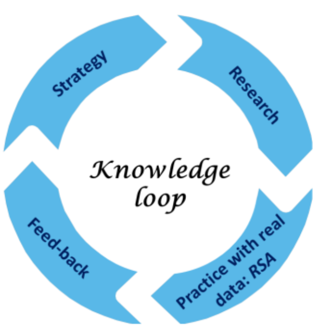
Institute and Faculty of Actuaries

Empowering Chain Ladder: A Cass Business School Production

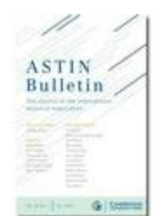
Jens Perch Nielsen
Cass Business School
Professor of Actuarial Science

23 September 2014

Background



2010 Including Count Data in Claims Reserving
2011 Cash flow simulation for a model of outstanding liabilities based on claim amounts and claim numbers
2012 Double Chain Ladder

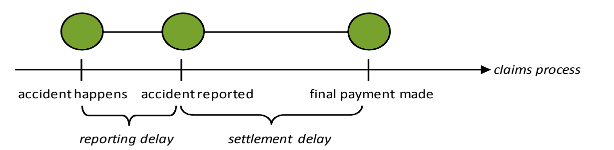


2012 Statistical modelling and forecasting in Non-life insurance
2013 Double Chain Ladder and Bornhuetter-Ferguson
2013 Double Chain Ladder, Claims Development Inflation and Zero Claims
2014 RBNS preserving Double Chain Ladder (submitted)

Our aim: a package implementing recent research developments.

The problem: the claims reserving exercise

The life of an individual claim in the general claims process:



Three categories of claim:

- Incurred but not reported, **IBNR**
- Reported but not settled, **RBNS**
- Reported and paid

The problem: the claims reserving exercise

The objectives:

- How large **future claims payments** are likely to be.
- The **timing** of future claim payments.
- The **distribution** of possible outcomes: future **cash-flows**.

Framework: Double Chain Ladder

What is Double Chain Ladder?

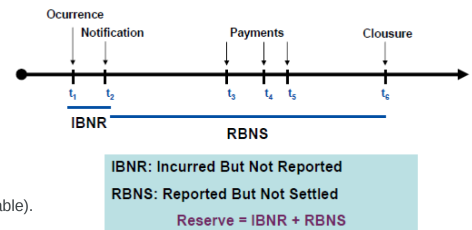
A firm statistical model which breaks down the chain ladder estimates into individual component

Why?

- Connection with classical reserving (tacit knowledge)
- RBNS and IBNR claims
- The distribution: full cash-flow

What is required?

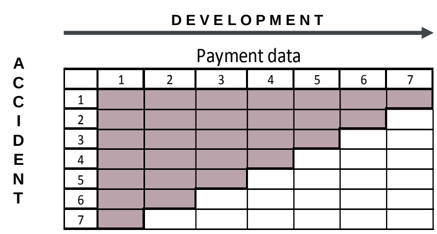
It works on **run-off triangles** (adding expert knowledge if available).



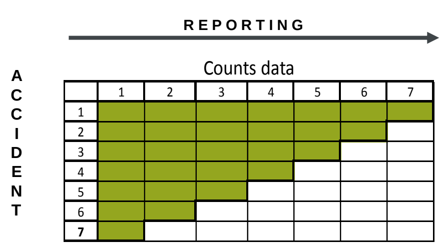
The modelled data: two run-off triangles

We model annual/quarterly run-off triangles:

- Incremental aggregated payments **(Paid triangle)**



- Incremental aggregated counts data, which is assumed to be fully run off. **(Counts triangle)**



The Double Chain Ladder Model

Parameters involved in the model:

Ultimate claim numbers: α_i

Reporting delay: $\beta_{j'}$

Settlement delay: π_l

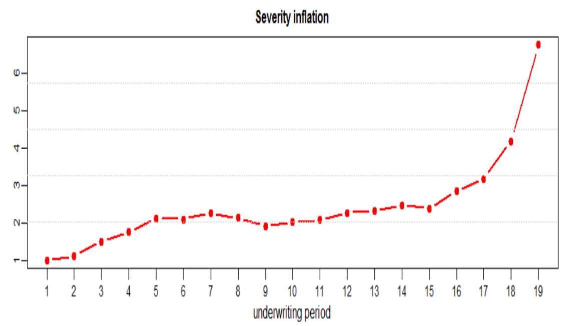
Development delay: $\tilde{\beta}_j$

Ultimate payment numbers: $\tilde{\alpha}_i$

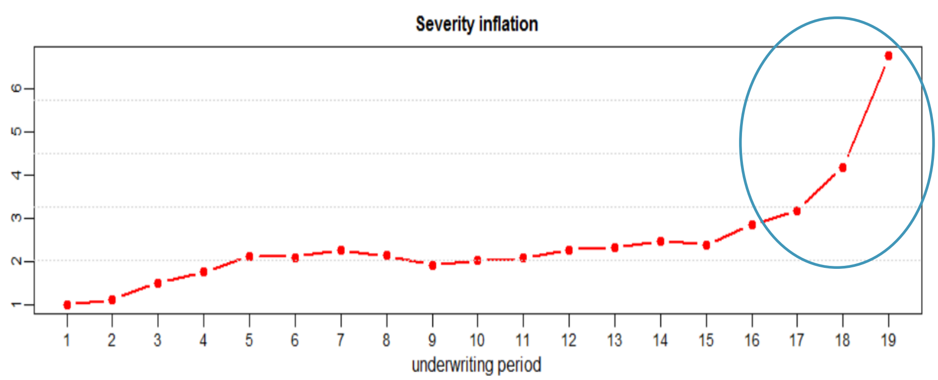
Severity:

underwriting inflation: γ_i

delay mean dependencies: μ



The Double Chain Ladder Model



The Double Chain Ladder Model

The payments triangle

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19
1	51645	513057	747581	554656	426090	211996	212916	16199	9091	36933	0	0	0	0	0	0	0	0	0
2	143607	1006115	910371	735878	593070	765997	614642	245482	116065	14633	36298	164636	14601	66826	6415	11400	0	0	NA
3	345758	1467254	1291694	1236995	1127060	779156	391920	844780	94346	229871	12232	11210	20826	84329	9938	16898	0	NA	NA
4	408108	1875253	1809624	1859877	1806412	1422161	761701	306739	109769	139582	53448	36557	0	6731	0	0	NA	NA	NA
5	711788	3253701	2695979	2592550	3376797	2100946	923045	434936	124256	29942	23026	324	58834	31180	12306	NA	NA	NA	NA
6	941448	3614819	3273886	4479163	3841136	2032530	1241700	471996	120135	59047	5081	295	9393	0	NA	NA	NA	NA	NA
7	1221479	5814000	5904668	7112406	5320976	2425835	856998	196958	133568	40099	11797	65669	98728	NA	NA	NA	NA	NA	NA
8	1684782	8163947	7609088	7722323	6298256	1981161	830186	580355	197501	124446	63687	28557	NA	NA	NA	NA	NA	NA	NA
9	2253183	9479779	7696767	8260492	5871622	2339555	1099429	363351	147355	43520	13782	NA	NA	NA	NA	NA	NA	NA	NA
10	2042830	8791743	9169217	7864324	5894987	1977707	722425	245391	59786	-1390	NA	NA	NA	NA	NA	NA	NA	NA	NA
11	1570388	9961564	9669606	8024282	6120733	2391815	617560	97794	70961	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
12	1455847	9182448	8261734	8373519	4994670	1885764	882915	241387	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
13	1128853	7675536	8515497	6467241	4505204	1502376	460521	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
14	1380818	11547624	8890421	7964029	4951038	1980364	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
15	2195835	12381318	10390839	7516444	4968713	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
16	2068049	14178820	11164349	7740463	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
17	1747083	11599608	8808101	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
18	3294583	15210026	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
19	4664157	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

