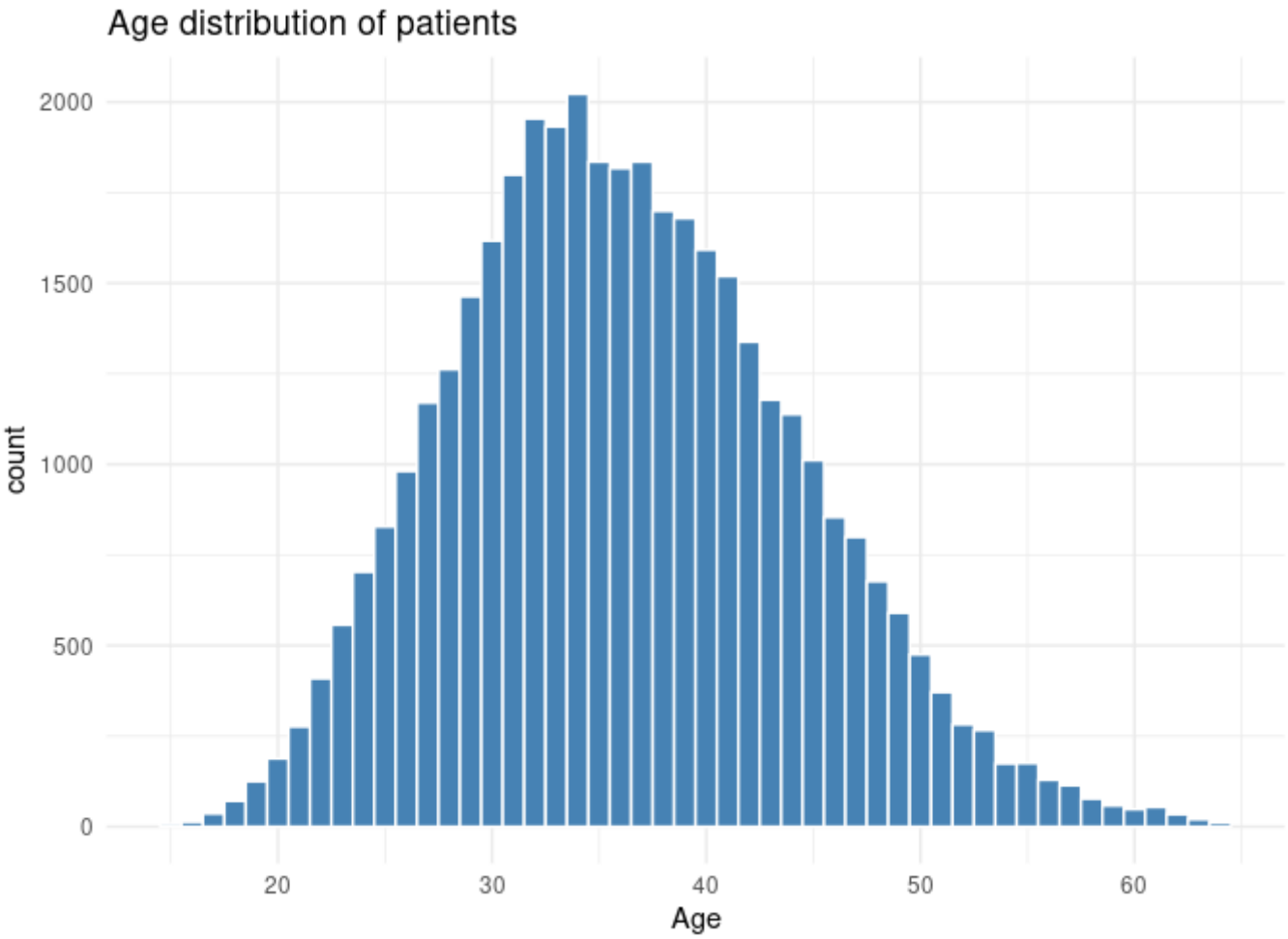
Cohort Data Report

1. Basic Overview

- Total patients: 39,263
- Total records: 723,972
- Age range: 15–64 years
- Sex distribution: Male majority (~68 %)



