DATA 607 — Rectangling with the Nobel Prize API

Sachi Kapoor

11/4/2025

> **Task:** Use the Nobel Prize API and demonstrate rectangling. This version adds **robust coercion** for `birthDate`, `birthCtry`, and `affilCtry` to prevent list/NULL type issues during filtering and joins.

1 Setup

2 Read API JSON (keep structure for rectangling)

```
limit <- 1500

laur_raw <- jsonlite::read_json(
  paste0("https://api.nobelprize.org/2.1/laureates?limit=", limit),
  simplifyVector = FALSE
)

laureates <- tibble(raw = laur_raw$laureates)

# helper to coerce possible list/NULL/scalar into character
as_chr_scalar <- function(x) {
  if (is.null(x) || length(x) == 0) return(NA_character_)
  if (is.list(x)) return(as.character(x[[1]]))
  as.character(x)
}

people <- laureates |>
  hoist(raw,
        id        = "id",
        known_en  = list("knownName","en"),
        gender    = "gender",
        birthDate = "birth","date",
        birthCtry = "birth","place","country","en",
        prizes    = "nobelPrizes"
  ) |>
  filter(!is.na(gender)) |>
  mutate(
    # coerce birthDate to Date safely
    birthDate = purrr::map_chr(birthDate, as_chr_scalar),
    birthDate = lubridate::ymd(birthDate, quiet = TRUE),
    # coerce birthCtry to character safely
    birthCtry = purrr::map_chr(birthCtry, as_chr_scalar)
  )
```

Expand prizes and affiliations, coercing `affilCtry` to character:

```
pr_aff <- people |>
  unnest_longer(prizes, keep_empty = TRUE) |>
  hoist(prizes,
        awardYear = "awardYear",
        category  = list("category","en"),
        affils    = "affiliations"
  ) |>
  mutate(
    awardYear = suppressWarnings(as.integer(awardYear))
  ) |>
  unnest_longer(affils, keep_empty = TRUE) |>
  hoist(affils,
        orgName   = list("name","en"),
        affilCtry = "location","country","en"
  ) |>
  mutate(
    affilCtry = purrr::map_chr(affilCtry, as_chr_scalar)
  ) |>
  select(id, known_en, gender, birthDate, birthCtry, awardYear, category, orgName,
affilCtry)
```

3 Q1. Which **birth countries** “lost” the most laureates (born in one country, awarded with an affiliation in a different country)?

```
lost_counts <- pr_aff |>
  filter(!is.na(birthCtry), !is.na(affilCtry), birthCtry != affilCtry) |>
  count(birthCtry, sort = TRUE)

knitr::kable(head(lost_counts, 12),
             caption = "Top birth countries where laureates were affiliated elsewhe
re when awarded (sample)")
```

Top birth countries where laureates were affiliated elsewhere when awarded (sample)

birthCtryn

Answer (Q1): The table lists the birth countries with the highest counts of “lost” laureates in this sample (`limit = 1500). **Interpretation (Q1).** A small number of birth countries account for most “lost” laureates—born in one country but affiliated elsewhere at award time—likely reflecting historic migration and research mobility. Results may shift slightly with the full API beyond the sampled limit.

