Package 'rfcipReSim'

July 22, 2025

Type Package

Title A Modular Simulator for FCIP Reinsurance Outcomes

Version 0.0.0.9000

Author Francis Tsiboe [aut, cre] (<https://orcid.org/0000-0001-5984-1072>)

Maintainer Francis Tsiboe <ftsiboe@hotmail.com>

Contributor - Dylan Turner [aut, ctb] (<https://orcid.org/0000-0002-0915-7384>)

Reviewer -

Creator Francis Tsiboe

Description A modular framework for simulating reinsurance outcomes under the Federal Crop Insurance Program (FCIP). Provides seamless integration with the 'rfcip', 'rmaADM', 'rfcipCalibrate', and 'rfcipCalcPass' packages to streamline data ingestion, scenario generation, risk-sharing computations, and calibration workflows.

License GPL-3 + file LICENSE

URL https://github.com/you/rFarmPolicySim

BugReports https://github.com/you/rFarmPolicySim/issues

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

VignetteBuilder knitr

Depends R (>= 4.1.0)

Imports data.table, dplyr, tidyr

Suggests rmaADM, rfcip, knitr, rmarkdown, testthat (>= 3.0.0)

Remotes github::dylan-turner25/rmaADM

LazyData true

Cite-us If you find it useful, please consider staring the repository and citing the following studies
- Tsiboe, F. and Turner, D. (2025). ``Incorporating buy-up price loss coverage into the United States farm safety net." Applied Economic Perspectives and Policy.
- Tsiboe, F., et al. (2025). ``Risk reduction impacts of crop insurance in the United States." Applied Economic Perspectives and Policy.
- Gaku, S. and Tsiboe, F. (2024). Evaluation of alternative farm safety net program combination strategies. Agricultural Finance Review.

Contents

`allocate_gain` *Allocate a premium gain across SRA gain-ratio tiers*

Description

Mirrors `allocate_loss()`, but for the case where premium exceeds indemnity. Dollar gains are apportioned upward through the gain tiers and multiplied by the retention percentages supplied.

Usage

```
allocate_gain(tiers_df, outstanding_gain)
```

Arguments

`tiers_df`	A `data.frame` with `tier_factors`, `lower_limit`, `upper_limit`. Rows must be sorted ascending by `lower_limit`.
`outstanding_gain`	Numeric, the excess premium dollars (`premiums` - `indemnities`).

Value

A single numeric **positive** value (cash inflow).

See Also

Other FCIP Re-insurance Calculators: `allocate_loss()`, `assign_state_group()`, `fcip_reinsurance_dispatcher()`, `fund_alocation_and_retention()`, `non_proportional_reinsurance()`, `proportional_split_and_quota_share()`, `reinsurance_control()`, `revealed_aip_state_operation()`

Examples

```
## Not run:
tiers <- data.frame(tier_factors = c(0.40, 0.05),
                    lower_limit  = c(0,50),
                    upper_limit  = c(50, 100))
allocate_gain(tiers, outstanding_gain = 75)

## End(Not run)
```

allocate_loss *Allocate an indemnity (loss) across SRA loss-ratio tiers*

Description

Given a data frame that defines tier boundaries in dollar terms and the AIP/FCIC retention percentage for each tier, distribute a single indemnity amount down the column until it is exhausted, then compute the cash flow for the party whose retention factors are supplied.

Usage

```
allocate_loss(tiers_df, outstanding_loss)
```

Arguments

tiers_df	A data.frame with at least the columns • tier_factors - numeric vector of retention percentages • lower_limit, upper_limit - dollar boundaries of each tier The rows must be sorted ascending by lower_limit; the last upper_limit may be Inf.
outstanding_loss	Numeric, the total indemnity dollars that exceed the premium (i.e., indemnities - premiums).

Value

A single numeric value negative for the cash outflow (consistent with SRA convention: losses are negative income).

See Also

Other FCIP Re-insurance Calculators: allocate_gain(), assign_state_group(), fcip_reinsurance_dispatcher(), fund_alocation_and_retention(), non_proportional_reinsurance(), proportional_split_and_quota_share_reinsurance_control(), revealed_aip_state_operation()

Examples

```
## Not run:
tiers <- data.frame(tier_factors = c(0.65, 0.45, 0.10, 0),
                    lower_limit  = c(100, 160, 220, 500),
                    upper_limit  = c(160, 220, 500, Inf))
allocate_loss(tiers, outstanding_loss = 300)

## End(Not run)
```

assign_state_group *Map a U.S. state or territory to its SRA* state-group

Description

The Standard Reinsurance Agreement splits states into three premium-volume groups. This helper reproduces Appendix II of the 2025-26 SRA verbatim.

