

Three outcomes that have to sum to one

70.4% promoted, 9.8% repeated, 10.2% dropped out, 5.7% finished and 2.9% transferred. Every one of those needs the same denominator, and the transfers are the reason it is not the enrolment count.

Pierre Robentz Cassion

Lesson 5 of 8

Education Programme Analysis — The cohort

What this lesson covers

- The year, as a set of exits
- The denominator is not the enrolment count
- Repetition is a decision, not an event
- Dropout by grade, and the grade-6 spike
- Report the exits, then the rates
- What comes next

The year, as a set of exits — In Python

```
import pandas as pd

enrolment = pd.read_csv("school-enrolment-2024.v1.csv")
clean = enrolment[enrolment["age_years"] <= 20]
cohort = clean[clean["school_year"] == 2023]

outcomes = cohort["end_of_year_status"].value_counts(normalize=True)
print((outcomes * 100).round(1))
print(f"\ncohort: {len(cohort)} students")
```

The year, as a set of exits — In R

```
library(dplyr)

enrolment |>
  filter(age_years <= 20, school_year == 2023) |>
  count(end_of_year_status) |>
  mutate(share = n / sum(n))
```

The year, as a set of exits

Outcome	Students	Share
Promoted	910	70.4%
Dropped out	132	10.2%
Repeated	127	9.8%
Completed the final grade	74	5.7%
Transferred out	37	2.9%
Still enrolled at the cut-off	12	0.9%

The year, as a set of exits

- The three headline rates are promotion, repetition and dropout, and they only mean anything if they share a denominator. . .

The denominator is not the enrolment count

- Transfers are not dropouts — A child who moved to another school is still in school
- Transfers are also not successes — They cannot be counted as promoted, because this register does not know whether they. . .

The denominator is not the enrolment count — In Python (cont.)

```
def rates(frame, exclude_transfers):
    base = frame[frame["end_of_year_status"] != "transferred-out"] \
        if exclude_transfers else frame
    return {
        "n": len(base),
        "promotion": (base["end_of_year_status"]
                      .isin(["promoted", "completed-final-grade"]).mean()),
        "repetition": base["end_of_year_status"].eq("repeated").mean(),
        "dropout": base["end_of_year_status"].eq("dropped-out").mean(),
    }

for exclude in (False, True):
    result = rates(cohort, exclude)
    label = "excluding transfers" if exclude else "all students"
    print(f"{label:20} n={result['n']} "
          f"promotion {result['promotion']:.1%} ")
```

The denominator is not the enrolment count — In Python (cont.)

```
f"repetition {result['repetition']:.1%}  "  
f"dropout {result['dropout']:.1%}")
```

The denominator is not the enrolment count — In R

```
cohort |>
  filter(end_of_year_status != "transferred-out") |>
  summarise(n = n(),
            promotion = mean(end_of_year_status %in% c("promoted", "completed-final-grade")),
            repetition = mean(end_of_year_status == "repeated"),
            dropout = mean(end_of_year_status == "dropped-out"))
```

The denominator is not the enrolment count

- This is the CMAM cure-rate denominator from earlier in this module, in a different sector — The convention there was to...

Repetition is a decision, not an event — In Python

```
by_district = clean.groupby("admin2")["repeating"].agg(["mean", "size"])  
print((by_district * [100, 1]).round(1))
```

Repetition is a decision, not an event — In R

```
enrolment |> filter(age_years <= 20) |>  
  summarise(repeating = mean(repeating), n = n(), .by = admin2)
```

Repetition is a decision, not an event

District	Repetition rate	n
Sud	12.3%	583
Artibonite	12.1%	618
Nord-Ouest	12.0%	625
Centre	7.4%	624

Repetition is a decision, not an event — In Python

```
centre = clean[(clean["admin2"] == "Centre") & (clean["school_year"] == 2023)]
elsewhere = clean[(clean["admin2"] != "Centre") & (clean["school_year"] == 2023)]
for name, frame in [("Centre", centre), ("elsewhere", elsewhere)]:
    over = frame["age_years"] > frame["grade"] + 5
    print(f"{name}: repetition {frame['repeating'].mean():.1%}, "
          f"over-age {over.mean():.1%}")
```

Repetition is a decision, not an event — In R

Repetition should predict over-age. Where it does not, suspect the coding.

Repetition is a decision, not an event

- A rate that is out of line with the indicator it mechanically causes is a question about the office before it is a . . .

Dropout by grade, and the grade-6 spike — In Python

```
by_grade = cohort[cohort["grade"].between(1, 6)].groupby("grade").agg(  
    n=("student_id", "size"),  
    dropout=("end_of_year_status", lambda s: s.eq("dropped-out").mean()),  
    repetition=("end_of_year_status", lambda s: s.eq("repeated").mean()),  
)  
print((by_grade * [1, 100, 100]).round(1))
```

Dropout by grade, and the grade-6 spike — In R

```
cohort |> filter(between(grade, 1, 6)) |>  
  summarise(n = n(),  
            dropout = mean(end_of_year_status == "dropped-out"),  
            repetition = mean(end_of_year_status == "repeated"), .by = grade)
```

Dropout by grade, and the grade-6 spike

Grade	n	Dropout	Repetition
1	149	8.1%	2.0%
2	271	8.5%	7.0%
3	332	13.0%	6.6%
4	231	8.7%	10.4%
5	165	8.5%	19.4%
6	110	18.2%	12.7%

Dropout by grade, and the grade-6 spike

- Repetition rises with grade and dropout does not — Grade 5 holds back 19.4% of its students while its dropout is 8.5%;...
- Grade 6 dropout is 18.2% on 110 students — That is the largest rate in the table and the smallest denominator, so...

Report the exits, then the rates — Example (cont.)

End of school year 2023, 1,292 students

Promoted	70.4%	910	
Dropped out	10.2%	132	
Repeated	9.8%	127	
Completed the final grade	5.7%	74	
Transferred out	2.9%	37	excluded from the rates below
Still enrolled at cut-off	0.9%	12	

Rates on 1,255 students, transfers excluded:

Promotion (incl. completion)	78.4%
Repetition	10.1%
Dropout	10.5%

Centre reports repetition at 7.4% against 12.0-12.3% elsewhere while holding the highest over-age share of the four districts. Treat as a

Report the exits, then the rates — Example (cont.)

recording difference pending verification, not as performance.

What comes next

- These are one year's exits.

Read the full lesson, with runnable code

`https://data-analysis.cassion.dev/courses/education-programme-analysis/
edu-05-promotion-repetition-dropout`