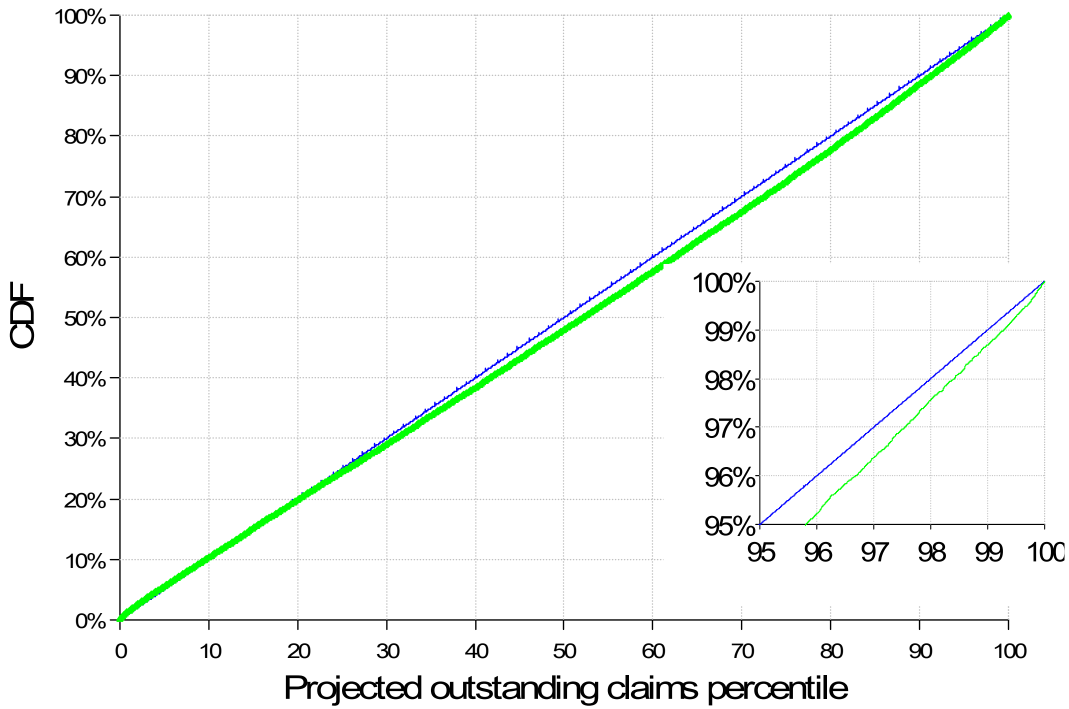
Bayesian MCMC Analysis

Bayesian MCMC Methods

- Suspecting that non-parametric bootstrapping has the potential for limiting the far extremes, we also investigated Bayesian Markov Chain Monte Carlo (MCMC) methods (using non-informative uniform priors), since these should not have the same issues.
  - See England & Verrall (2006)
- Using MCMC methods, parameters are sampled directly from the log-likelihood function
  - There is no need to generate sets of pseudo data and re-fit the model to each set to obtain a distribution of parameters
- We used exactly the same data generation method that we used for testing bootstrapping
  - 30,000 test samples
- Instead of using bootstrapping for each test sample, we used MCMC methods instead
  - Gibbs sampling with ARS
  - Adaptive Metropolis-Hastings
  - 10,000 iterations for each test sample

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

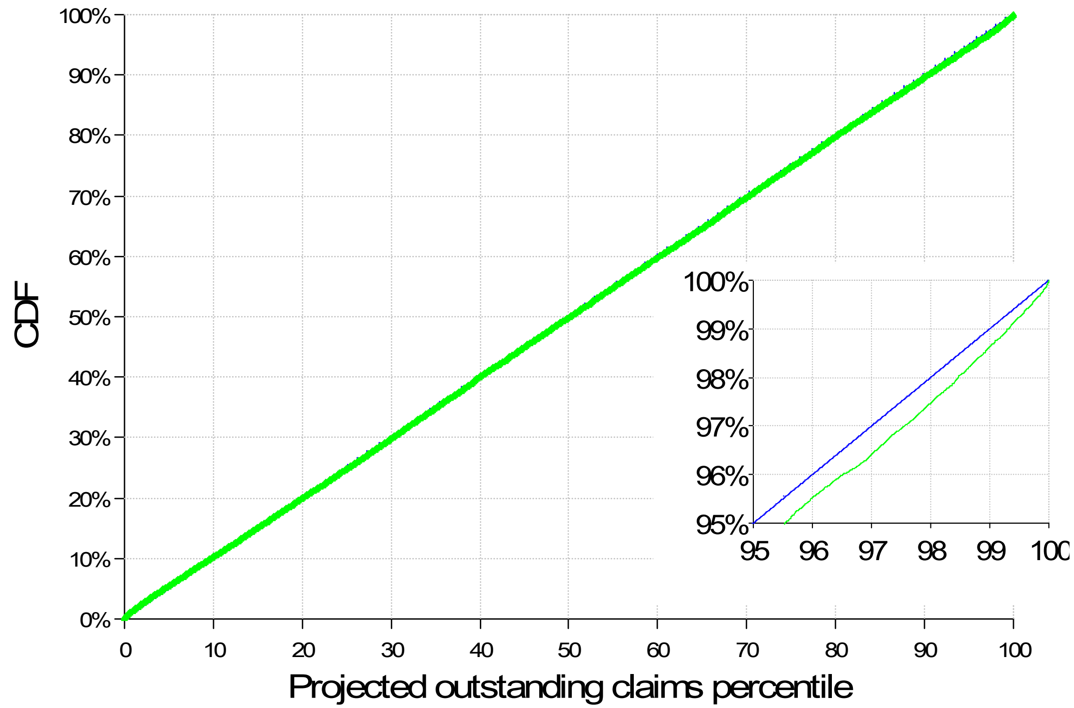
- Results using:
  - ODP model with a constant scale parameter
  - MCMC using adaptive Metropolis-Hastings



1.31% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

- Results using:
  - ODP with non-constant scale parameter
  - MCMC using Gibbs/Adaptive Rejection Sampling

