Reserve applications

2020-03-03, Stig Rosenlund

An all-in-one Rapp package is available at http://www.stigrosenlund.se/rappzip.htm as a zip file. It contains the essentials and example Rapp programs. Suppose you have started Rappmenus.Exe:
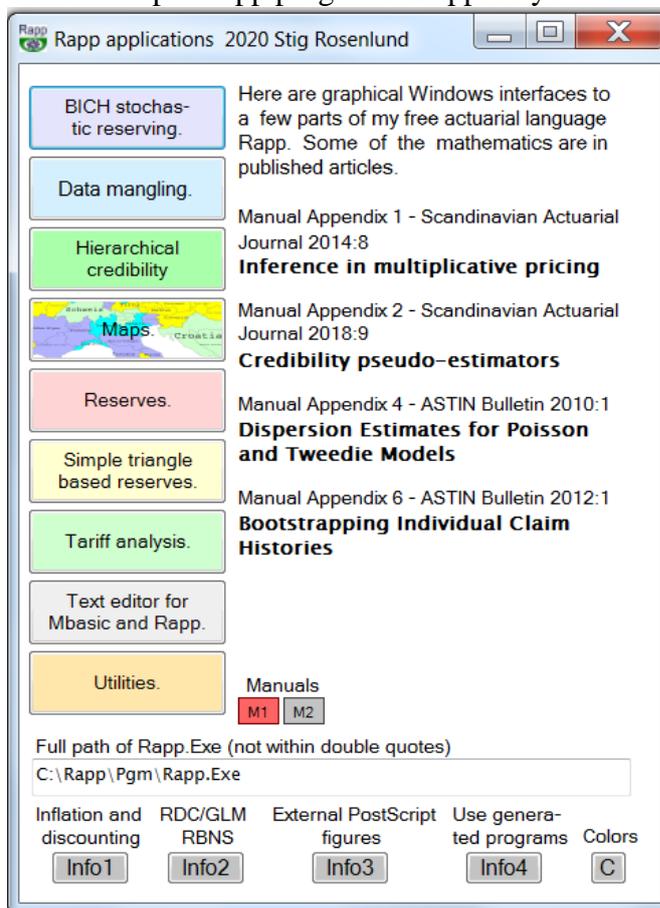
Input data for menu Reserves

Data requirements
The applications presuppose that there is a table of individual payments and their payment dates (Paid) and/or a table of changes in known claim cost (= Paid + Claimhandler-reserve) and their dates (Incurred). Programs in e.g. SAS shall then make two text files for Rapp from these tables.

It is also possible to use simple aggregated triangle files, but RDC cannot be used with them.

Paid infile (file for payments)
After Paid infile: in Reserves you give a text file with lines that look like this. The header line is not part of the file.

```
ID              A1 A2 A3 A4  T Claimdt   Reportdt Paydt    Finaldt   Paymentamount
02123455111234THEFT 02 01 L1 110 0 19940103 19940121 19940301 19940901      000001234
```

where

ID = unique claim ID
A1 = Arg1 = argument 1, such as geographic area
A2 = Arg2 = argument 2, such as line of business
A3 = Arg3 = argument 3, e.g. type of damaged object, e.g. L1 for personal property
A4 = Arg4 = argument 4, such as type of damage, e.g. 010 for fire
T = flag 0/1 for top (large) claim as judged at reporting

The last non-blank character in ID must be in the same position for all claims, and no blanks must be present in ID before that position.

The arguments A1-A4 shall occur with no embedded blanks. Each is required to have the same length in all data lines. Numerical arguments shall have left zero padding if necessary for the requirement. They shall be separated by exactly one blank, not more. If you do not have four arguments, let the superfluous ones have dummy values eg all X. Exactly one blank between dates.

There must a line with Paydt = Reportdt for each claim. If there is none naturally, make such a line with Paymentamount = 0. The file shall be sorted on ID.

2015-09-19. New choice in Rappmenus / Reserves / Files and folders to let Rapp make a line with Paydt = Reportdt for each claim if none exists. The files will be sorted on ID also.

Paymentamount need not be filled with zeros to the left and can have any length.