23 September 2014 9

The Double Chain Ladder Model

The payments triangle

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19
1	51645	513057	747581	554656	426090	211996	212916	16199	9091	36933	0	0	0	0	0	0	0	0	0
2	143607	1006115	910371	735878	593070	765997	614642	245482	116065	14633	36298	164636	14601	66826	6415	11400	0	0	NA
3	345758	1467254	1291694	1236995	1127060	779156	391920	844780	94346	229871	12232	11210	20826	84329	9938	16898	0	NA	NA
4	408108	1875253	1809624	1859877	1806412	1422161	761701	306739	109769	139582	53448	36557	0	6731	0	0	NA	NA	NA
5	711788	3253701	2695979	2592550	3376797	2100946	923045	434936	124256	29942	23026	324	58834	31180	12306	NA	NA	NA	NA
6	941448	3614819	3273886	4479163	3841136	2032530	1241700	471996	120135	59047	5081	295	9393	0	NA	NA	NA	NA	NA
7	1221479	5814000	5904668	7112406	5320976	2425835	856998	196958	133568	40099	11797	65669	98728	NA	NA	NA	NA	NA	NA
8	1684782	8163947	7609088	7722323	6298256	1981161	830186	580355	197501	124446	63687	28557	NA	NA	NA	NA	NA	NA	NA
9	2253183	9479779	7696767	8260492	5871622	2339555	1099429	363351	147355	43520	13782	NA	NA	NA	NA	NA	NA	NA	NA
10	2042830	8791743	9169217	7864324	5894987	1977707	722425	245391	59786	-1390	NA	NA	NA	NA	NA	NA	NA	NA	NA
11	1570388	9961564	9669606	8024282	6120733	2391815	617560	97794	70961	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
12	1455847	9182448	8261734	8373519	4994670	1885764	882915	241387	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
13	1128853	7675536	8515497	6467241	4505204	1502376	460521	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
14	1380818	11547624	8890421	7964029	4951038	1980364	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
15	2195835	12381318	10390839	7516444	4968713	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
16	2068049	14178820	11164349	7740463	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
17	1747083	11599608	8808101	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
18	3294583	15210026	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
19	4664157	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

23 September 2014 10

The Double Chain Ladder Model

The reserve per underwriting year

	reserve	proportion of toal reserve
1	0.000000e+00	0.00
2	8.304134e+02	0.00
3	1.073025e+02	0.00
4	8.348906e+02	0.00
5	4.007342e+03	0.00
6	3.141223e+04	0.00
7	1.417988e+05	0.00
8	2.498179e+05	0.00
9	3.595187e+05	0.00
10	3.824873e+05	0.00
11	5.252174e+05	0.00
12	6.315314e+05	0.00
13	9.770538e+05	0.01
14	2.549259e+06	0.01
15	5.449377e+06	0.03
16	1.543851e+07	0.08
17	2.174178e+07	0.11
18	4.445951e+07	0.23
19	9.897470e+07	0.52

The Double Chain Ladder Model

The reserve per underwriting year

	reserve	proportion of toal reserve
1	0.000000e+00	0.00
2	8.304134e+02	0.00
3	1.073025e+02	0.00
4	8.348906e+02	0.00
5	4.007342e+03	0.00
6	3.141223e+04	0.00
7	1.417988e+05	0.00
8	2.498179e+05	0.00
9	3.595187e+05	0.00
10	3.824873e+05	0.00
11	5.252174e+05	0.00
12	6.315314e+05	0.00
13	9.770538e+05	0.01
14	2.549259e+06	0.01
15	5.449377e+06	0.03
16	1.543851e+07	0.08
17	2.174178e+07	0.11
18	4.445951e+07	0.23
19	9.897470e+07	0.52

= 86% of total reserve

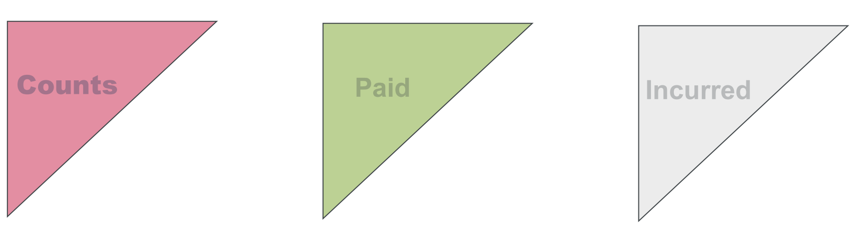
The Double Chain Ladder Model

Summary of the major drawback of classical Chain Ladder (and thus the basic Double Chain Ladder method):

- The lack of sufficient data in the most recent underwriting years yields to a severity inflation estimation being too instable and thus not trustable in those most recent years.
- Even worse, those most recent underwriting years account for the very major part of the reserve.