2. Key Findings

Treatment-interval Issues

- Zero-day gap: 23,049 records (3.2 %) have days_at_risk = 0
  These records show int_start == int_end, mostly on 1 January
  Example: Patient 24000004 has days_at_risk = 0 on 2023-01-01

IAIN	year	DOB	int_start	int_end	date_int
24000004	2022	1/1/2001	10/1/2022	12/31/2022	2022-10-01 UTC--2022-12-31
24000004	2023	1/1/2001	1/1/2023	1/1/2023	2023-01-01 UTC--2023-01-01
24000004	2023	1/1/2001	1/2/2023	3/19/2023	2023-01-02 UTC--2023-03-19

Clinical Events

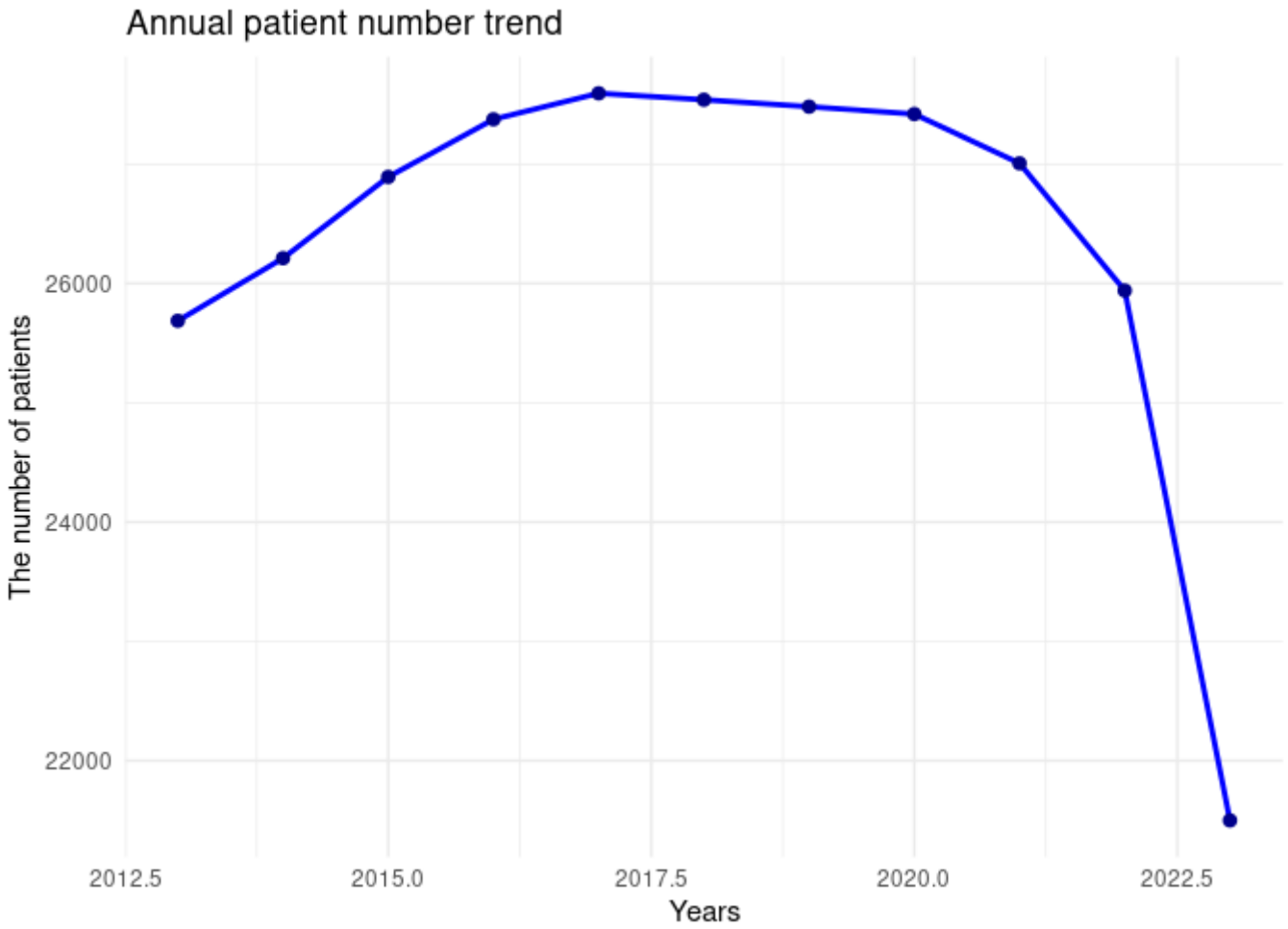
- Non-fatal overdose: 6,230 patients (15.9 %) have ⩾1 record
  - 33,033 patients have no overdose records
  - Extreme: one patient has 46 overdose records
- Death events:– All-cause mortality: 6,866 patients (17.5 %)– Drug-related mortality: 3,633 patients (9.3 %)