Incurred infile (file for changes in known-claim-cost)
After Incurred infile: in Reserves you give a text file with lines that look like this. The header line is not part of the file.

```
ID                   A1 A2 A3 A4  T  Claimdt  Reportdt Changedt  Finaldt  Changeamount
02123455111234THEFT 02 01 L1 110 0 19940103 19940121 19940301 19940901     00000016å
```

where

Changedt   = date of change in known-claim-cost (payment or reserve change)

There must be a line with Changedt = Reportdt. If there is none naturally, make such a line with Changeamount = 0.

Changeamount need not be filled with zeros to the left and can have any length. Rapp can read numbers with a minus sign, eg -10753, and negative Zoned Decimal numbers generated by Easytrive in the mainframe. For example, 00001075L which is equal to -10753, and 00000016å, which is equal to -160. Positive Zoned Decimal numbers generated by COBOL do not work, however. (Note. These comments on Zoned Decimal is a legacy from my time at the mainframe. Would be almost totally without interest by July 2014. I keep them for sentimental reasons.)

If you have no Incurred file it can be omitted. Likewise, if you have Incurred but not Paid (a case not likely to appear in practice), the Paid infile can be omitted.

For both files the menus admit wild card specifications of argument values with underscores. This is done in the menu accessed by button Adv (for Advanced).

Overview of the applications' function
Extracts of the files selected by values of A1-A4 and T are created. Exactly two of A1-A4, e.g. A2 and A3, shall be kept in the extracts. The menu "Reserves" then performs reserve calculation with one or both of the two kept arguments as grouping variables. Menu "BICH stochastic reserving" uses the same kind of infiles.

Setup instruction
Create folder C:\Rapp, or another suitable folder, and fill it with the files from Rapp.zip at my website. Template Rapp programs, for use in the menus "Reserves" and "BICH stochastic reserving" and in four other menus, are found in folder Rpp. Init.Rpp is also found there. Template (example) parmfiles, for use in "Open parmfile" in the menus, are found in BRparm and other folders ending

with parm. All needed data files for the template reserving parmfiles are found in Data, after running Mangle1.Rpp and Mangle2.Rpp. They are all fakes - no business secrets revealed. Needed folders and files are stated in the "Files and folders" menu in Reserves. Their names are included in the parmfile.

Copy the right one of the Rapp exes on http://www.stigrosenlund.se/rappexes.htm to Rapp.Exe. The folder for Exe's can be on a server. But large data files should be placed on the C-disk for speed.

Running the application Reserves

Start Rappmenus.Exe. You do not install it in the Windows Install sense. A shortcut can be pinned to the Destop and/or Startbar. The submenu Reserves is self-explaining. It looks like this.

The background colors are intended to help locate parameter groups quickly. For example green for Excel parameters and output, turquoise for parameters belonging only to method RDC, and sandy brown for price correction parameters. Yellow buttons for start of Rapp runs, with a red border if partial computations relying on previous runs are to be made. Press third grey button C in first menu for information.

Recomputation after changes: You change increment values in View Xml 1 / first sheet, including tails, and make them affect accumulated values and all other values deduced from increments. Also some other output files, including PDF graphs, are affected. Do not change increment values in the block Total - they will be recomputed automatically. Do not put formulas in increment blocks. Leave Excel while saving changes in the same Xml-file. Press Recompute. Press View Xml 1 to see the results. Do not make another Run or open another parmfile between the Run that produced this Xml 1 and the second View Xml 1.

The GLM and "User factor" buttons and their output at "GLM related" are described in menus of Rappmenus. For some explanation of box Adv, see Appendix 1.

New 2018-04: Choice to use a file with yearly interest rates that change every month. See label "Yearly interest rate % per month" in the menu. Explanations given in the manual Rappmane.doc or Rappmane.pdf and in right-click windows in the menu.

Interpretation of the output

In the parameter file you select a part of the out-file-names. Assume that the part is X.

Output files in C:\Rapp\Txt, C:\Rapp\Pdf, C:\Rapp\Xml

The following folders are given in"Files and folders" :
  C:\Rapp\Txt for text files
  C:\Rapp\Pdf for PDF files
  C:\Rapp\Xml for XML Excel files

Output file	Brief contents description
ResvX-1.Pdf	One graph per region and/or line of business showing Known-claim-cost + IBNS + tail for the claim periods and for rolling 12-month periods.
ResvX-1.Xml	XML-file with several sheets. Main sheet made from ResvX-1x.Txt if it exists, otherwise from ResvX-1.Txt. **The text file is easier to treat further with other programs**.
ResvX-2.Pdf	One graph per region and/or line of business and claim period showing the incremental values.
ResvX-3.Pdf	One graph per region and/or line of business and claim period showing the accumulated values.