Usage

```
assign_state_group(state, control = reinsurance_control())
```

Arguments

state	A two-letter FIPS postal abbreviation (character scalar).
control	control list of SRA treaty parameters; see reinsurance_control().

Value

Integer 1, 2, or 3.

See Also

Other FCIP Re-insurance Calculators: allocate_gain(), allocate_loss(), fcip_reinsurance_dispatcher(), fund_alocation_and_retention(), non_proportional_reinsurance(), proportional_split_and_quota_share reinsurance_control(), revealed_aip_state_operation()

Examples

```
## Not run:
assign_state_group("IA")   # 1
assign_state_group("TX")   # 2
assign_state_group("PR")   # 3

## End(Not run)
```

fcip_reinsurance_dispatcher *End-to-end FCIP re-insurance dispatcher*

Description

Runs a *complete* Federal Crop Insurance Program re-insurance pipeline on a policy-level book:

1. Calls fund_alocation_and_retention() to tag every policy with its SRA fund ("ARF" / "CF") and the proportional retention share chosen in the Plan of Operations.
2. Executes the simple proportional split **plus** the mandatory quota-share via proportional_split_and_quota_shar
3. Aggregates the retained dollars to each *state by fund* block and pushes them through the Appendix-II non-proportional bands with non_proportional_reinsurance() (once for the AIP, once for FCIC, using the hard-coded 2025-26 SRA factors-overridable via arguments).
4. Re-distributes the state-fund non-proportional result back to individual policies pro-rata on their AIP underwriting result, yielding a final aip_final_after_all per policy.

Usage

```
fcip_reinsurance_dispatcher(
  policies,
  aip_state_operation = NULL,
  ELR_calc = NULL,
  assign_state_group_fun = assign_state_group,
  allocate_loss_fun = allocate_loss,
  allocate_gain_fun = allocate_gain,
  control = reinsurance_control()
)
```

Arguments

policies	A data.table with, at minimum, state, total_premium_amount, indemnity_amount and any join keys required by aip_state_operation. Extra columns are preserved.
aip_state_operation	State-level Plan of Operations table. Defaults to the historical averages from revealed_aip_state_operation() but may be replaced with your own planned figures. Must contain arf_share and cf_retention.
ELR_calc	Optional function that takes the policy table and returns it with an elr column for ranking. If NULL, the fallback $ELR =$ premium/liability is used.
assign_state_group_fun	Function that maps a state/territory code to SRA state group; defaults to assign_state_group().
allocate_loss_fun	Function that performs the loss allocation; defaults to the internal allocate_loss().
allocate_gain_fun	Function that performs the gain allocation; defaults to the internal allocate_gain().
control	control list of SRA treaty parameters; see reinsurance_control().

Details

All default retention vectors reproduce Appendix II of the 2025-26 SRA. Override them to back-cast older agreements or to run what-if analyses.

Value

A data.table identical to policies but with, at the end of the pipeline, these extra columns:

fund ARF or CF.

retention_prop AIP proportional share (0.20 in ARF, 0.35-1.00 in CF).

aip_underwriting Policy-level underwriting result after the proportional split, before quota-share.

aip_underwriting_after_quota_share After 6.5 % cession.

aip_final_after_all Final AIP result after the non-proportional state-fund band is allocated back to policies.

See Also

Other FCIP Re-insurance Calculators: allocate_gain(), allocate_loss(), assign_state_group(), fund_alocation_and_retention(), non_proportional_reinsurance(), proportional_split_and_quota_share reinsurance_control(), revealed_aip_state_operation()

Examples

```
## Not run:
result <- fcip_reinsurance_dispatcher(my_policy_dt)
head(result)

## End(Not run)
```

`fund_alocation_and_retention` *Allocate policies to SRA funds and append retention percentages*

Description

Combines a **policy-level data table** with a `revealed` or planned state-level Plan of Operations to decide

- which *fund* each policy is ceded to - Assigned-Risk Fund (`"ARF"`) vs. Commercial Fund (`"CF"`); and
- what *proportion of premium & losses* (`retention_prop`, 0-1) the Approved Insurance Provider (AIP) must keep on that policy, as required by the Standard Reinsurance Agreement (20 % in the ARF; 35-100 % in the CF).