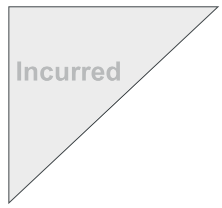
RBNS Preserving Double Chain Ladder

Solution: Incorperate **Case estimates (expert knowledge):**



RBNS Preserving Double Chain Ladder

The incurred triangle:



- It is not data, but a mixture of data and expert knowledge
- It contains payments and case estimates of RBNS claimes

RBNS Preserving Double Chain Ladder

- From the incurred triangle, one can extract the RBNS part estimated by the case department.
- The RBNS case estimates differ from the DCL RBNS estimates

RBNS Preserving Double Chain Ladder

The values of the severity inflation estimates in the most recent calendar years result in a big difference between DCL and case estimates based RBNS numbers

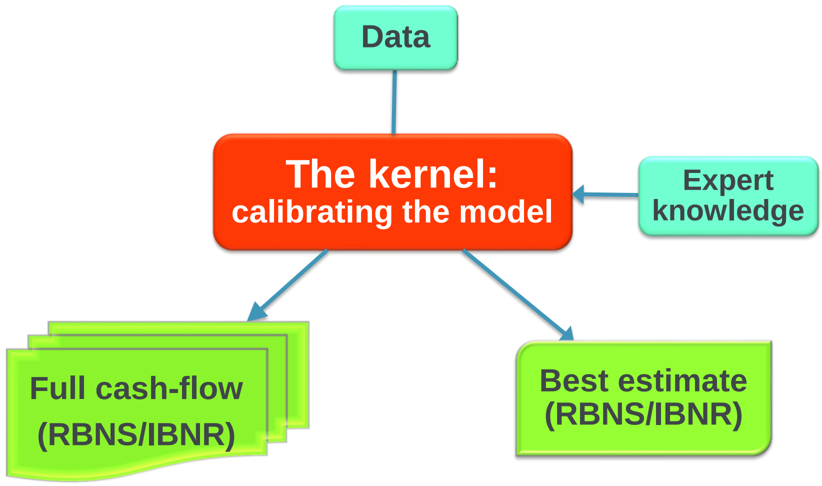
	RBNS via DCL	RBNS via case estimates	diff/ultimate
1	0	0	0.0000000000
2	830	0	0.0001524044
3	107	4011	0.0004901895
4	835	-9524	0.0009776366
5	4007	36500	0.0019828617
6	29477	5000	0.0012162334
7	138978	1381	0.0046895341
8	244550	92278	0.0042858333
9	352419	57627	0.0077738288
10	369966	190335	0.0048338207
11	506266	241142	0.0067846411
12	602066	1444	0.0167014947
13	929374	1210062	0.0089660533
14	2453703	2719667	0.0067760876
15	5301958	6123466	0.0190207477
16	15190206	9249185	0.1206529140
17	21248200	13099480	0.1888002860
18	42539709	24828096	0.2802813405
19	74094249	31454377	0.4114369497

RBNS Preserving Double Chain Ladder

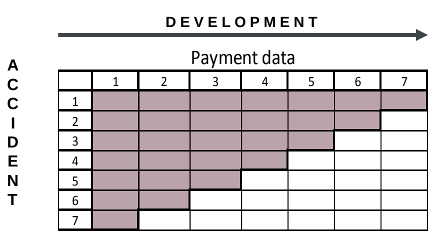
What does RBNS preserving Double Chain Ladder (PDCL) do?

