Using profit commissions

Profit commissions on an aggregate quota share with the DecL pc clause

Stephen J. Mildenhall

2026-11-01

A **profit commission** returns to the cedant a share of the reinsurer's profit on a treaty, settled after the year from the treaty's own experience. It is common on quota shares and appears on some excess of loss treaties. With ceded loss A, a fixed ceded premium P_C and ceded loss ratio $\lambda = A/P_C$, a profit share s and an allowance a for the reinsurer's expenses and margin, the commission is

$$K = s\,(1 - \lambda - a)^+ \, P_C,$$

which is zero once the loss ratio reaches $1 - a$. As a function of the loss ratio it is a put option struck at $1 - a$: the cedant shares in a good year and pays nothing back in a bad one.

In DecL the clause `pc <share> after <allowance>` decorates one layer of an `aggregate net of` program, and the layer needs a `deposit`, `rol`, or `rate` premium to form the loss ratio. The profit commission is one of the variable features of DecL, with sliding scale commissions, loss corridors, swing rating, and retro rating. A program takes at most one of them, the reinsurance features decorate aggregate covers only, and all take effect in a `pnl` or `xpnl` (see Using pnl and xpnl). The commission is booked in the expense column of the cover, `E` in `economic_ratios_df`, as a random credit to the cedant.

```
aggregate net of 50% po inf xs 0 rate 100% pc 0.35 after 0.1 as QS
aggregate net of 50% po inf xs 0 rate 100% pc 50% after 15% as QS
aggregate net of 2000 xs 8000 rate 6% pc 0.25 after 0.2 as "Stop loss"
```

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 setup cell wraps the book in an `xpnl` with a slot for each reinsurance program.

```
import numpy as np
import pandas as pd
from aggregate import build, qd, __version__
from style import wprint, GT, refuse, figures, figure
figures()
print(__version__)
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'''
def quota_share(name, terms):
    """The base book under a 50% quota share at a 100% rate with the given terms."""
    return build(program.format(name=name, occurrence='',
                 aggregate=f'aggregate net of 50% po inf xs 0 rate 100% {terms} as QS'))
```

```
1.53.0
```

A profit commission on a 50% quota share

The cover is a 50% quota share of the base book at 100% of the subject premium, so the ceded premium is 5,000 and the ceded loss ratio equals the gross loss ratio, with mean 70%. The reinsurer returns 35% of its profit after an allowance of 10%, a commission rate of $0.35(0.9 - \lambda)^+$: 7% at the expected loss ratio and at most 31.5%, in a year with no loss. The quota share pays no other commission.

```
pc = quota_share('PC', 'pc 0.35 after 0.1')
print(pc.engine.variable_terms)
qd(pc)
```

```
ProfitCommissionTerms(share=0.35, allowance=0.1)
PnL('PC': 2 groups, 6 legs -> 'margin')

                         EX     SD       CV     Skew     P01  Median     P99
Step  Side
Gross Consideration   10000      0        0            10000   10000   10000
      Obligation      -9500   2704 -0.28463 -0.67687  -17071   -9193   -4633
      Margin            500   2704    5.408 -0.67687   -7071     807    5367
QS    Consideration   -5000      0        0            -5000   -5000   -5000
      Obligation     3923.6 1037.9  0.26452   1.2482  2268.2  3750.2  7285.5
      Margin        -1076.4 1037.9 -0.96419   1.2482 -2731.8 -1249.8  2285.5
      Net           -576.43 1679.7  -2.9141 -0.37235 -4785.5 -442.78  2635.2
All   Consideration    5000      0        0             5000    5000    5000
      Obligation    -5576.4 1679.7 -0.30122 -0.37235 -9785.5 -5442.8 -2364.8
      Margin        -576.43 1679.7  -2.9141 -0.37235 -4785.5 -442.78  2635.2
      Impact        -1076.4 1037.9 -0.96419   1.2482 -2731.8 -1249.8  2285.5
```

The `QS` step is the cover from the cedant's side: the ceded premium of 5,000 against the recovery and the profit commission, which together make the obligation row, with mean 3,924. The ledger `economic_df` lists the profit commission as its own leg, with mean 424 and standard deviation 349. It is zero in the 1 in 100 bad year, column `κ01`, and 1,202 in the 1 in 100 good year, `κ99`.

```
GT(pc.economic_df.loc['QS', ['EX', 'SD', 'κ01', 'κ50', 'κ99']], float_format=',.0f')
```