ResvX-4.Pdf	As ResvX-1.pdf, but with
	  Paid + IBNS + tail-paid
	next to
	  Known-claim-cost + IBNS + tail
	for the claim periods.

ResvX-5.Pdf	One graph per claim period for the regions and/or lines of business next to each other. Shows Known-claim-cost, Known-claim-cost + IBNS + tail in mono-color bars and Paid, Paid + IBNS + paid-tail in bars with lines.
ResvX-n-inc.Pdf	As ResvX-2.Pdf for the number of claims.
ResvX-n-acc.Pdf	As ResvX-3.Pdf for the number of claims.

Annual calendar year accounting for chain ladder method

Assume that the data is per RRRRMM30 (or 31, it's the same thing for the program counts business days). Assume that the last claim date is RRRRMM30. (It is not advisable to put a later claim period, for then the program believes that there has been no development after RRRRMM30.)
If month MM < 12 two runs are needed:1. Check the Best cal.year box in the menu.
Then a preparatory chain ladder calculation is made that draws the development to the end of the year, with period length adapted to month MM. For example, if MM = 10, the period-length will be two months, for then each calendar year, including hitherto developed in the current year, consists of a certain number of two-month periods. If MM = 9, the period-length will be three months (quarter), for then each calendar year, including hitherto developed in the current year, consists of a certain number of quarters.

Output is then for calendar year, but for the current year accounting is only for the period up to and including month MM. To get a full-year estimate of the outcome it is however not appropriate to multiply by 12/MM, because the remaining months have a greater proportion of development the following years than the past months. Development factors are calculated as if the remaining months of this year were included, so multiplying by 12/MM gives on average an overestimation of the outcome.

This account thus fits the years RRRR - 1 RRRR - 2, . , but not RRRR.

2. Uncheck the Best cal.year box in the menu.
Set Period to MM, eg 10 if MM = 10. Then the outcome of the last period can be multiplied by 12/MM for a full-year estimate, of course apart from seasonal variations.

Bich stochastic reserving menu

Runs as Reserves. The same parmfile can be used for both applications. The menu looks like this.

Same coloring of parameters as in Reserves. This should help, since I have not succeeded completely in putting corresponding parameters in the same places as in Reserves.

Appendix 1

Including LofBus as index variable whose reserve sums are reproduced in new GLM statres files
2016-04-23, Stig Rosenlund

On the first white line, 1 2 denotes Region in the underscore-separated variable Regetc. 4 2 denotes LofBus. This entails segmentation on the combination of Region and LofBus. Each such combination is reserved separately in the step before GLM.
Om the white lines where Old word numbers and New word numbers are written, the word number 7 has been appended to 2 6. This means that blank-separated word no 7, which contains LofBus, is included in the index variable combinations that shall have their reserve sums reproduced from non-GLM to GLM+adjustments reserves R4.
The more detailed segmentation in step before GLM means that the total reserve can be different and that possibly some claims will not get any positive Statres, although they did get a Statres without LofBus. This is because that combinations might appear without any finalized claims and hence cannot get any reserve. See Bich.pdf. On the last page under B5 is explained how Rapp merges quantile intervals sometimes. Merging groups is not however done over arguments such as Region and LofBus. It would not be appropriate; better to let the user design the grouping to avoid segments without finalized claims.

As you can read in the upper part of Adv, the numbers 1 2 etc denote pairs of starting position and number of positions for the parts of the argument that are to be used. I illustrate with a line from some data, putting starting position numers above the line. It should make clear why 1 2 denotes only region and 1 2 4 2 denotes the combination of Region and LofBus.

```
000000000111
123456789012
38_02_VF_011
```

Another illustration.

```
00000000011111111122
12345678901234567890 1
F_BIGC-38_M_HC_09_P_1
T_COUN-23_F_VL_07_Q_2
```

1 1   3 4 selects both fire/theft and bigcity/smallcity/countryside to determine R1. (I have written several blanks between pairs to emphasize that the numbers come in pairs.) If you want M/F (gender) also, then write 1 1   3 4   11 1.

The use of starting positions and number of positions as parameters works because of Rappmenus' requirement of fixed length for each argument.