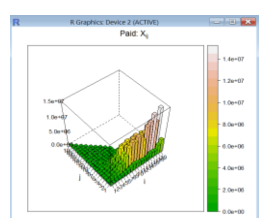
- PDCL preserves the RBNS case estimates.
- Hereby, the RBNS reserve part is not just replaced by the case estimates.
- The DCL parameters estimates are also adjusted accordingly.
- Therefore, PDCL estimates the exact RBNS case estimates but also corrects the IBNR estimates.

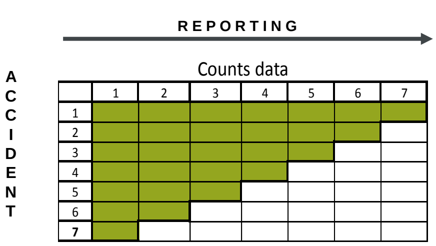
The Double Chain Ladder package

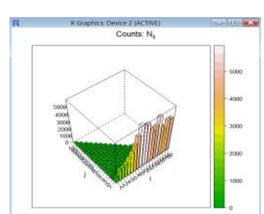


Visualizing the data: the histogram









The kernel: parameter estimation using DCL

- **dcl.estimation(), bdcl.estimation(),**

  idcl.estimation(), pdcl.prediction()

dcl.estimation {DCL} R Documentation

Parameter estimation - Double Chain Ladder model

Description

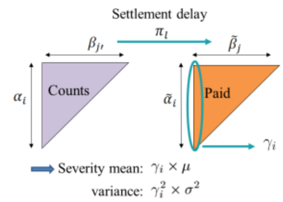
Compute the estimated parameters in the model (delay parameters, severity underwriting inflation, severity mean and variance) using the Double Chain Ladder method.

Usage

```
dcl.estimation( Xtriangle , Ntriangle , adj = 1 , Tables = TRUE , num.dec = 4 )
```

Arguments

`Xtriangle`	The paid run-off triangle: incremental aggregated payments. It should be a matrix with incremental aggregated payments located in the upper triangle and the lower triangle consisting in missing or zero values.
`Ntriangle`	The counts data triangle: incremental number of reported claims. It should be a matrix with the observed counts located in the upper triangle and the lower triangle consisting in missing or zero values. It should has the same dimension as `Xtriangle` (both in the same aggregation level (quarters, years,etc.))
`adj`	Method to adjust the estimated delay parameters for the distributional model. It should be 1 (default value) or 2. See more in details below.
`Tables`	Logical. If `TRUE` (default) it is showed a table with the estimated parameters.
`num.dec`	Number of decimal places used to report numbers in the tables (if `Tables=TRUE`).



The kernel: parameter estimation using DCL

- The function **Plot.dcl.par()** to visualize the break down of the classical chain ladder parameters

Plot.dcl.par {DCL} R Documentation

Plotting the estimated parameters in the DCL model

Description

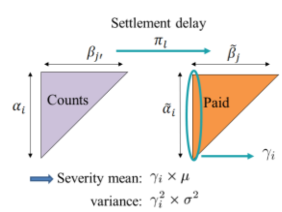
Show a two by two plot with the estimated parameters in the Double Chain Ladder model

Usage

```
Plot.dcl.par( dcl.par , type.inflat = 'DCL' )
```

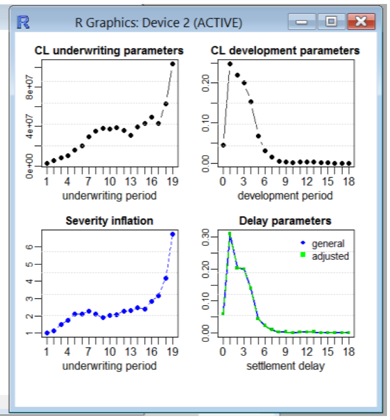
Arguments

`dcl.par`	A list object with the estimated parameters: the value returned by the functions `dcl.estimation`, `bdcl.estimation` or `idcl.estimation`.
`type.inflat`	Method used to estimate the inflation . Possible values are: 'DCL' (default) if it was used `dcl.estimation`, 'BDCL' if `bdcl.estimation`, and 'IDCL' if `idcl.estimation`