Drug Types

- Primary type: Opioids 93.5 %, poly-drug 6.5 %
- Treatment days:
  - Opioid cohort: mean 2,340 days (~6.4 years)
  - Poly-drug cohort: mean 2,084 days (~5.7 years)

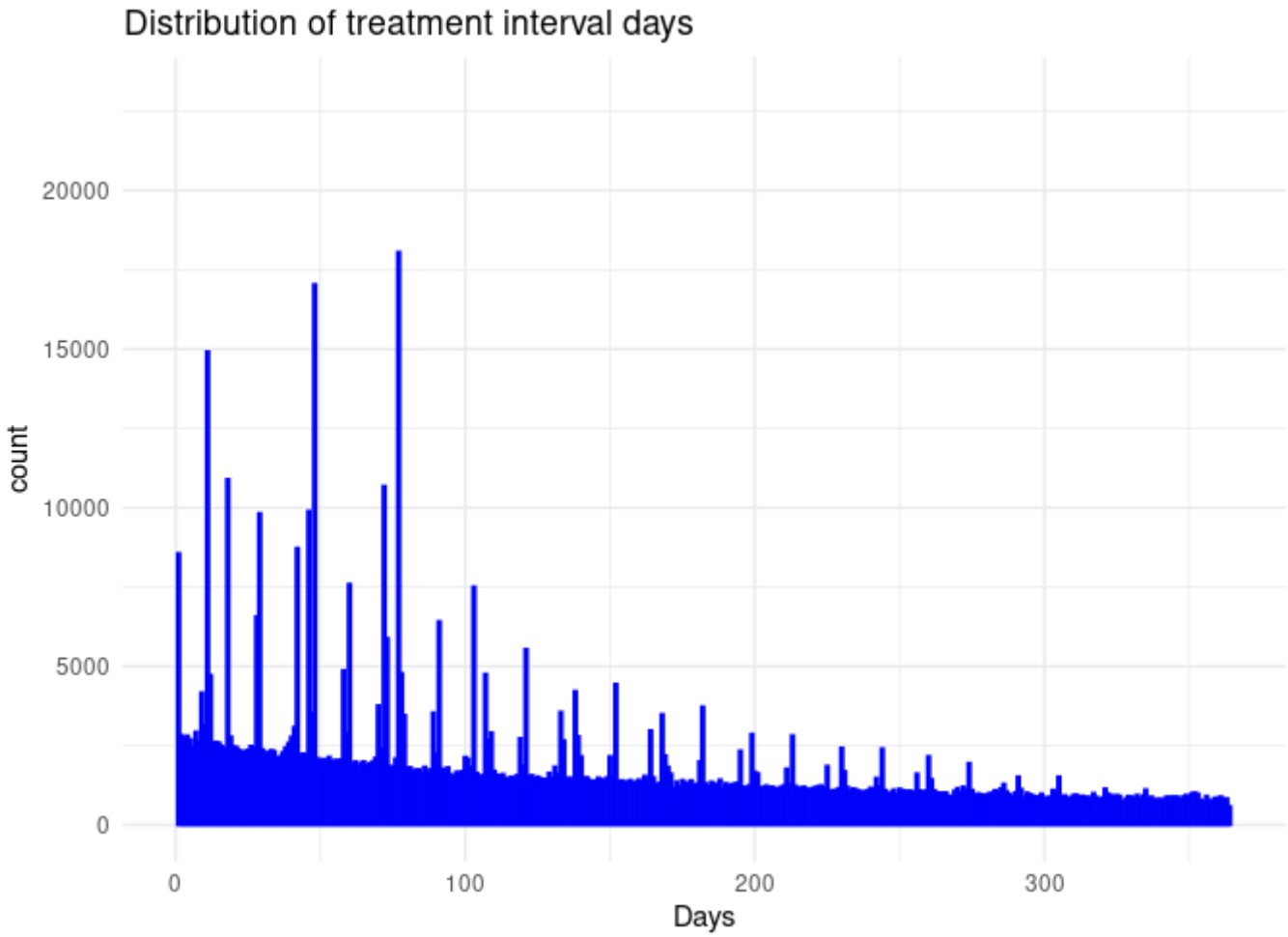
Temporal Trends

- Patient counts: increased 2013–2017, then slowly declined
- Event trends: non-fatal overdose peaked in 2019 (1,780 events), dropped sharply to 145 events in 2023



3. Data-Quality Problems

Zero-day gap problem



- Large volume of records show 0-day treatment intervals, potentially biasing time-to-event analyses
- Concentrated on 1 January each year

Root-cause Analysis

1. Annual-splitting effect
    a. Our code uses `mutate(year = year(lubridate::ymd(day))` to create a year variable
    b. When computing `days_at_risk`, the system forces a split at 1 January for intervals spanning two calendar years
    c. Example: a continuous treatment from 2022-12-15 to 2023-01-15 is split into
        i. · 2022-12-15 – 2022-12-31
        ii. · 2023-01-01 – 2023-01-01 (zero-day interval)
        iii. · 2023-01-02 – 2023-01-15

2. Time-window definition

Code block

```
episodes <- dt[, .(int_start = min(day), int_end = max(day)),
                by = .(IAIN, episode, age, sex, ..., year)]
```