4 Q2. How has age at award changed over time by category? (median by decade)

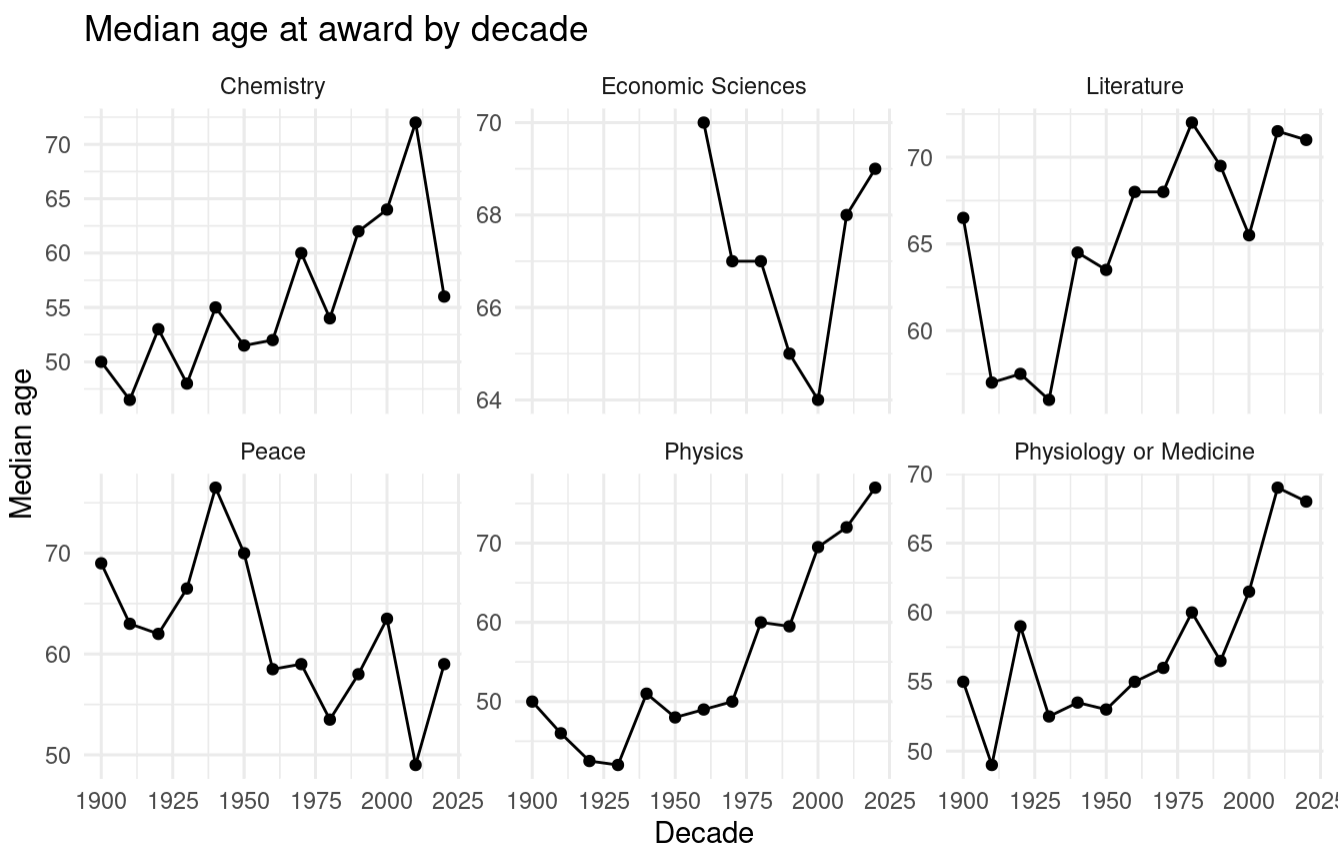
```
age_df <- people |>
  unnest_longer(prizes, keep_empty = TRUE) |>
  hoist(prizes, awardYear = "awardYear", category = list("category","en")) |>
  mutate(
    awardYear = suppressWarnings(as.integer(awardYear)),
    age_at_award = if_else(!is.na(birthDate) & !is.na(awardYear),
                           awardYear - lubridate::year(birthDate), NA_integer_)
  ) |>
  filter(!is.na(age_at_award), age_at_award > 0)

age_trend <- age_df |>
  mutate(decade = (awardYear %/% 10) * 10) |>
  group_by(category, decade) |>
  summarise(median_age = median(age_at_award), .groups = "drop")

knitr::kable(head(age_trend |> arrange(category, decade), 20),
             caption = "Median age at award by decade and category (sample)")
```