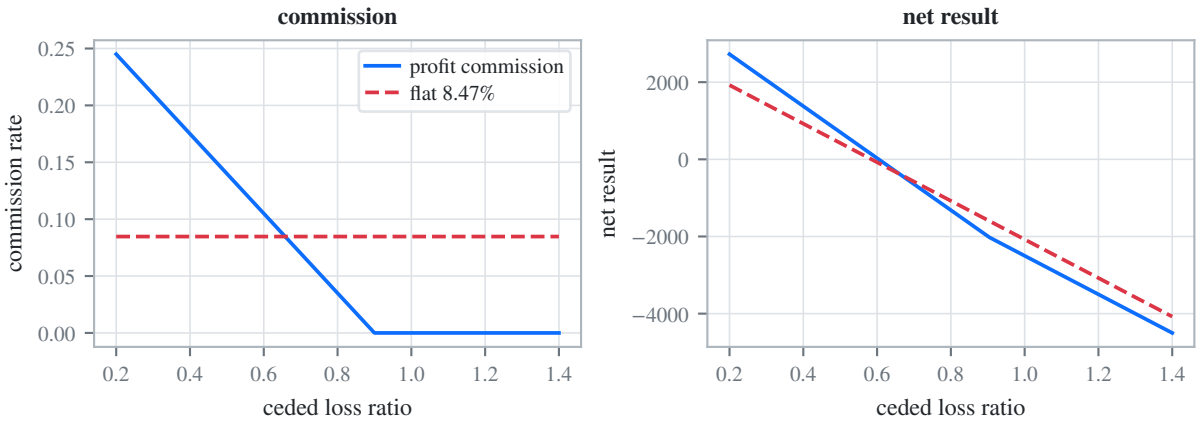
Side	*Label*	EX	SD	κ01	κ50	κ99
Consideration	QS premium	-5,000	0	-5,000	-5,000	-5,000
Obligation	QS recovery	3,500	1,352	7,286	3,346	1,066
	profit commission	424	349	0	404	1,202
	Total	3,924	1,038	7,286	3,750	2,268
Margin	Total	-1,076	1,038	2,286	-1,250	-2,732
	Net	-576	1,680	-4,786	-443	2,635

The mean commission, 424, is 8.47% of the ceded premium, more than the 7% paid at the expected loss ratio. Since $x^+ = x + (-x)^+$, the mean rate is $\mathsf{P}\,0.35(0.9 - \lambda)^+ = 0.07 + 0.35\,\mathsf{P}(\lambda - 0.9)^+$. The commission is floored at zero, so the years with a loss ratio above 90% do not reduce it, and they add 1.47 points to its mean.

Profit commission and net result against a flat commission

To compare shapes rather than prices, the flat cover pays a ceding commission equal to the expected profit commission, 8.47% of the ceded premium. The gross loss is $10{,}000\lambda$ and the recovery $5{,}000\lambda$, so the net result, premium less expense and ceded premium, plus commission and recovery, less gross loss, is $2{,}500 + 5{,}000(c(\lambda) + \lambda) - 10{,}000\lambda$ for a commission rate c. The left panel shows the two commission rates and the right panel the two net results.

```
rate = -pc.economic_ratios_df.loc['QS', 'E'] / 5000
flat = quota_share('Flat', f'cede {rate:.6f}')
lr = np.linspace(0.2, 1.4, 241)
rates = {'profit commission': pc.engine.variable_terms.phi(lr), f'flat {rate:.2%}':
np.full_like(lr, rate)}
fig, (ax0, ax1) = figure('full', cols=2)
for (k, c), ls in zip(rates.items(), ['-', '--']):
    ax0.plot(lr, c, ls=ls, label=k)
    ax1.plot(lr, 10000 - 2500 - 5000 + 5000 * (c + lr) - 10000 * lr, ls=ls, label=k)
ax0.set(xlabel='ceded loss ratio', ylabel='commission rate', title='commission')
ax1.set(xlabel='ceded loss ratio', ylabel='net result', title='net result')
ax0.legend();
```



Below a loss ratio of 90% the commission rate falls 0.35 points per point of loss ratio, so the net result falls at 1.35 times the slope of the flat cover. The two lines cross at a loss ratio of about 66%, below which the profit commission pays more, and above 90% they are parallel, the profit commission 424 below. The table compares the two covers over the whole distribution, with `P01` the 1 in 100 bad result.

```
df = pd.DataFrame({k: [-x.economic_ratios_df.loc['QS', 'E'], x.est_m, x.est_sd, x.q(0.01),
x.cdf(0)]
                   for k, x in {'flat': flat, 'profit commission': pc}.items()},
                  index=['commission', 'mean', 'SD', 'P01', 'Pr(loss)']).T
df.index.name = 'cover'
GT(df, float_format=',.0f', ratio_cols=['Pr(loss)'])
```

cover	commission	mean	SD	P01	Pr(loss)
flat	424	-576	1,352	-4,362	62.9%
profit commission	424	-576	1,680	-4,786	60.0%

The two covers have the same mean result, −576, which is negative because the cedant keeps all of its 2,500 expense against half the premium. The profit commission raises the standard deviation (SD) of the net result from 1,352 to 1,680 and lowers `P01` from −4,362 to −4,786, by the 424 of flat commission it does not pay in a bad year. The probability of a loss, a result of zero or less, falls from 62.9% to 60.0%, since the profit commission pays more than the flat commission in good years.

Refused programs

The profit share must lie between 0 and 1. A program takes at most one variable feature, so a profit commission on the quota share and a corridor on a second aggregate layer, 2000 xs 8000, in the same program are refused. A profit commission replaces the flat commission across its aggregate program, so a `cede` on the same layer or on another aggregate layer is refused too.

```
refuse(lambda: quota_share('R1', 'pc 1.2 after 0.1'),
       lambda: quota_share('R2', 'pc 0.35 after 0.1 and 2000 xs 8000 rate 6% corridor 0.5 po 0.5
       xs 1'),
       lambda: quota_share('R3', 'cede 0.25 pc 0.35 after 0.1'),
       lambda: quota_share('R4', 'pc 0.35 after 0.1 and 2000 xs 8000 rate 6% cede 0.2'))
```

```
ProfitCommissionTerms: share must be in [0, 1], got 1.2.
DecL: at most one variable-rating feature per program (decision 0: no stacking);
found 2.
DecL: 'pc' replaces the fixed 'cede' commission; give one or the other.
DecL: 'pc' replaces the fixed 'cede' commission across its aggregate program, so
a 'cede' on another layer would not be booked; give one or the other.
```