Usage

```
fund_alocation_and_retention(
  policies,
  aip_state_operation = NULL,
  ELR_calc = NULL
)
```

Arguments

`policies` A `data.table` with at minimum

- `state` - two-letter postal code;
- `total_premium_amount` - gross premium dollars;
- `liability_amount` - liability dollars (only used by the default ELR formula);
- any join key(s) that also appear in `aip_state_operation` (usually `state` and optionally `commodity_year`).

Extra columns are preserved.

`aip_state_operation`
A state-level Plan of Ops table. Defaults to the historical averages returned by `revealed_aip_state_operation()`, but you can pass your own `data.table` with (at least) columns `state`, `arf_share`, `cf_retention`.

`ELR_calc` `NULL` or a user-supplied function `f(dt)` that returns the input `data.table` with an `elr` column. If `NULL`, the fallback definition $ELR = \frac{\text{total premium}}{\text{liability}}$ is applied (edit that line to use your own quick proxy).

Details

The function reproduces the `ranking` logic most companies use: within each state, sort policies by *expected loss ratio* so the riskiest premium is pushed into the ARF first, up to the state historical `arf_share`. All remaining business stays in the CF.

Value

The original `policies` table (copied, not modified in place) with **four new columns**:

`state_group` SRA group 1/2/3, via `assign_state_group()`.

`fund` Character - `"ARF"` or `"CF"`.

`retention_prop` Numeric 0-1 - AIP share of premium & loss on this policy (0.20 in ARF, user-selected 0.35-1 in CF).

`elr` Expected loss ratio used for the ranking step.

See Also

Other FCIP Re-insurance Calculators: `allocate_gain()`, `allocate_loss()`, `assign_state_group()`, `fcip_reinsurance_dispatcher()`, `non_proportional_reinsurance()`, `proportional_split_and_quota_share()`, `reinsurance_control()`, `revealed_aip_state_operation()`

Examples

```
## Not run:
policies_aug <- fund_alocation_and_retention(
                  policies = my_policy_dt,
                  aip_state_operation = revealed_aip_state_operation(),
                  ELR_calc = NULL)
head(policies_aug)

## End(Not run)
```

---

non_proportional_reinsurance

Compute AIP or FCIC dollar share under SRA non-proportional reinsurance

---

Description

A one-stop wrapper that (i) chooses the correct SRA retention vector based on fund, state group and agent; (ii) builds tier boundaries from user-supplied premium and indemnity totals; and (iii) hands off to `allocate_loss()` or `allocate_gain()` as needed.

Usage

```
non_proportional_reinsurance(
  premiums,
  indemnities,
  fund,
  state,
```

```
  agent,
  assign_state_group_fun = assign_state_group,
  allocate_loss_fun = allocate_loss,
  allocate_gain_fun = allocate_gain,
  control = reinsurance_control()
)
```

Arguments

premiums	Gross premium dollars for the book being analysed.
indemnities	Indemnity dollars paid or expected.
fund	Character: "ARF" (Assigned-Risk Fund) or "CF" (Commercial Fund).
state	Two-letter state or territory code.
agent	Character: "AIP" for the private insurer, or "FCIC" for the federal complement.
assign_state_group_fun	Function that maps a state/territory code to SRA state group; defaults to assign_state_group().
allocate_loss_fun	Function that performs the loss allocation; defaults to the internal allocate_loss().
allocate_gain_fun	Function that performs the gain allocation; defaults to the internal allocate_gain().
control	control list of SRA treaty parameters; see reinsurance_control().

Details

All default retention vectors reproduce Appendix II of the 2025-26 SRA. Override them to back-cast older agreements or to run what-if analyses.

Value

Numeric dollar amount: negative for losses paid, positive for gains retained, from the perspective of the chosen agent.

See Also

Other FCIP Re-insurance Calculators: allocate_gain(), allocate_loss(), assign_state_group(), fcip_reinsurance_dispatcher(), fund_alocation_and_retention(), proportional_split_and_quota_share(reinsurance_control(), revealed_aip_state_operation()

Examples

```
## Not run:
# A simple Iowa example (Commercial Fund, Group 1, small loss):
non_proportional_reinsurance(
  premiums    = 100,
  indemnities = 120,
  fund        = "CF",
  state       = "IA",
  agent       = "AIP")

## End(Not run)
```

proportional_split_and_quota_share
Apply SRA proportional split + national quota-share to a policy table

Description

Apply SRA proportional split + national quota-share to a policy table

Usage

```
proportional_split_and_quota_share(policies, control = reinsurance_control())
```

Arguments

policies	A data.table with at least columns premium, indemnity, retention_prop (0-1 fraction the AIP keeps).
control	control list of SRA treaty parameters; see reinsurance_control().

