Using reinstatements

Paid reinstatements on an occurrence layer, with a ceded premium that varies with the annual recovery

Stephen J. Mildenhall

2026-10-30

An occurrence layer y xs a recovers $R = \sum_i \min((X_i - a)^+, y)$ in a year from the claims $X_1, \dots, X_N$. Property catastrophe and per-risk treaties rarely give that cover without limit. Each loss to the layer uses up part of its limit, and the cedant **reinstates** the limit by paying a **reinstatement premium**, pro rata to the amount reinstated but not to the time left in the year. With base premium B, rate on line $r = B/y$, and m reinstatements priced at multiples $\alpha_1, \dots, \alpha_m$ of r, the recovery and the ceded premium are

$$\min(R, (m+1)y) \qquad \text{and} \qquad B + r \sum_{j=1}^{m} \alpha_j \min\big((R - (j-1)y)^+, y\big).$$

Both are nondecreasing functions of R, so the ceded premium is random and is largest in the years when the recovery is largest.

The `reinstatements` clause decorates one occurrence layer and needs a base premium clause, `rol`, `deposit`, or `rate`, to fix r. The schedule is written in treaty language, with `free` for a multiplier of zero and counts as digits or the words one to five, or as an explicit list of multipliers. The form `no reinstatements` gives one annual limit and no reinstatement premium; omitting the clause leaves the cover free and unlimited. A `pnl` or `xpnl` books the random premium (see Using pnl and xpnl); a plain `agg` has no premium context and ignores the clause with a warning. Reinstatements are the occurrence counterpart of the variable features on aggregate covers: swing rating, the sliding scale, the profit commission, and the loss corridor.

```
occurrence net of 1000 xs 1000 rol 0.45 reinstatements 1 at 100%
1000 xs 1000 rol 0.45 reinstatements 1 free and 2 at 100%   # four annual limits
1000 xs 1000 rol 0.45 reinstatements [0 1 1]                # the same schedule
1000 xs 1000 deposit 450 reinstatements two at 50%
1000 xs 1000 rol 0.45 no reinstatements                     # one annual limit
```

```
import pandas as pd
from aggregate import build, qd, __version__
from style import wprint, GT, refuse, figures
figures()
print(__version__)
```

```
1.43.1
```

A reinstated layer on the base book

The variable feature pages share one base book: premium 10,000 at a 70% loss ratio, a policy limit of 2,000, lognormal severity with mean 150 and CV 2.5, gamma mixed claim counts with mixing CV 0.25, and expenses of 25% of premium. The layer 1000 xs 1000 is a working layer, with a rate on line of 45% and one reinstatement at 100%. About one claim a year exceeds 1,000, and 0.27 a year reach the policy limit and exhaust the layer.

```
book = '''agg Book
  10000 premium at 0.7 lr
  2000 xs 0
  sev lognorm 150 cv 2.5 {occurrence}
  mixed gamma 0.25 {aggregate}'''
program = '''xpnl {name}
  inherit premium
  less
    ''' + book + '''
  less
    0.25 premium expenses'''
layer = 'occurrence net of 1000 xs 1000 rol 0.45 {} as Layer'
x = build(program.format(name='Reinstated', aggregate='',
                         occurrence=layer.format('reinstatements 1 at 100%')))
qd(x)
e = x.engine
print(f'claims a year over 1,000 {e.n * e.sevs[0].sf(1000):.4f}, '
      f'at 2,000 {e.n * e.sevs[0].sf(1999.99):.4f}')
```

```
PnL('Reinstated': 2 groups, 5 legs -> 'margin')

                         EX     SD       CV     Skew    P01  Median   P99
Step  Side
Gross Consideration   10000      0        0           10000   10000 10000
      Obligation      -9500 2704.6  -0.2847 -0.67669 -17080   -9200 -4640
      Margin            500 2704.6   5.4092 -0.67669  -7080     800  5360
Layer Consideration -635.95 194.51 -0.30587 -0.38086   -900    -549  -450
      Obligation     508.15 602.68    1.186   1.0039      0     220  2000
      Margin        -127.8 427.51  -3.3453   1.5169   -450    -329  1100
      Net            372.2 2450.8   6.5847  -0.6246  -6420     620  4930
All   Consideration  9364.1 194.51 0.020772 -0.38086   9100    9451  9550
      Obligation    -8991.9   2358 -0.26223 -0.64943 -15580   -8740 -4620
      Margin          372.2 2450.8   6.5847  -0.6246  -6420     620  4930
      Impact         -127.8 427.51  -3.3453   1.5169   -450    -329  1100
claims a year over 1,000 0.9925, at 2,000 0.2705
```