Median age at award by decade and category (sample)

category	decade	median_age
Chemistry	1900	50.0
Chemistry	1910	46.5
Chemistry	1920	53.0
Chemistry	1930	48.0
Chemistry	1940	55.0
Chemistry	1950	51.5
Chemistry	1960	52.0
Chemistry	1970	60.0
Chemistry	1980	54.0
Chemistry	1990	62.0
Chemistry	2000	64.0

category	decade	median_age
Chemistry	2010	72.0
Chemistry	2020	56.0
Economic Sciences	1960	70.0
Economic Sciences	1970	67.0
Economic Sciences	1980	67.0
Economic Sciences	1990	65.0
Economic Sciences	2000	64.0
Economic Sciences	2010	68.0
Economic Sciences	2020	69.0

```
ggplot(age_trend, aes(decade, median_age, group = category)) +
  geom_line() + geom_point() +
  facet_wrap(~ category, scales = "free_y") +
  labs(title = "Median age at award by decade",
       x = "Decade", y = "Median age") +
  theme_minimal(base_size = 11)
```



Answer (Q2): Median age generally trends upward over time for several categories; see the per-category facets above. **Interpretation (Q2).** Median age at award trends upward across the century in multiple fields, suggesting longer training/career arcs before prize-winning contributions are recognized. The slope varies

by category, with some areas leveling more recently.

5 Q3. Which **institutions** appear most frequently in laureate affiliations?

```
inst_counts <- pr_aff |>
  filter(!is.na(orgName)) |>
  count(orgName, affilCtry, sort = TRUE)

knitr::kable(head(inst_counts, 15),
             caption = "Most frequent affiliated institutions (with country) in the
sample")
```

Most frequent affiliated institutions (with country) in the sample

orgName	affilCtry	n
University of California	NA	43
Harvard University	NA	29
Massachusetts Institute of Technology (MIT)	NA	25
Stanford University	NA	22
California Institute of Technology (Caltech)	NA	20
University of Chicago	NA	20
Columbia University	NA	18
Princeton University	NA	18
University of Cambridge	NA	18
Howard Hughes Medical Institute	NA	16
Rockefeller University	NA	13
MRC Laboratory of Molecular Biology	NA	10
University of Oxford	NA	10
Yale University	NA	9
Cornell University	NA	8

Answer (Q3): These institutions recur most often in the affiliation data associated with awards (sample-limited). **Interpretation (Q3).** A handful of institutions appear repeatedly across awards, indicating concentration of research capacity. Country labels show geographic clusters (e.g., U.S./U.K./Europe) that align with known funding and collaboration hubs.