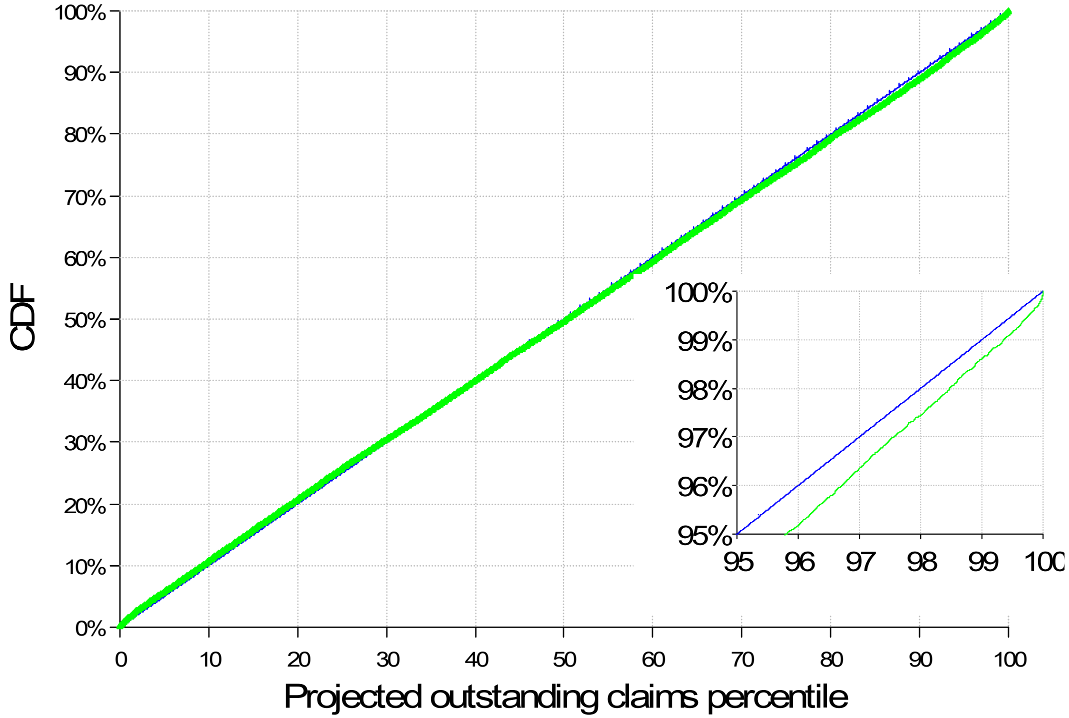


1.37% chance of exceeding projected 99th percentile

Taylor & Ashe Data

(as used in England & Verrall 1999 and 2006)

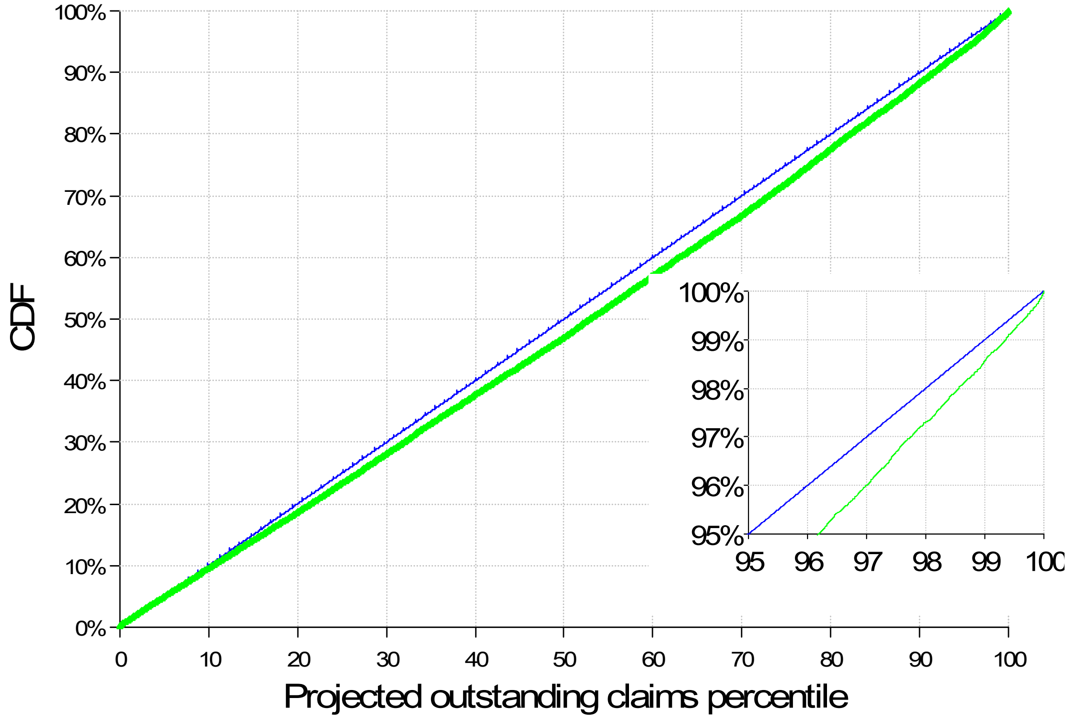
- Results using:
  - ODP model with non-constant scale parameters
  - MCMC using adaptive Metropolis-Hastings



1.38% chance of exceeding projected 99^{th} percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