Value

A list with elements

- policies - the input table, now containing aip_gross_premium, aip_gross_indemnity, fcic_gross_premium, fcic_gross_indemnity, aip_underwriting, fcic_underwriting, quota_share_ceded, aip_underwriting_after_quota_share
- book_totals - one-row data.table with aip_final_net_underwriting and fcic_final_net_underwriting.

See Also

Other FCIP Re-insurance Calculators: allocate_gain(), allocate_loss(), assign_state_group(), fcip_reinsurance_dispatcher(), fund_alocation_and_retention(), non_proportional_reinsurance(), reinsurance_control(), revealed_aip_state_operation()

Examples

```
## Not run:
res <- apply_split_and_qs(policies_augmented)
res$book_totals

## End(Not run)
```

`reinsurance_control` *Controls of Standard Reinsurance Agreement (SRA) parameters*

Description

Convenience wrapper that packages every numeric knob used by the crop-insurance re-insurance pipeline into a single named list. Pass the list to higher-level functions (e.g. `fcip_reinsurance_dispatcher`()) to override any SRA default without cluttering those function calls with dozens of arguments.

Usage

```
reinsurance_control(
  quota_share = 0.065,
  aip_arf_loss_share = c(0, 0.075, 0.06, 0.03, 0),
  aip_arf_gain_share = c(0.225, 0.135, 0.03),
 aip_cf_loss_share = list(`1` = c(0, 0.65, 0.45, 0.1, 0), `2` = c(0, 0.425, 0.2, 0.05,
    0), `3` = c(0, 0.425, 0.2, 0.05, 0)),
 aip_cf_gain_share = list(`1` = c(0.75, 0.4, 0.05), `2` = c(0.975, 0.4, 0.05), `3` =
    c(0.975, 0.4, 0.05)),
  aip_loss_lr_lower_limits = c(1, 1.6, 2.2, 5),
  aip_gain_lr_lower_limits = c(0.65, 0.5),
 state_grouping = list(group_01 = c("IL", "IN", "IA", "MN", "NE"), group_02 = c("AL",
   "AZ", "AR", "CA", "CO", "FL", "GA", "ID", "KS", "KY", "LA", "MI", "MO", "MS", "MT",
    "NC", "ND", "NM", "OH", "OR", "SC", "SD", "TN", "TX", "VA", "WA", "WI", "OK"),
   group_03 = c("AK", "CT", "DE", "HI", "ME", "MA", "MD", "NV", "NH", "NJ", "NY", "PA",
    "RI", "UT", "VT", "WV", "WY", "PR", "VI", "GU", "AS", "DC"))
)
```

Arguments

`quota_share`	National quota-share rate (default 0.065 = 6.5%).
`aip_arf_loss_share`	Numeric(5) vector - AIP loss retention rates for the Assigned-Risk Fund (ARF) loss bands.
`aip_arf_gain_share`	Numeric(3) vector - AIP gain retention rates for the ARF gain bands.
`aip_cf_loss_share`	Numeric(5) vector - AIP loss retention rates for each state-group Commercial Fund (CF) loss bands (1, 2, 3).
`aip_cf_gain_share`	Numeric(3) vector - AIP gain retention rates for each state-group CF gain bands.
`aip_loss_lr_lower_limits`	Numeric(4) vector - lower loss-ratio break-points (defaults c(1, 1.6, 2.2, 5)).
`aip_gain_lr_lower_limits`	Numeric(2) vector - lower gain break-points (defaults c(0.65, 0.50)).
`state_grouping`	SRA state groups by premium-volume.

Value

A named `list` holding all arguments exactly as supplied.

See Also

Other FCIP Re-insurance Calculators: allocate_gain(), allocate_loss(), assign_state_group(), fcip_reinsurance_dispatcher(), fund_alocation_and_retention(), non_proportional_reinsurance(), proportional_split_and_quota_share(), revealed_aip_state_operation()

Examples

```
## Not run:
# build a control object that bumps the quota-share to 7 %
ctl <- reinsurance_control(quota_share = 0.07)
str(ctl)

## End(Not run)
```

`revealed_aip_state_operation` *Build a state-level* `revealed` *Plan of Operations from historical SRA data*

Description

Pulls the public `rfcip::stateSRA` history, collapses it to one row per state by reinsurance year and returns the observed (i) share of each state book ceded to the three SRA funds and (ii) average premium-retention percentage the AIP kept inside each fund.

