

The households you did not get

Fifty-four missing interviews concentrated in the stratum least like the others, two replacement clusters that change a probability nobody recomputes, and the assumption every adjustment rests on.

Pierre Robentz Cassion

Lesson 6 of 8

Survey Analysis, Sampling and Weighting — Estimating properly

What this lesson covers

- Non-response is not missing data
- The adjustment, and the assumption under it
- Adjust at the right level
- Replacement clusters
- Response rates belong in the report
- When response is bad enough to stop
- What comes next

Non-response is not missing data — In Python

```
selected = frame[frame["selected"] == True]

response = selected.groupby("stratum").agg(
    selected=("households_selected", "sum"),
    interviewed=("households_interviewed", "sum"),
)
response["rate"] = response["interviewed"] / response["selected"]
print(response)
```

Non-response is not missing data — In R

```
frame |>
  filter(selected) |>
  summarise(selected = sum(households_selected),
            interviewed = sum(households_interviewed), .by = stratum) |>
  mutate(rate = interviewed / selected)
```

Non-response is not missing data

Stratum	Selected	Interviewed	Response
Urban	350	316	90.3%
Rural accessible	350	337	96.3%
Rural remote	350	343	98.0%

The adjustment, and the assumption under it — In Python

```
survey["nr_adjustment"] = 14 / survey["ea_id"].map(
    selected.set_index("ea_id")["households_interviewed"]
)
print(survey.groupby("stratum")["nr_adjustment"].mean().round(3))
```

The adjustment, and the assumption under it — In R

```
survey |> summarise(mean_adjustment = mean(14 / households_interviewed), .by = stratum)
```

The adjustment, and the assumption under it

- The assumption is that non-respondents resemble respondents in the same area — It is not free and it is not testable. . .

The adjustment, and the assumption under it

Non-response was adjusted within enumeration area. Urban response was 90.3% against 96–98% rural. If urban non-respondents are systematically better off than urban respondents — locked compounds and daytime absence both suggest they may be — the adjusted estimate remains slightly too high.

Adjust at the right level

- **Within area.** What this survey does. Best when response varies between areas, which it always does.
- **Within stratum.** Cruder, and it throws away the information that one area had four refusals and its neighbour none.
- **Within a response class.** Group areas by something that predicts response — urban density, market day, security —. . .

Adjust at the right level — In Python

```
weights_by_area = survey.groupby("ea_id")["weight"].first().sort_values()
print(weights_by_area.tail(3))
print(f"largest weight / smallest: {weights_by_area.max() / weights_by_area.min():.1f}x")
```

Adjust at the right level — In R

```
survey |>  
  summarise(weight = first(weight), n = n(), .by = ea_id) |>  
  arrange(desc(weight)) |>  
  head(3)
```

Adjust at the right level

- Trim or flag any weight far outside the rest — A weight three times its stratum's base is one area doing the work of. . .

Replacement clusters — In Python

```
replacements = frame[frame["replacement_for"].notna()]  
print(replacements[["ea_id", "stratum", "households", "replacement_for"]])
```

Replacement clusters — In R

```
frame |> filter(!is.na(replacement_for)) |>  
  select(ea_id, stratum, households, replacement_for)
```

Replacement clusters

- Treat the replacement as the original — The pragmatic choice, and what almost everyone does
- Treat the original as non-response at the cluster level — More honest and harder: the stratum effectively has 23. . .

Replacement clusters — In Python

```
remote_areas = (frame["stratum"] == "rural-remote") & (frame["selected"] == True)
responding = int(remote_areas.sum()) - len(replacements)
print(f"25 selected, {responding} reached, cluster-level adjustment "
      f"{25 / responding:.3f}")
```

Replacement clusters — In R

```
c(selected = 25, reached = 23, adjustment = 25 / 23)
```

Replacement clusters

Report replacement even when you treat it as exchangeable. "Two of 25 rural remote clusters were replaced for inaccessibility; both replacements were accessible areas, so the rural remote estimate is likely to be conservative" is a sentence a reader needs and cannot reconstruct.

Response rates belong in the report — In Python

```
report = response.reset_index()
report["non_response_bias_risk"] = ["high", "low", "low"]
report["adjustment"] = "within enumeration area"
print(report)
```

Response rates belong in the report — In R

```
tibble::tribble(  
  ~stratum,          ~selected, ~interviewed, ~rate, ~risk,  
  "urban",           350,          316, 0.903, "high",  
  "rural-accessible", 350,          337, 0.963, "low",  
  "rural-remote",     350,          343, 0.980, "low"  
)
```

When response is bad enough to stop

- **Above 90%** — adjust and report. Residual bias is usually small relative to sampling error.
- **80 to 90%** — adjust, report, and describe the likely direction of residual bias. Do not compare against a previous. . .
- **Below 80%** — the adjustment is carrying too much. Report the estimate with a prominent caveat, or report by stratum. . .
- **Below 60% in a stratum** — that stratum's estimate is not usable and saying so is the correct finding.

What comes next

- You have an estimate, a design and an honest account of what is missing from it.

Read the full lesson, with runnable code

[https://data-analysis.cassion.dev/courses/survey-analysis-sampling/
survey-06-non-response-and-replacement](https://data-analysis.cassion.dev/courses/survey-analysis-sampling/survey-06-non-response-and-replacement)