Undoing a flattened repeat group

pivot_longer() with .value, the one call that turns a CommCare or Kobo export back into one row per person — plus the pivot_wider() warning that means your key is not a key.

Pierre Robentz Cassion

Lesson 7 of 8

R and the Tidyverse for Programme Data — Reshaping and repeating

What this lesson covers

- What a form platform does to a repeat group
- `pivot_longer()` with `.value`
- Check the row count against what you expect
- `pivot_wider()` and the warning that means something
- Separating and uniting columns
- Nesting, briefly
- Where this fits
- What comes next

What a form platform does to a repeat group — Example

hh	member_1_age	member_1_sex	member_2_age	member_2_sex
H1	34	f	8	m

`pivot_longer()` with `.value` — In R

```r
library(tidyr)

long <- wide |>
  pivot_longer(
    starts_with("member_"),
    names_to = c("member", ".value"),
    names_pattern = "member_(\\d+)_(.*)"
  )
```

`pivot_longer()` with `.value` — Example

```
# A tibble: 2 × 4
  hh    member   age sex
  <chr> <chr>  <dbl> <chr>
1 H1    1         34 f
2 H1    2          8 m
```

`pivot_longer()` with `.value`

- The pattern

pivot_longer() with .value — In R

```r
# Fixed separator, no regex needed.
pivot_longer(wide, starts_with("member_"),
             names_to = c("member", ".value"), names_sep = "_(?=[a-z]+$)")

# One value column, name kept whole.
pivot_longer(wide, starts_with("crop_"),
             names_to = "slot", names_prefix = "crop_", values_to = "crop",
             values_drop_na = TRUE)
```

Check the row count against what you expect — In R

```r
stopifnot(nrow(long) == sum(!is.na(wide$member_1_age)) + sum(!is.na(wide$member_2_age)))
```

Check the row count against what you expect — In R

```
long |>
  count(hh, name = "members_found") |>
  left_join(households |> select(hh, hh_size), by = "hh") |>
  filter(members_found != hh_size)
```

`pivot_wider()` and the warning that means something — In R

```r
by_commune_month |>
  pivot_wider(names_from = month, values_from = cases, values_fill = 0)
```

`pivot_wider()` and the warning that means something — Example

```
Warning: Values from `v` are not uniquely identified; output will contain
list-cols.
```

`pivot_wider()` and the warning that means something

- That warning means your key is not a key — The cell now holds a list of the several values that collided, and every...

`pivot_wider()` and the warning that means something — In R

```
by_commune_month |> count(commune, month) |> filter(n > 1)
```

`pivot_wider()` and the warning that means something

- `values_fill`

`pivot_wider()` and the warning that means something — In R

```
pivot_wider(..., values_fill = 0)
```

Separating and uniting columns — In R

```
tibble(x = c("2024-01", "2024-02")) |>
  separate_wider_delim(x, delim = "-", names = c("year", "month"))
```

Separating and uniting columns — Example

```
# A tibble: 2 × 2
  year  month
  <chr> <chr>
1 2024  01
2 2024  02
```

Separating and uniting columns — In R

```
separate_wider_delim(x, "-", names = c("year", "month"), too_few = "align_start")
```

Separating and uniting columns — In R

```
unite(df, "key", district, community, sep = "-", remove = FALSE)
```

Nesting, briefly — In R

```r
by_commune <- muac |>
  tidyr::nest(.by = commune)

by_commune$data[[1]]      # the full tibble for the first commune
```

Where this fits

- The Python course covers the same reshaping under `melt` and `pivot`.

What comes next

- The data has the right shape.

Read the full lesson, with runnable code

https://data-analysis.cassion.dev/courses/r-for-programme-data/r-07-pivots