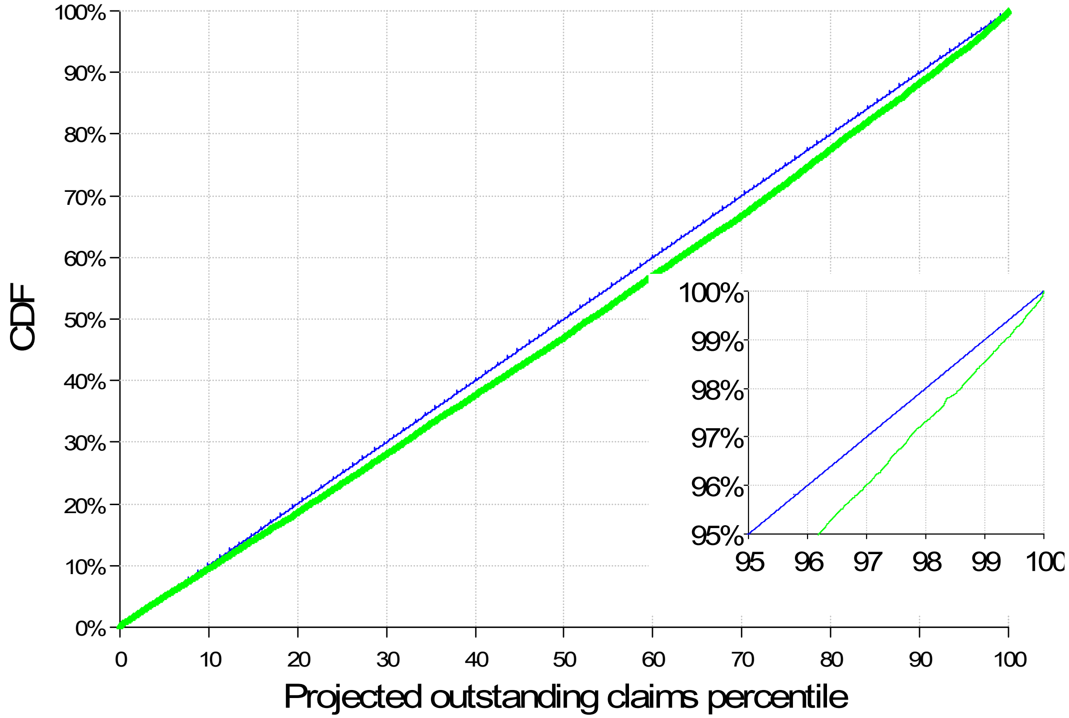
- Results using:
  - Mack's model with Normal forecast distributions
  - MCMC using Gibbs/Adaptive Rejection Sampling



1.43% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

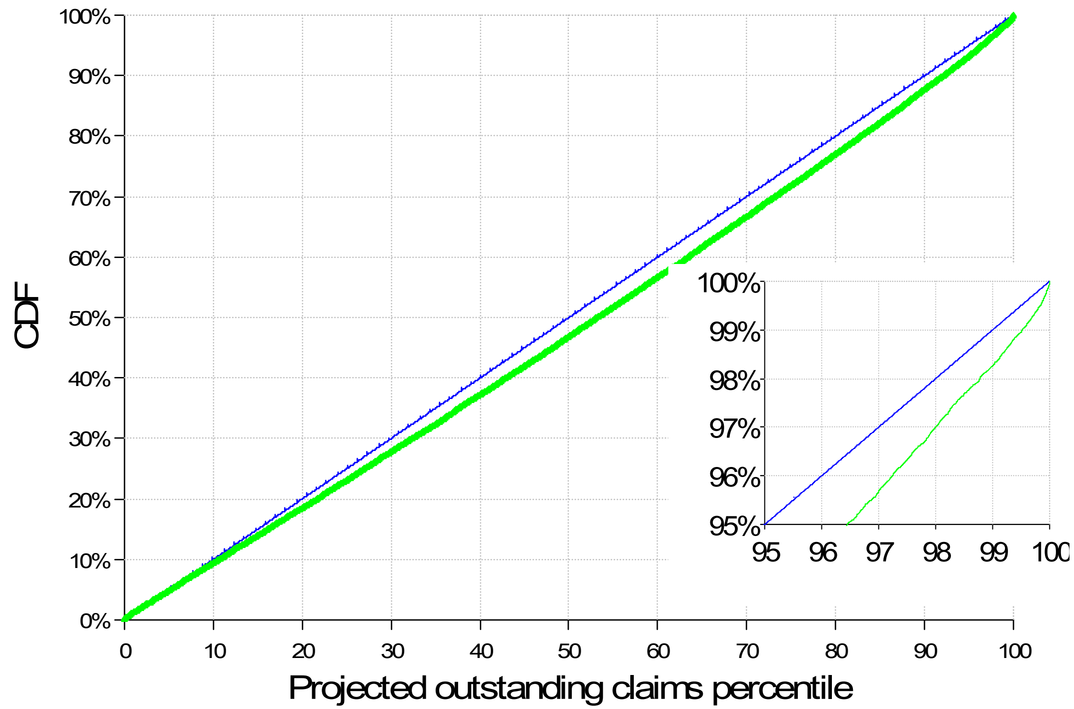
- Results using:
  - Mack's model with Normal forecast distributions
  - MCMC using adaptive Metropolis-Hastings



1.46% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

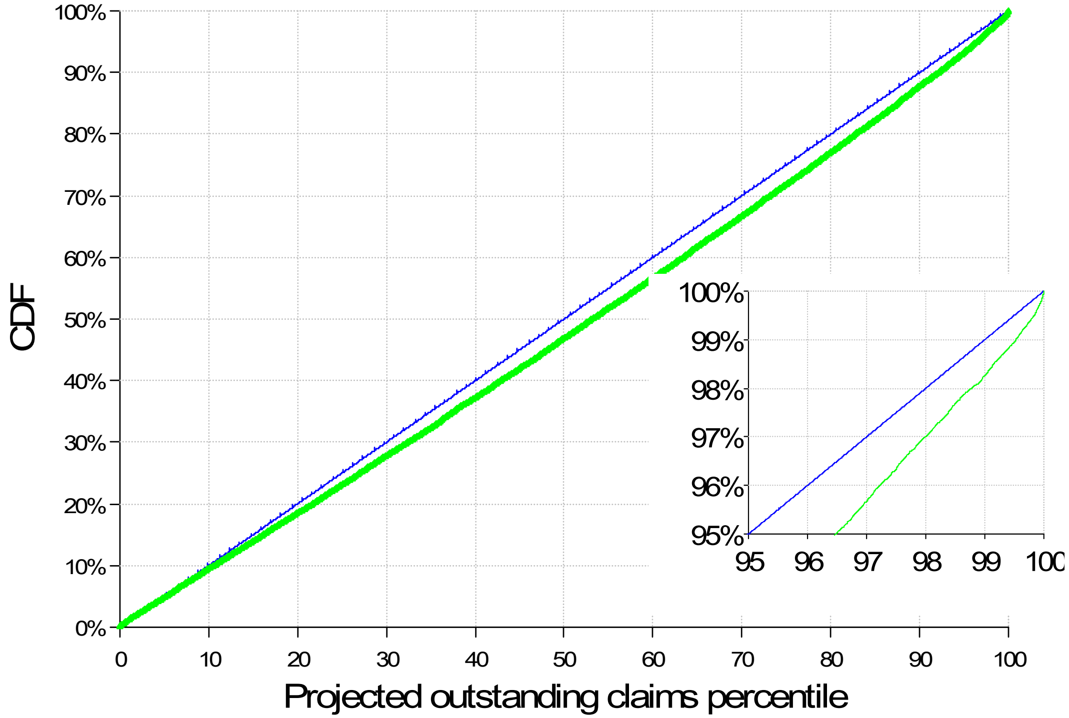
- Results using:
  - Mack's model with Gamma forecast distributions
  - MCMC using Gibbs/Adaptive Rejection Sampling



