Package: enrollcast (via r-universe)

June 22, 2026

Title Project School Enrollment with Grade Progression Ratios

Version 0.0.0.9000

Description Projects school enrollment using the cohort survival / grade progression ratio method, implemented as a matrix projection. Works at any level of aggregation and any number of grades. Provides functions to compute progression ratios from historical grade-level enrollment and to project future enrollment forward an arbitrary horizon.

License MIT + file LICENSE

URL https://gitlab.com/localopen/enrollcast

BugReports https://gitlab.com/localopen/enrollcast/-/work_items

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 8.0.0

Depends R (>= 4.0)

Imports stats, utils

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Repository https://rorylawless.r-universe.dev

Date/Publication 2026-06-22 04:12:29 UTC

RemoteUrl https://gitlab.com/localopen/enrollcast

RemoteRef HEAD

RemoteSha 6d1299220bc2a3f8041fbeb4bac8e4f7943862b9

Contents

`progression_ratios` *Compute grade progression ratios*

Description

Calculates cohort survival / grade progression ratios from historical grade-level enrollment. For each non-entry grade, the ratio is enrollment in that grade divided by enrollment in the grade below one year earlier, summarised across the available year-to-year transitions.

Usage

```
progression_ratios(
  data,
  year = "year",
  grade = "grade",
  enrollment = "enrollment",
  method = c("mean", "geometric", "median", "last", "weighted"),
  n_years = NULL,
  weights = NULL,
  grade_order = NULL
)
```

Arguments

`data`	A long data frame of historical enrollment with one row per grade per year.
`year`, `grade`, `enrollment`	Column names in `data` (character scalars). Defaults are `"year"`, `"grade"`, `"enrollment"`.
`method`	How to summarise per-year ratios into one ratio per grade: `"mean"` (default), `"geometric"`, `"median"`, `"last"` (most recent transition only), or `"weighted"`.
`n_years`	Optional. Use only the most recent `n_years` transitions. If `n_years` exceeds the number of available transitions, all are used.
`weights`	For `method = "weighted"`, a numeric vector aligned most-recent to oldest, with one weight per transition year used.
`grade_order`	Optional character vector giving the low-to-high grade order. If omitted, factor levels, numeric ordering, or (with a warning) alphabetical ordering is used.

Value

A data frame with columns `grade_from`, `grade_to`, and `ratio`, one row per non-entry grade.

Examples

```
history <- data.frame(
  year = rep(2021:2023, each = 3),
  grade = factor(rep(c("K", "1", "2"), 3), levels = c("K", "1", "2")),
  enrollment = c(100, 90, 80, 110, 95, 88, 120, 99, 91)
)
progression_ratios(history)
```

`project_enrollment` *Project enrollment forward*

Description

Projects grade-level enrollment forward an arbitrary horizon using the grade progression ratio method. Internally builds a projection matrix from `ratios` and advances enrollment one year at a time (one matrix-vector product per projected year), overwriting the entry grade with the supplied exogenous value each year.

Usage

```
project_enrollment(
  base,
  ratios = NULL,
  horizon = NULL,
  entry = NULL,
  schedule = NULL,
  start_year = NULL
)
```

Arguments

`base`	Most recent observed enrollment: either a data frame with columns `grade` and `enrollment` (optionally `year`), or a named numeric vector (names are grades).
`ratios`	A data frame of progression ratios from `progression_ratios()`. Optional when a `schedule` is supplied.
`horizon`	Number of years to project (a positive integer).
`entry`	Exogenous entry-grade enrollment for each projected year: a numeric vector of length `horizon`, or a data frame with an `enrollment` or `value` column. If `NULL`, the entry grade is held constant at its base value and a warning is issued.
`schedule`	Optional prebuilt projection schedule: a list of per-year steps, each `list(matrix = <square projection` as produced by `swing_schedule()`. When supplied, `ratios` and `entry` must be `NULL` and `horizon` defaults to the schedule length. Step matrices must share identical grade dimnames, which determine the grade order `base` is aligned to.
`start_year`	Optional integer label for the base year; output years run from `start_year + 1`. If `NULL`, it is derived from a `year` column in `base` when present, otherwise output years are `1..horizon`.

Value

A long data frame with columns `year`, `grade`, and `enrollment`, covering the projected years only.

Examples

```
history <- data.frame(
  year = rep(2021:2023, each = 3),
  grade = factor(rep(c("K", "1", "2"), 3), levels = c("K", "1", "2")),
  enrollment = c(100, 90, 80, 110, 95, 88, 120, 99, 91)
)
ratios <- progression_ratios(history)
base <- subset(history, year == 2023, c("grade", "enrollment"))
project_enrollment(base, ratios,
  horizon = 3, entry = c(125, 130, 128),
  start_year = 2023
)
```

`projection_matrix` *Build the projection matrix*

Description

Assembles the projection matrix used to advance enrollment. Progression ratios are placed on the sub-diagonal (each non-entry grade is fed by the grade below); the entry-grade row is left at zero because entry enrollment is supplied exogenously to `project_enrollment()`.

Usage

```
projection_matrix(ratios, grade_order = NULL)
```

Arguments

`ratios`	A data frame with columns `grade_from`, `grade_to`, and `ratio`, as returned by `progression_ratios()`.
`grade_order`	Optional character vector giving the low-to-high grade order. If omitted, the order is reconstructed from the transition chain. Every non-entry grade in `grade_order` must appear as a `grade_to` in `ratios`.

Value

A square numeric matrix with grade dimnames.

Examples

```
ratios <- data.frame(
  grade_from = c("K", "1"),
  grade_to = c("1", "2"),
  ratio = c(0.92, 0.97)
)
projection_matrix(ratios)
```

`swing_schedule` *Build a swing/recovery projection schedule*

Description

Assembles a per-year `project_enrollment()` schedule for a school passing through a temporary relocation ("swing"): enrollment is held flat at the depressed observed level during the swing (identity steps), scaled by year-over-year recovery multipliers for the recovery window (diagonal steps), then projected with the grade progression ratio method (the normal projection matrix) for the remaining years.

Usage

```
swing_schedule(
  ratios,
  horizon,
  swing_years,
  recovery,
  entry = NULL,
  grade_order = NULL
)
```

Arguments

`ratios`	A data frame of progression ratios from `progression_ratios()`.
`horizon`	Number of years to project (a positive integer).
`swing_years`	Number of leading years the school is swinging (a non-negative integer); enrollment is held flat at `base`.
`recovery`	Recovery multipliers applied for one year each, immediately after the swing and compounding on the prior year: a numeric vector (whole-school, one multiplier per recovery year) or a grade-by-year numeric matrix (one row per grade). Use `numeric(0)` for no recovery window.
`entry`	Exogenous entry-grade enrollment for the normal (GPR) years only — the `horizon - swing_years - length(recovery)` years after recovery. Must be empty when there are no normal years.
`grade_order`	Optional low-to-high grade order, passed to `projection_matrix()`.

Value

A list of `horizon` projection steps suitable for the `schedule` argument of `project_enrollment()`.

Examples

```
ratios <- data.frame(
  grade_from = c("K", "1"), grade_to = c("1", "2"), ratio = c(0.92, 0.97)
)
schedule <- swing_schedule(ratios,
  horizon = 6, swing_years = 2,
  recovery = c(1.10, 1.10, 1.05), entry = 130
)
project_enrollment(c(K = 80, `1` = 66, `2` = 60), schedule = schedule)
```

Index