The `Layer` step shows the effect of the clause. The ceded premium has mean 635.95 and standard deviation (SD) 194.51. It runs from the base premium of 450 in a year with no recovery to 900 once the first limit is used up and reinstated. The recovery has mean 508.15, and its 99th percentile, 2,000, is the cap of two limits. Net of the layer, the cedant's mean result falls from 500 to 372.20 and its SD from 2,704.60 to 2,450.82.

Ceded premium and recovery under three reinstatement terms

The same layer at the same rate on line is built with no clause, which is free and unlimited cover, with one reinstatement at 100%, and with no reinstatements. The table shows the mean and SD of the ceded premium, the mean and 99th percentile of the recovery, and the mean and SD of the cedant's net result.

```
terms = {'none': '', '1 at 100%': 'reinstatements 1 at 100%',
         'no reinstatements': 'no reinstatements'}
rows = {}
for k, v in terms.items():
    s = build(program.format(name='R', occurrence=layer.format(v),
                             aggregate='')).summary_df
    rows[k] = [-s.loc[('Layer', 'Consideration'), 'EX'],
               s.loc[('Layer', 'Consideration'), 'SD'],
               s.loc[('Layer', 'Obligation'), 'EX'],
               s.loc[('Layer', 'Obligation'), 'P99'],
               s.loc[('All', 'Margin'), 'EX'], s.loc[('All', 'Margin'), 'SD']]
df = pd.DataFrame.from_dict(rows, orient='index', columns=[
    'premium', 'premium SD', 'recovery', 'recovery P99', 'net', 'net SD'])
df.index.name = 'reinstatements'
GT(df, float_format=',.2f')
```

reinstatements	premium	premium SD	recovery	recovery P99	net	net SD
none	450.00	0.00	525.16	2,660	575.16	2,325.77
1 at 100%	635.95	194.51	508.15	2,000	372.20	2,450.82
no reinstatements	450.00	0.00	413.21	1,000	463.21	2,471.50

Without the clause the layer recovers 525.16 on average for a fixed premium of 450, and its recovery exceeds two limits in more than 1% of years. One paid reinstatement caps the recovery at 2,000, which takes 17.01 off the mean recovery, and adds a mean reinstatement premium of 185.95. That premium is the rate on line, 0.45, times 413.21, the mean recovery from the first limit alone, which is the recovery under `no reinstatements`. The reinstatement premium is due in the years with large recoveries, so it offsets part of the protection: the cedant's net SD under one paid reinstatement, 2,450.82, sits between the 2,325.77 of unlimited cover and the 2,471.50 of a single limit.

Refused programs

A `reinstatements` clause needs a base premium on its layer, decorates an occurrence layer only, and requires that layer to be the only occurrence layer.

```
reinst = 'rol 0.45 reinstatements 1 at 100%'
refuse(program.format(name='R1', aggregate='',
                      occurrence='occurrence net of 1000 xs 1000 '
                                 'reinstatements 1 at 100%'),
       program.format(name='R2', occurrence='',
                      aggregate=f'aggregate net of 4000 xs 8000 {reinst}'),
       program.format(name='R3', aggregate='',
                      occurrence='occurrence net of 500 xs 500 rol 0.9 and '
                                 f'1000 xs 1000 {reinst}'))
```

```
DecL: a 'reinstatements' clause needs a base premium clause (deposit / rol /
rate) on the same layer; the base rate on line r = base_premium / limit would
otherwise be undefined ([reins-premium]).
DecL: a 'reinstatements' clause decorates an occurrence layer, not aggregate
reinsurance.
DecL: 'reinstatements' require a single occurrence layer; found 2
([reins-single-layer]). A reinstated layer cannot share the occurrence tier with
other layers (the engine ceiling is the 2-D (L, R) joint).
```