1.74% chance of exceeding projected 99th percentile

Taylor & Ashe Data
(as used in England & Verrall 1999 and 2006)

- Results using:
  - Mack's model with Gamma forecast distributions
  - MCMC using adaptive Metropolis-Hastings



1.72% chance of exceeding projected 99th percentile

Bayesian MCMC Methods
Results

- The graphs for ODP non-constant scale do, in fact, look remarkably uniform
  - About 1.38% of 'true' values are above the 99th percentile for the Taylor & Ashe data
- For Mack's model:
  - About 1.46% of 'true' values are above the 99th percentile for the Taylor & Ashe data using normal distributions
  - About 1.74% of 'true' values are above the 99th percentile for the Taylor & Ashe data using gamma distributions
- The same results are obtained regardless of MCMC algortihm used
  - Gibbs/ARS or Metropolis-Hastings
- These results are highly consistent with the analogous results using bootstrapping, suggesting that the bootstrapping procedure itself is not creating a bias at the upper percentiles
  - This hints at a possible systematic issue associated with the test procedure

Is the test procedure valid?

Is the test procedure valid?

- The test procedure is justified in the WP report as:
  - "…we can use the well known fact that if *X* is a random variable, and *F(x)* is its cumulative distribution function, then the random variable *F(x)* has a uniform distribution on the unit interval [0,1]"
  - "In the context of stochastic reserving, *X* represents the total of future claim payments… A stochastic method produces a function *F(X)* that purports to be the distribution function of *X*."
  - "If a stochastic method is reasonably good, *F(X)* should therefore have approximately a uniform distribution"
  - "Having carried out the stochastic method on a particular triangle, we can obtain one instance of the random variable *F(X)* by waiting for the triangle to reach its ultimate position: this give us one instance (x_0 say) of the random variable *X*, hence one instance $F(x_0)$ purportedly from the uniform [0,1] distribution"

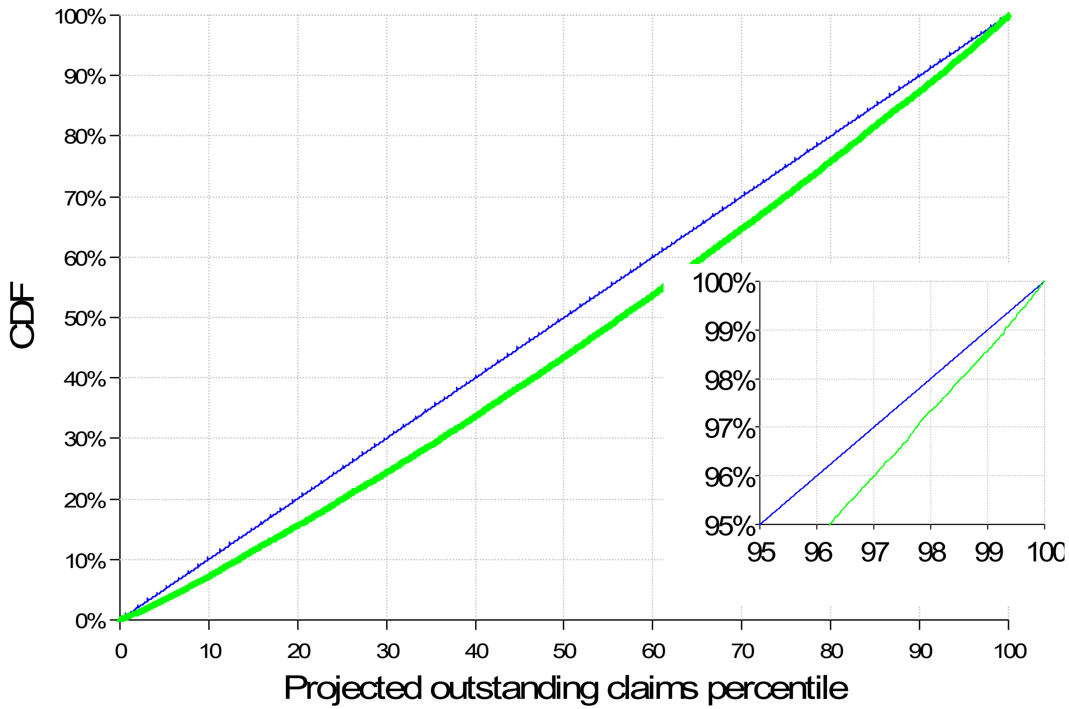
Is the test procedure valid?

- At first sight, the test procedure seems intuitively reasonable.
- So we tested it using very simple examples.
- A one parameter example.
  - Suppose observations follow a Poisson distribution with known mean μ.
  - Simulate 30,000 samples of n observations from a Poisson(μ), together with a ‘true’ forecast t, also from a Poisson(μ)
  - For each sample of n observations, estimate the mean m, and perform a parametric bootstrap:
    - Create 10,000 bootstrap samples of n observations from a Poisson(m), and for each bootstrap iteration, estimate the mean m^*, giving a distribution of m^*
  - For each bootstrap iteration, simulate a forecast value f from a Poisson(m^*), giving a distribution of the forecast f. Find the percentile (p) of the ‘true’ value t from the distribution of f.
  - This give 30,000 values of p, and according to the ROC/GIRO WP methodology, p should follow a uniform distribution, and 1% of values of p should be greater than 0.99

Is the test procedure valid?