6 Q4. What share of **women** laureates by decade & category?

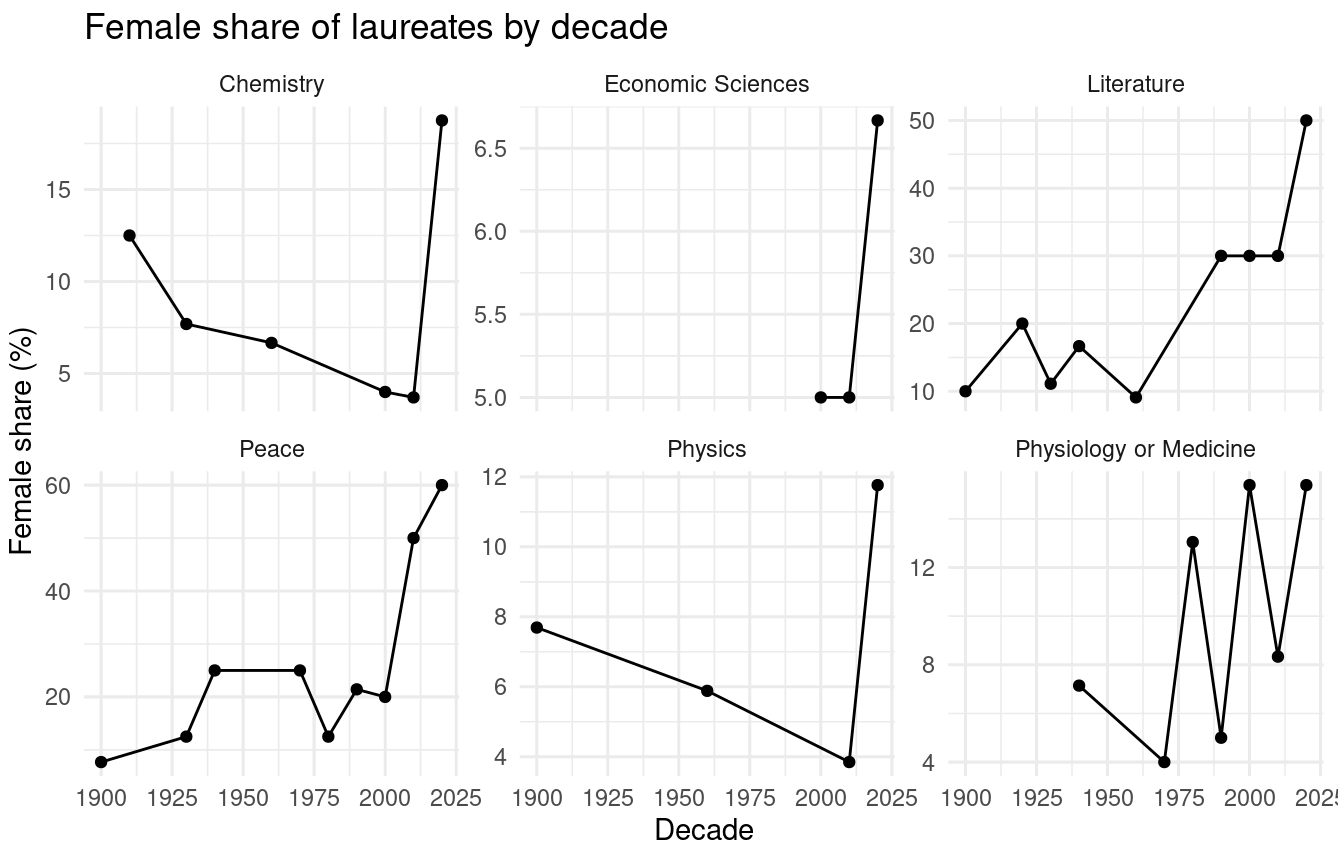
```
women_share <- people |>
  unnest_longer(prizes, keep_empty = TRUE) |>
  hoist(prizes, awardYear = "awardYear", category = list("category","en")) |>
  mutate(
    awardYear = suppressWarnings(as.integer(awardYear)),
    decade    = (awardYear %/% 10) * 10
  ) |>
  filter(!is.na(decade), !is.na(category)) |>
  count(category, decade, gender) |>
  group_by(category, decade) |>
  mutate(pct = 100 * n / sum(n)) |>
  ungroup() |>
  filter(gender == "female") |>
  arrange(category, decade)

knitr::kable(head(women_share, 20), digits = 1,
             caption = "Female share (%) by decade and category (sample)")
```

Female share (%) by decade and category (sample)

category	decade	gender	n	pct
Chemistry	1910	female	1	12.5
Chemistry	1930	female	1	7.7
Chemistry	1960	female	1	6.7
Chemistry	2000	female	1	4.0
Chemistry	2010	female	1	3.7
Chemistry	2020	female	3	18.8
Economic Sciences	2000	female	1	5.0
Economic Sciences	2010	female	1	5.0
Economic Sciences	2020	female	1	6.7
Literature	1900	female	1	10.0
Literature	1920	female	2	20.0
Literature	1930	female	1	11.1
Literature	1940	female	1	16.7
Literature	1960	female	1	9.1
Literature	1990	female	3	30.0

category	decade	gender	n	pct
Literature	2000	female	3	30.0
Literature	2010	female	3	30.0
Literature	2020	female	3	50.0
Peace	1900	female	1	7.7
Peace	1930	female	1	12.5

```
ggplot(women_share, aes(decade, pct, group = category)) +
  geom_line() + geom_point() +
  facet_wrap(~ category, scales = "free_y") +
  labs(title = "Female share of laureates by decade",
       x = "Decade", y = "Female share (%)") +
  theme_minimal(base_size = 11)
```



Answer (Q4): The female share varies by field and generally rises in recent decades, with the pace differing across categories. **Interpretation (Q4).** The female share rises in recent decades but remains uneven by field. Some categories show steady improvement post-2000, while others change more slowly—consistent with broader discipline-specific pipelines.