The functions in action: an example

```
> my.dcl.par<-dcl.estimation(XtriangleBDCL,NtriangleBDCL)
   delay.par delay.prob inflation severity.mean severity.var
1     0.0592     0.0592    1.0000      2579.064    286808926
2     0.3098     0.3098    1.1173      2881.570    358036053
3     0.2032     0.2032    1.4947      3855.014    640796811
4     0.1996     0.1996    1.7461      4503.280    874432486
5     0.1388     0.1388    2.1075      5435.263   1273824141
6     0.0440     0.0440    2.0936      5399.464   1257099346
7     0.0227     0.0227    2.2495      5801.697   1451371123
8     0.0095     0.0095    2.1250      5480.521   1295126156
9     0.0018     0.0018    1.9028      4907.442   1038433681
10    0.0029     0.0029    2.0197      5208.871   1169918179
11    0.0002     0.0002    2.0704      5339.587   1229373075
12    0.0026     0.0026    2.2666      5845.709   1473474978
13    0.0019     0.0019    2.3157      5972.242   1537953134
14    0.0032     0.0032    2.4747      6382.359   1756429648
15   -0.0002     0.0006    2.3829      6145.592   1628530112
16    0.0013     0.0000    2.8391      7322.296   2311867264
17   -0.0004     0.0000    3.1815      8205.383   2903127034
18    0.0000     0.0000    4.1747     10766.824   4998544792
19    0.0000     0.0000    6.7501     17409.045  13068274219
  mean.factor mean.factor.adj variance.factor
1    2579.002        2579.064       286808926
> Plot.dcl.par(my.dcl.par)
> |
```



Parameter estimates in two cases: the basic DCL model (only mean specifications) and the distributional model.

The best estimate: RBNS/IBNR split using DCL

- The function **dcl.predict()**

dcl.predict {DCL} R Documentation

Pointwise predictions (RBNS/IBNR split)

Description

Pointwise predictions by calendar years and rows of the outstanding liabilities. The predictions are splitted between RBNS and IBNR claims.

Usage

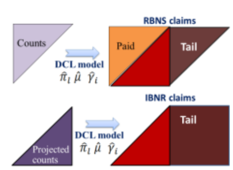
```
dcl.predict( dcl.par , Ntriangle , Model = 2 , Tail = TRUE , Tables = TRUE , summ.by="diag", num.dec = 2 )
```

Arguments

`dcl.par`	A list object with the estimated parameters: the value returned by the functions `dcl.estimation`, `bdcl.estimation` or `idcl.estimation`.
`Ntriangle`	Optional. The counts data triangle: incremental number of reported claims. It should be a matrix with the observed counts located in the upper triangle and the lower triangle consisting in missing or zero values. It should has the same dimension as the `Xtriangle` (both in the same aggregation level (quarters, years,etc.)) used to derive `dcl.par`
`Model`	Possible values are 0, 1 or 2 (default). See more details below.
`Tail`	Logical. If `TRUE` (default) the tail is provided.
`Tables`	Logical. If `TRUE` (default) it is shown a table with the predicted outstanding liabilities in the future calendar periods (`summ.by="diag"`) or by underwriting period (`summ.by="row"`).
`summ.by`	A character value such as `"diag"`, `"row"` or `"cell"`.
`num.dec`	Number of decimal places used to report numbers in the tables. Used only if `Tables=TRUE`

Details

If `Model=0` or `Model=1` then the predictions are calculated using the DCL model parameters in assumptions M1-M3 (general delay parameters, see Martinez-Miranda, Nielsen and Verrall 2012). If `Model=2` the adjusted delay probabilities (distributional model D1-D4) are considered. By



The full cash-flow: Bootstrapping using DCL

- The function **dcl.boot()**

dcl.boot {DCL} R Documentation

Bootstrap distribution: the full cashflow

Description

Provide the distribution of the IBNR, RBNS and total (RBNS+IBRN) reserves by calendar years and rows using bootstrapping.