Usage

```
revealed_aip_state_operation()
```

Details

The result is a ready-made `aip_state_operation`-style table that you can feed into simulation helpers such as `fund_alocation_and_retention()`.

The function:

1. Removes `Total` and `All Other States` rows.
2. Maps FCIC fund abbreviations to `arf`, `cf`, `df`.
3. Keeps only `gross_premium` and `retained_premium`.
4. Sums dollars by state by year by fund by value type.
5. Computes each fund state share and retention ratio.
6. Averages where multiple company records exist.

Value

A `data.table` with columns

- `state` - two-letter postal code
- `commodity_year` - FCIC reinsurance year (taken as commodity year)
- `arf_share`, `arf_retention`
- `cf_share`, `cf_retention`

- `df_share`, `df_retention`

where $fund\_share = \frac{\text{fund gross premium}}{\text{state total}}$, and $fund\_retention = \frac{\text{retained premium}}{\text{gross premium}}$ averaged across all companies writing in that state-fund-year.

See Also

Other FCIP Re-insurance Calculators: `allocate_gain()`, `allocate_loss()`, `assign_state_group()`, `fcip_reinsurance_dispatcher()`, `fund_alocation_and_retention()`, `non_proportional_reinsurance()`, `proportional_split_and_quota_share()`, `reinsurance_control()`

Examples

```
## Not run:
library(data.table)

state_ops <- revealed_aip_state_operation()
head(state_ops)


## End(Not run)
```

`sra_2026` *Standard Reinsurance Agreement - 2026*

Description

2026 Standard Reinsurance Agreement:

Usage

```
sra_2026
```

Format

An object of class `list` of length 8.

Details

Assigned Risk Fund Retention

- (A) The Company shall retain a 20 percent interest in premium and associated ultimate net losses in the Assigned Risk Fund in each State. The remainder is ceded to FCIC.
- (B) The associated net book premium of eligible crop insurance contracts assigned to the Assigned Risk Fund shall not exceed 75 percent of the Company net book premium in each State.
- (C) Unless otherwise specified in the Agreement, in the event the percentage of net book premium for eligible crop insurance contracts in the Assigned Risk Fund exceeds 75 percent of the aggregate net book premium for any State, the amount of premiums and associated liabilities in the Assigned Risk Fund will be reduced pro-rata to 75 percent and the excess will be assigned by FCIC to the Commercial Fund for that State.

Commercial Fund Retention

- (A) The Company shall retain at least a 35 percent interest in premium and associated ultimate net losses in the Commercial Fund in each State. The remainder shall be ceded to FCIC.
- (B) The retention percentage for the Commercial Fund in each State shall be made in 5 percent increments and designated in the Company Plan of Operations according to Appendix II.

Underwriting Loss (A) Commercial Fund After the retentions under paragraph (4), the amount of underwriting loss retained by the Company for the Commercial Fund will be calculated within each State as the sum of the following: (i) For that portion of the underwriting loss amount for which the Company loss ratio exceeds 100 percent and is less than or equal to 160 percent, the Company shall retain an amount of the underwriting loss equal to the product of the following:

- (i) For that portion of the underwriting loss amount for which the Company loss ratio exceeds 100 percent and is less than or equal to 160 percent, the Company shall retain an amount of the underwriting loss equal to the product of the following: * (I) Its retained net book premium; * (II) The lesser of the Company actual loss ratio or 160 percent, minus 100 percent; and * (III) The following percentage for the applicable State Group: State Group 1 = 65.0 percent; State Groups 2 and 3 = 42.5 percent
- (ii) For that portion of the underwriting loss amount for which the Company loss ratio exceeds 160 percent and is less than or equal to 220 percent, the Company shall retain an amount of the underwriting loss equal to the product of the following:
  - (I) Its retained net book premium
  - (II) The lesser of the Company actual loss ratio or 220 percent, minus 160 percent; and
  - (III) The following percentage for the applicable State Group: State Group 1 = 45.0 percent; State Groups 2 and 3 = 20.0 percent #’
  - (iii) For that portion of the underwriting loss amount for which the Company loss ratio exceeds 220 percent and is less than or equal to 500 percent, the Company shall retain an amount of the underwriting loss equal to the product of the following:
  - (I) Its retained net book premium;
  - (II) The lesser of the Company actual loss ratio or 500 percent, minus 220 percent; and
  - (III) The following percentage for the applicable State Group: State Group 1 10.0 percent State Groups 2 and 3 5.0 percent
  - (iv) FCIC will assume 100 percent of that portion of the underwriting loss amount for which the Company loss ratio exceeds 500 percent.

Index