3. Treatment-interval calculation

Code block

```
mutate(days_at_risk = floor(date_int %/% ddays(1)))
```

When int_start == int_end the result is naturally zero

Solution?

Code block

```

episodes <- cohort_data %>%
  group_by(IAIN) %>%
  mutate(
    merge_flag = days_at_risk == 0 & row_number() < n()
  ) %>%
  mutate(
    int_start = if_else(lag(merge_flag, default = FALSE),
                        lag(int_start), int_start)
  ) %>%
  filter(!merge_flag, !is.na(int_start)) %>%
  ungroup() %>%

  mutate(days_at_risk = as.numeric(int_end - int_start)) %>%
  mutate(
    date_int   = interval(int_start, int_end)
  )

```

Code block

```r

episodes <- cohort_data %>%
  group_by(IAIN) %>%
  mutate(
    to_merge = days_at_risk == 0 &
      (int_start == lag(int_end) + 1 | int_end == lead(int_start) - 1)
  ) %>%
  mutate(
    merge_group = cumsum(!coalesce(to_merge, TRUE))   # coalesce handles
leading NA
  ) %>%
  group_by(IAIN, merge_group) %>%
  summarise(
    int_start     = min(int_start),
    int_end       = max(int_end),
    across(c(age, sex, drug, nfod_count), first),
    across(c(nfod, acm_flag, drd_flag), max),
    days_at_risk  = as.numeric(int_end - int_start),
    year          = year(int_start),
    .groups       = "drop"
  )
```