A one parameter Poisson problem

- Results using:
  - True Poisson parameter μ=10
  - Generate 20 "data" from a Poisson(μ)
  - Calculate the mean m of the "data"
  - Use a parametric bootstrap (10,000 iterations) to generate 20 "pseudo data" from a Poisson(m) distribution, and calculate the mean m^* for each iteration
  - For each bootstrap iteration, generate a forecast from a Poisson(m^*)
  - Compare a single draw from the true Poisson(μ) distribution against the forecast Poisson(m^*) distribution
  - Repeat 30,000 times

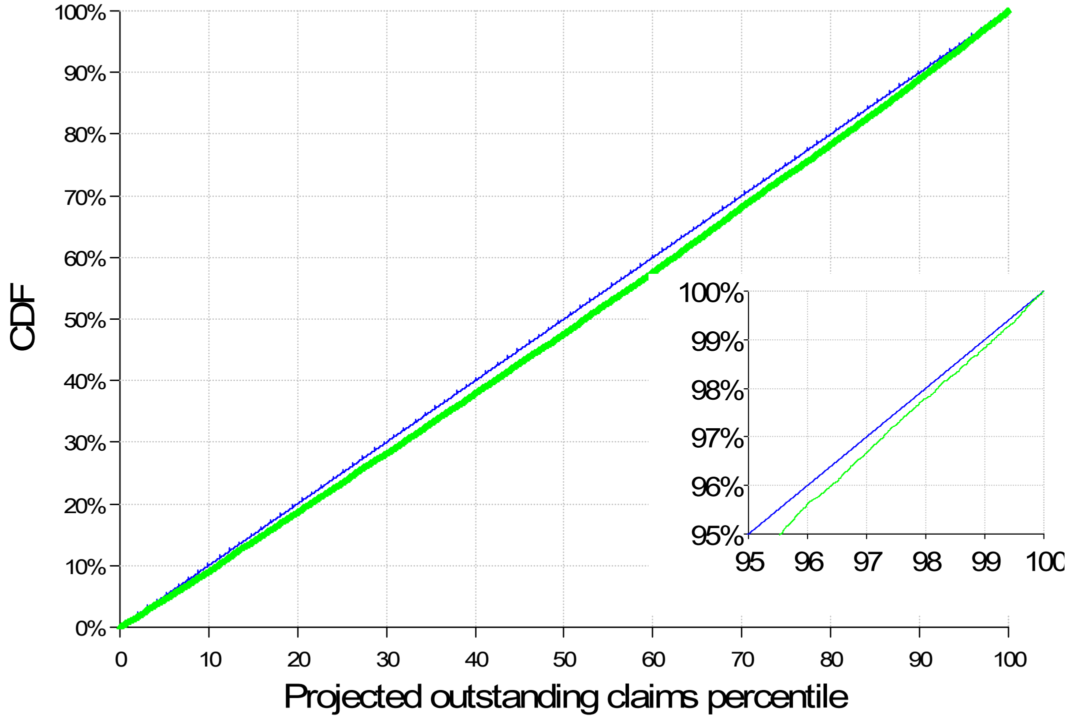


1.43% chance of exceeding projected 99th percentile

Is the test procedure valid?

A one parameter Poisson problem

- Results using:
  - True Poisson parameter μ=100
  - Generate 20 “data” from a Poisson(μ)
  - Calculate the mean m of the “data”
  - Use a parametric bootstrap (10,000 iterations) to generate 20 “pseudo data” from a Poisson(m) distribution, and calculate the mean m^* for each iteration
  - For each bootstrap iteration, generate a forecast from a Poisson(m^*)
  - Compare a single draw from the true Poisson(μ) distribution against the forecast Poisson(m^*) distribution
  - Repeat 30,000 times

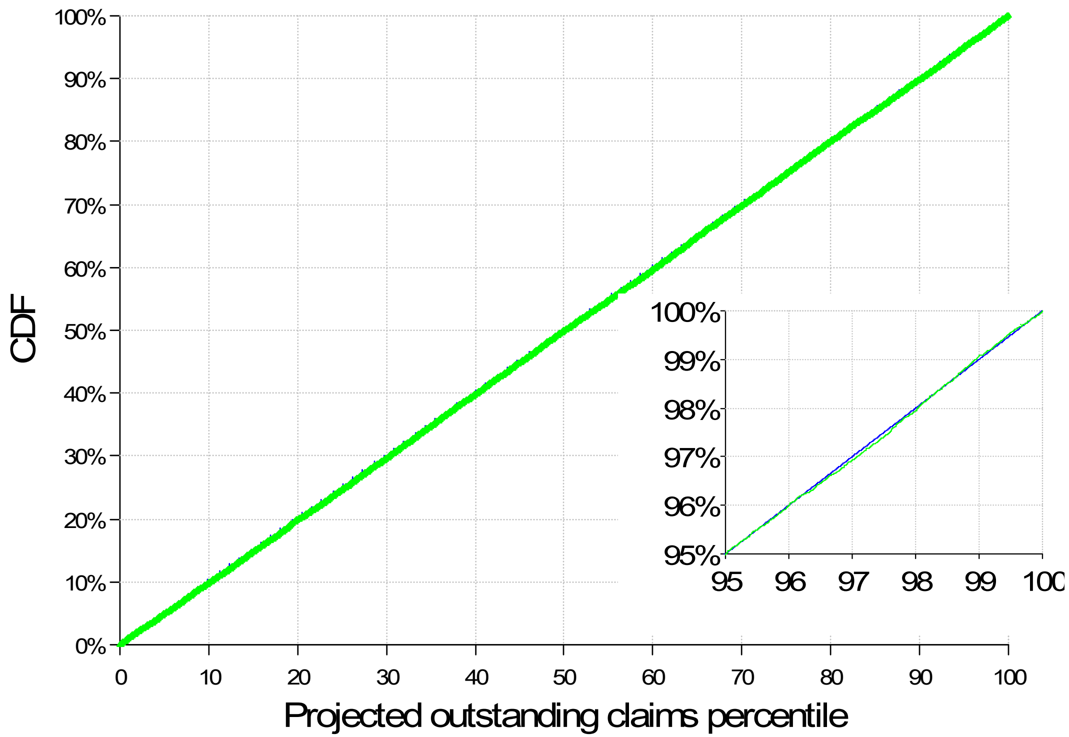


1.16% chance of exceeding projected 99th percentile

Is the test procedure valid?