Usage

```
dcl.boot( dcl.par , sigma2 , Ntriangle , boot.type = 2 , B = 999 , Tail = TRUE , summ.by = "diag" , Tables = TRUE , num.dec = 2 )
```

Arguments

`dcl.par`	A list object with the estimated parameters: the value returned by the functions `dcl.estimation`, `bdcl.estimation` or `idcl.estimation`.
`sigma2`	Optional. The variance of the individual payments in the first underwriting period.
`Ntriangle`	The counts data triangle: incremental number of reported claims. It should be a matrix with the observed counts located in the upper triangle and the lower triangle consisting in missing or zero values. It should be the same triangle used to get the value passed by the argument `dcl.par`.
`boot.type`	Choose between values 1, to provide only the variance process, or 2 (default), to take into account the uncertainty of the parameters.
`B`	The number of simulations in the bootstrap algorithm. The defaul value is 999.
`Tail`	Logical. If `TRUE` (default) the tail is provided.
`summ.by`	A character value such as `"diag"`, `"row"` or `"cell"`.
`Tables`	Logical. If `TRUE` (default) it is showed a table with the summary (mean, standard deviation, 1%, 5%, 50%, 95%, 99%) of the distribution of the outstanding liabilities in the future calendar periods (if `summ.by="diag"`) or by underwriting period (if `summ.by="row"`).
`num.dec`	Number of decimal places used to report numbers in the tables. Used only if `Tables=TRUE`

Details

- The function **Plot.cashflow()**

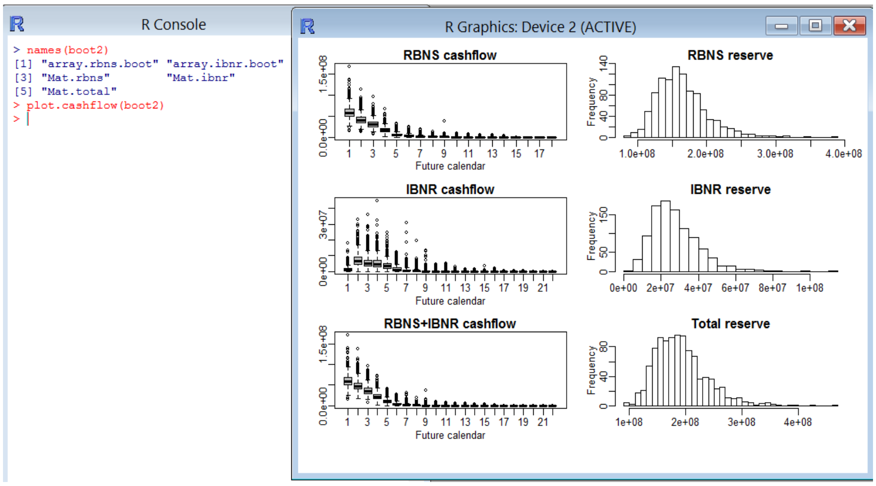
The functions in action: an example

R Console

```
> boot2<-dcl.boot(my.dcl.par,Ntriangle=NtriangleBDCL)
[1] "Please wait, simulating the distribution..."
[1] "Done!"
   period        rbns   mean.rbns     sd.rbns     Q1.rbns     Q5.rbns
1       1 59845052.96 60104639.86 15073076.18 33597995.93 40915229.26
2       2 41447058.01 41679263.34 11015109.71 22154274.56 27730524.16
3       3 31016097.53 31146079.93  9826146.21 14928552.44 18278582.96
4       4 17542089.42 17251007.40  6956163.95  4958809.18  8595612.19
5       5  6443018.76  6403003.15  3843801.59  1016332.33  2071798.34
6       6  3192176.74  3510417.33  2623285.96   177136.11   762954.18
7       7  1445598.60  1597909.78  1873972.33     2517.46    84721.67
8       8   675017.48   852208.53  1174759.22        0.00      412.15
9       9   642274.45   536551.31  1424260.00        0.00        0.00
10     10   423522.65   376713.73   827293.75        0.00        0.00
11     11   535548.94   164627.71   503755.34        0.00        0.00
12     12   404459.01    96801.19   355413.76        0.00        0.00
13     13   334964.95    56962.35   324013.16        0.00        0.00
14     14    60022.99    13651.87   137771.89        0.00        0.00
15     15        0.00       95.42     2144.22        0.00        0.00
16     16        0.00        0.00        0.00        0.00        0.00
17     17        0.00        0.00        0.00        0.00        0.00
18     18        0.00        0.00        0.00        0.00        0.00
19     19        0.00        0.00        0.00        0.00        0.00
20     20
21     21
22     22
23     23
24     24
25     25
26     26
27     27
28     28
```