A one parameter Poisson problem

- Results using:
  - True Poisson parameter μ=1000
  - Generate 20 "data" from a Poisson(μ)
  - Calculate the mean m of the "data"
  - Use a parametric bootstrap (10,000 iterations) to generate 20 "pseudo data" from a Poisson(m) distribution, and calculate the mean m^* for each iteration
  - For each bootstrap iteration, generate a forecast from a Poisson(m^*)
  - Compare a single draw from the true Poisson(μ) distribution against the forecast Poisson(m^*) distribution
  - Repeat 30,000 times



0.93% chance of exceeding projected 99th percentile

Is the test procedure valid?
A two parameter problem

- A two parameter example.
  - Suppose observations follow a Normal distribution with known mean μ and standard deviation σ.
  - Simulate 30,000 samples of *n* observations from a Normal(μ, σ), together with a 'true' forecast *t*, also from a Normal(μ, σ)
  - For each sample of *n* observations, estimate the mean *m* and standard deviation *s*, and perform a parametric bootstrap:
    - Create 10,000 bootstrap samples of *n* observations from a Normal(*m*,*s*), and for each bootstrap iteration, estimate the mean m^* *and standard deviation* s^*, giving a joint distribution of (m^*,s^*)
  - For each bootstrap iteration, simulate a forecast value *f* from a Normal(m^*,s^*), giving a distribution of the forecast *f*. Find the percentile (*p*) of the 'true' value *t* from the distribution of *f*.
  - This give 30,000 values of *p*, and according to the ROC/GIRO WP methodology, *p* should follow a uniform distribution, and 1% of values of *p* should be greater than 0.99

Is the test procedure valid?

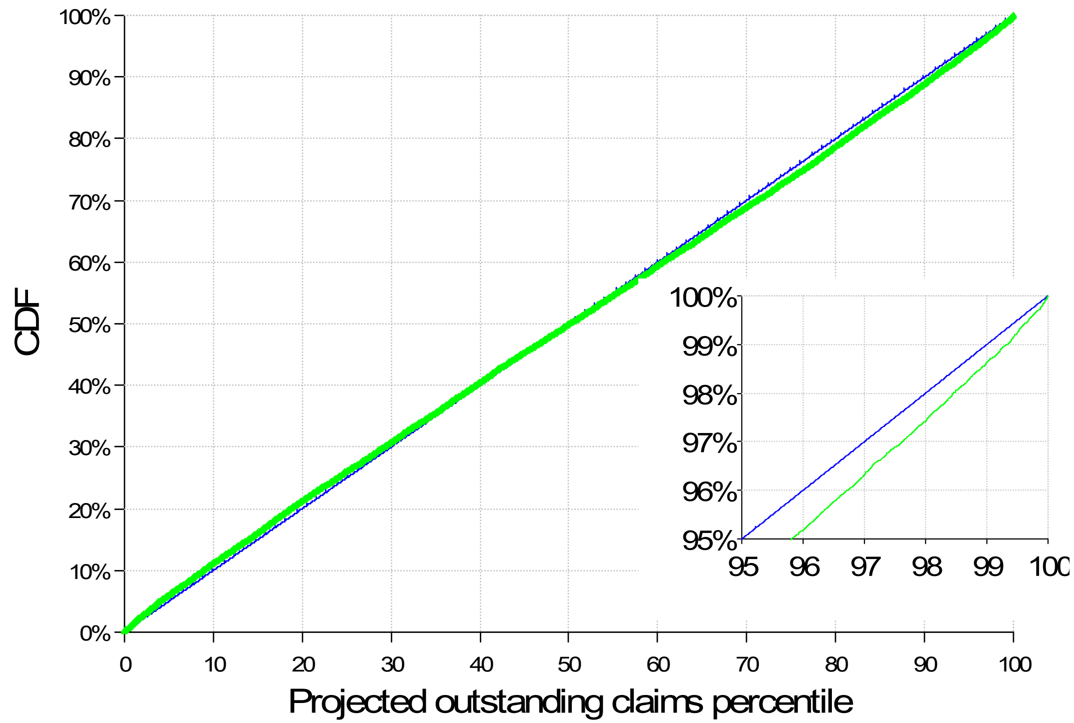
Variations of the two parameter problem

- The bootstrapping and forecasting stages of the bootstrapping procedure were also varied, as follows.
  - For each sample of *n* observations, estimate the mean *m* and standard deviation *s*, and perform a parametric bootstrap:
    – Create 10,000 bootstrap samples of *n* observations from a Normal(*m,s*), and for each bootstrap iteration, estimate the mean *m* (but not s*)*
    – For each bootstrap iteration, simulate a forecast value *f* from a Normal(*m*,s*), giving a distribution of the forecast *f.* Find the percentile (*p*) of the 'true' value *t* from the distribution of *f.*
    – This is analogous to the usual GLM assumption that the 'scale' parameter is a nuisance parameter. When applying bootstrapping in the context of stochastic reserving, England & Verrall also consider the scale parameters as nuisance parameters, and do not re-estimate them in the bootstrapping procedure. This is to give results from the bootstrap procedure that are analogous to results obtained analytically.
  - We also considered the unrealistic assumption that σ is always known:
    – Create 10,000 bootstrap samples of *n* observations from a Normal(m,σ), and for each bootstrap iteration, estimate the mean *m* (but not s*)*
    – For each bootstrap iteration, simulate a forecast value *f* from a Normal(m^*,σ), giving a distribution of the forecast *f.* Find the percentile (*p*) of the 'true' value *t* from the distribution of *f.*

Is the test procedure valid?

A two parameter Normal problem

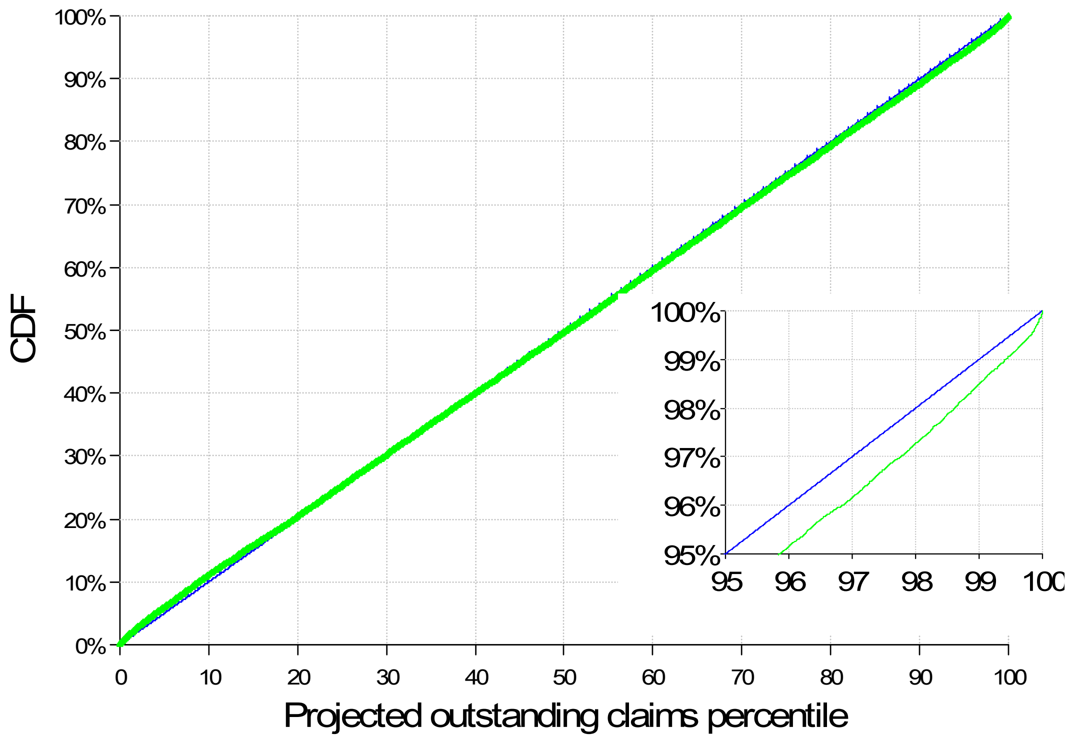
- At the forecasting stage:
  - Using the mean m^* and standard deviation s^* calculated from the bootstrap data sets
  - That is, parameter uncertainty is considered on both the mean and standard deviation



1.37% chance of exceeding projected 99th percentile

Is the test procedure valid?
A two parameter Normal problem

- At the forecasting stage:
  - Using the mean *m** calculated from the bootstrap data sets and standard deviation *s* calculated from the data sample
  - That is, parameter uncertainty is considered on the mean only, and a ‘plug-in’ estimate of the standard deviation is used
  - This is analogous to the usual situation in stochastic reserving where parameter uncertainty on the ODP scale parameters/Mack’s alphas is not considered

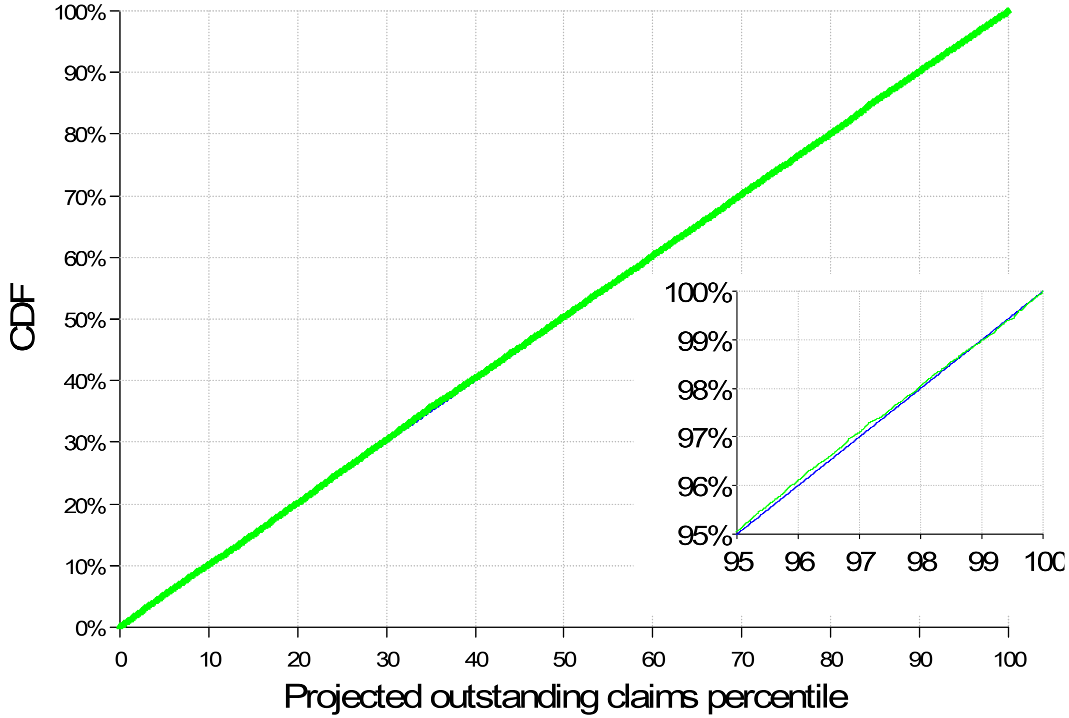


1.50% chance of exceeding projected 99th percentile

Is the test procedure valid?

A two parameter Normal problem

- At the forecasting stage:
  - Using the mean m^* calculated from the bootstrap data sets and the true standard deviation σ
  - That is, parameter uncertainty is considered on the mean only, and the standard deviation is considered known
  - Obviously this is not realistic, but it gives the best results for the GIRO/ROC WP test.



1.01% chance of exceeding projected 99th percentile

Is the test procedure valid?