- ➢ A table showing a summary of the distribution: mean, std. deviation, quantiles.
- ➢ Arrays and matrices with the full simulated distributions

The functions in action: an example



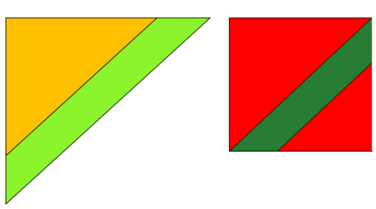
23 September 2014 27

Validation

- The function **validating.incurred()**

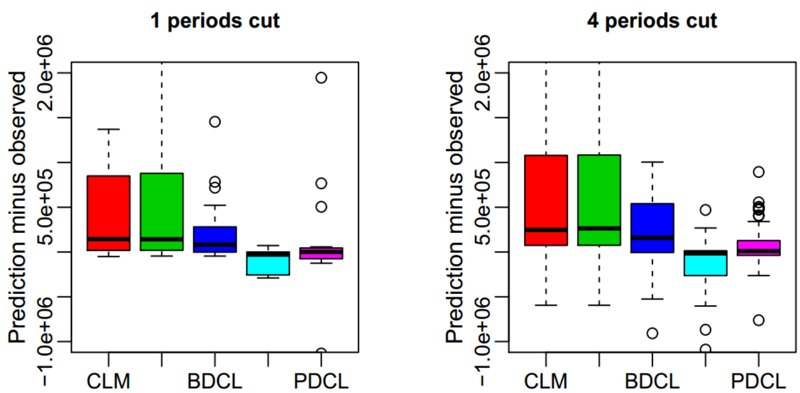
Testing results against experience:

1. Cut c=1,2,…. diagonals (periods) from the observed triangle.
2. Apply the estimation methods.
3. Compare forecasts and actual values.

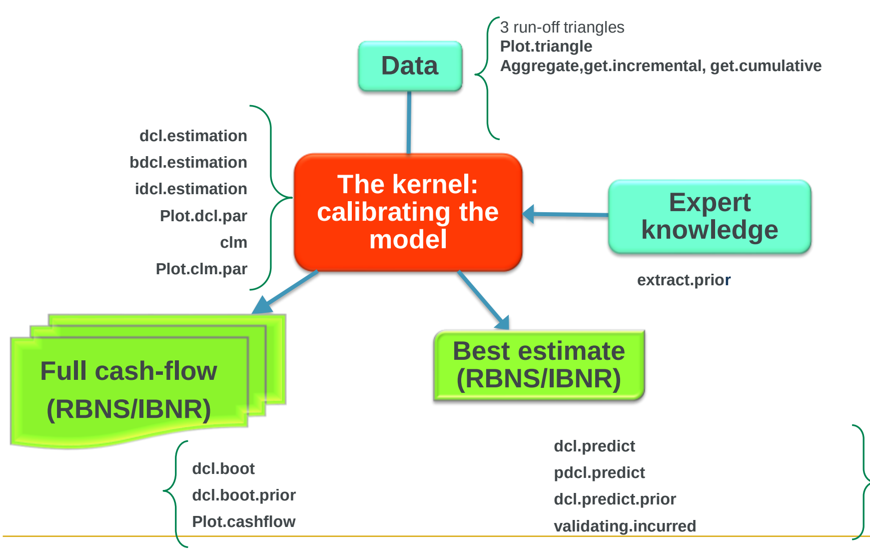


23 September 2014 28

Validation



Summary: the content of the package



Questions

Comments

Expressions of individual views by members of the Institute and Faculty of Actuaries and its staff are encouraged.

The views expressed in this presentation are those of the presenter.