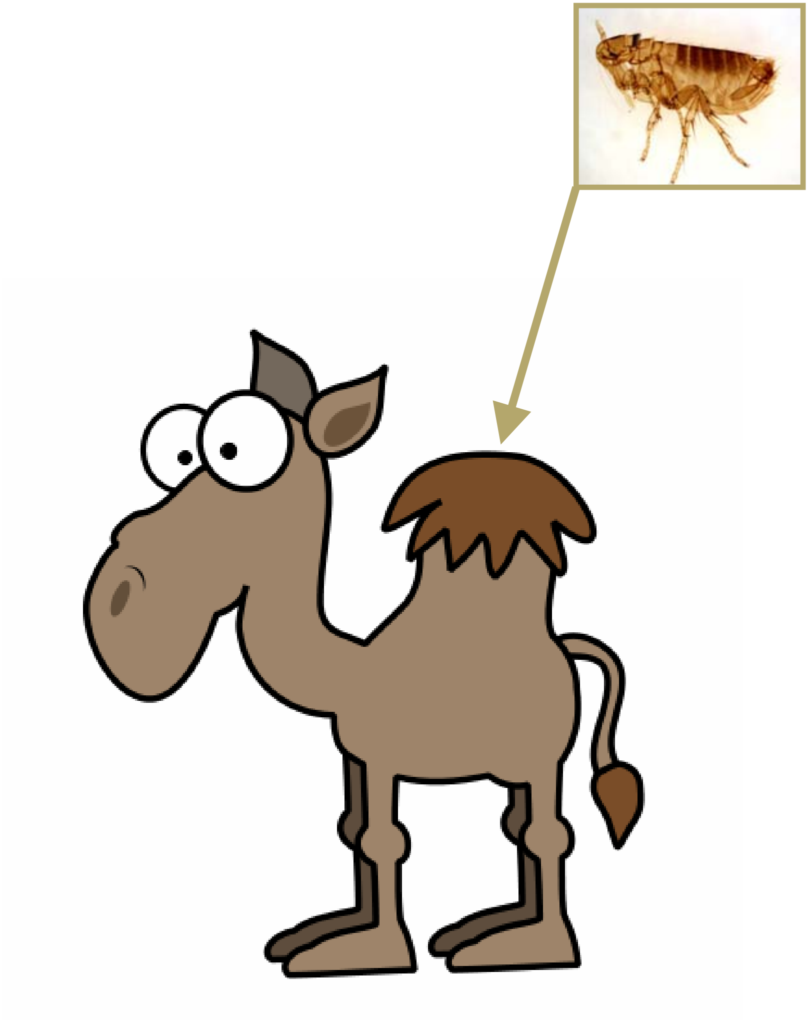
- Although the test seems intuitively reasonable, it appears to show counter-intuitive behaviour with even the simplest examples
  - For the Poisson case, the results depend on the magnitude of the mean
  - For the Normal case, the test only seems to work where the true standard deviation is considered known.
- These simple cases seem to suggest that the standard procedure of including parameter/process uncertainty when forecasting is inconsistent with this test.
- *"It is beyond the scope of this presentation to definitively explain where the test goes wrong."*
- These simple cases seem to indicate that we may *expect* slightly more than 1% of 'true' values to lie above the 99th percentile of the forecast distribution using the data generation and testing procedures.
- We would recommend further research on the validity of the test itself, before placing over-confidence in the results of the working party (in this area).

Conclusions and Recommendations

Conclusions and Recommendations

- When bootstrapping the ODP model, we recommend using non-constant scale parameters, and investigating parametric bootstrapping.
  - With the Taylor & Ashe data, 1.06% of 'true' values were above the 99th percentile.
- For Mack's model, we recommend using a bootstrapping approach to obtain predictive distributions of cash-flows and hence reserves.
  - With the Taylor & Ashe data, 1.5% of 'true' values were above the 99th percentile.
- We do not believe that "Algorithm A" as used by the ROC/GIRO WP is appropriate
  - This invalidates ALL results in the 2007/2008 reports based on Algorithm A
- Our results are repeated, even when Bayesian MCMC methods are used in place of bootstrapping.
- Even with simple cases, the test procedure used by the ROC/GIRO WP seems to be inconsistent with the standard procedure of including parameter/process uncertainty when forecasting (except for special cases).
  - It is possible that we should *expect* more than 1% of true values to lie above the 99th percentile under the test procedure

A Final Word…



- We have performed trillions of calculations to arrive at these results
- Although the results are interesting, focussing on whether the upper tails are slightly underestimated when using bootstrapping is a bit like focusing on a flea and ignoring the camel it is sitting on.
- There are bigger issues!
  - Is the expected value correct in the first place?
  - What about model error?

References

Mack, T (1993). *Distribution-free calculation of the standard error of chain ladder reserve estimates*. ASTIN Bulletin, 23, pp214-225.

Mack, T (1999). *The standard error of chain ladder reserve estimates: recursive calculation and inclusion of a tail factor*. ASTIN Bulletin, 29, 2, pp361-366.

England, P and Verrall, R (1999). *Analytic and bootstrap estimates of prediction errors in claims reserving*, Insurance: Mathematics and Economics 25, pp281-293.

England, P (2002). *Addendum to "Analytic and bootstrap estimates of prediction errors in claims reserving"*, Insurance: Mathematics and Economics 31, pp461-466.

England, PD & Verrall, RJ (2002). *Stochastic Claims Reserving in General Insurance*, British Actuarial Journal 8, III, pp443-544.

England, PD & Verrall, RJ (2006). *Predictive distributions of outstanding claims in general insurance*, Annals of Actuarial Science 1, II, pp221-270.

ROC/GIRO Working Party (2007). *Best estimates and reserve uncertainty*. GIRO Convention paper.

ROC/GIRO Working Party (2008). *Reserving uncertainty*. GIRO Convention paper.

Björkwall, S, Hössjer, O & Ohlsson, E (2008). *Non-parametric and parametric bootstrap techniques for arbitrary age-to-age development factor methods in stochastic claims reserving*. Research Report 2008:2, Department of Mathematical Statistics, Stockholm University.

Brooks, D *et al* (2009). *Actuarial aspects of internal models for Solvency II*. Presented to the Institute of Actuaries 23 Feb 2009

Murphy, D (2009). *Where's the Beef? Does the Mack Method produce an undernourished range of possible outcomes*. Presented at the Casualty Loss Reserve Seminar, Chicago.

Implementation details

- The bootstrap calculations were implemented using EMB Igloo Enterprise, which allowed the computations to be run in parallel across 20 computers (using distributed computing).
  - Generating 30,000 samples, and performing 10,000 bootstrap iterations on each, took about 15 minutes in total.
- The Bayesian MCMC computations were performed using a specially written C++ dll, used in association with Igloo Enterprise. The run times varied depending on the MCMC algorithm and settings used.
  - We implemented both Gibbs sampling with Adaptive Rejection Sampling (ARS), and single component adaptive Metropolis-Hastings with eigen-vector rotation.

Acknowledgements

We are very grateful for the assistance of the following people:

- Chris Seah, James Wakefield and Stuart Tomlinson for setting up the EMB Igloo Enterprise network of computers
- Andy Race and David Thomas for EMB Igloo Enterprise software technical support
- Mark Sinclair-McGarvie for developing the MCMC C++ dll

Contact us:

To obtain more detailed results, or to discuss this presentation further, please contact:

peter.england@emb.com

martin.cairns@emb.com

