

L'effet de plan, et celui qui est sorti sous un

La mise en grappes coûte de la précision, la stratification en rachète, et l'effet de plan est le solde. 1,60 pour l'insécurité alimentaire et 0,91 pour l'accès à l'eau, sur les mêmes 996 entretiens.

Pierre Robentz Cassion

Leçon 4 sur 8

Analyse d'enquêtes, échantillonnage et pondération — Ce que le plan achète

Ce que couvre cette leçon

- Ce que c'est
- Pourquoi les grappes vous coûtent
- Pourquoi la stratification en rachète
- Le solde, mesuré
- Celui qui est sous un
- D'où viennent les effets de plan quand vous n'en avez pas
- Rapportez-le
- La suite

Ce que c'est — Exemple

$deff = \text{variance under this design} / \text{variance under simple random sampling}$

Ce que c'est — Exemple

`effective n = n / deff`

Pourquoi les grappes vous coûtent — Exemple

```
deff_clustering = 1 + (m - 1) * ICC
```

Pourquoi les grappes vous coûtent — En Python

```
import pandas as pd

by_area = (
    survey.assign(insecure=survey["food_insecure"] == "true")
    .groupby("ea_id")["insecure"]
    .agg(["mean", "size"])
)

p = (survey["food_insecure"] == "true").mean()
m = by_area["size"].mean()
between = by_area["mean"].var(ddof=0)
icc = (between - p * (1 - p) / m) / (p * (1 - p))

print(f"mean cluster size {m:.1f}, ICC {icc:.3f}, "
      f"clustering deff {1 + (m - 1) * icc:.2f}")
```

Pourquoi les grappes vous coûtent — En R

```
by_area <- survey |>
  summarise(rate = mean(food_insecure == "true"), n = n(), .by = ea_id)

p <- mean(survey$food_insecure == "true")
m <- mean(by_area$n)
icc <- (var(by_area$rate) * (nrow(by_area) - 1) / nrow(by_area) -
      p * (1 - p) / m) / (p * (1 - p))

c(m = m, icc = icc, deff = 1 + (m - 1) * icc)
```

Pourquoi les grappes vous coûtent

- **C'est une propriété du résultat, pas de l'enquête.** La source d'eau est très regroupée — un village a un forage ou...
- **Les valeurs publiées sont la meilleure estimation disponible.** Les rapports EDS et SMART publient des effets de...

Pourquoi la stratification en rachète

- La stratification fait l'inverse.

Le solde, mesuré — En Python (suite)

```
import numpy as np

def design_effect(df, outcome, weight="weight"):
    p = np.average(df[outcome] == "true", weights=df[weight])
    total_w = df[weight].sum()

    variance = 0.0
    for _, stratum in df.groupby("stratum"):
        areas = stratum.groupby("ea_id").apply(
            lambda g: (g[weight] * ((g[outcome] == "true") - p)).sum()
        )
        n = len(areas)
        if n < 2:
            continue
        variance += n / (n - 1) * ((areas - areas.mean()) ** 2).sum()
```

Le solde, mesuré — En Python (suite)

```
variance /= total_w**2

srs = p * (1 - p) / len(df)
return p, np.sqrt(variance), variance / srs

for outcome in ["food_insecure", "improved_water_source"]:
    p, se, deff = design_effect(survey, outcome)
    print(f"{outcome:24} p={p:.1%} se={se:.4f} deff={deff:.2f} "
          f"n_eff={len(survey) / deff:.0f}")
```

```
library(survey)

design <- svydesign(ids = ~ea_id, strata = ~stratum, weights = ~weight,
                  data = survey, nest = TRUE)

svymean(~I(food_insecure == "true"), design, deff = TRUE)
svymean(~I(improved_water_source == "true"), design, deff = TRUE)
```

Le solde, mesuré

Résultat	Estimation	Effet de plan	n effectif
Insécurité alimentaire	29,1 %	1,60	623
Source d'eau améliorée	69,8 %	0,91	1 098

- « Effet de plan » n'est donc pas synonyme de « pénalité » — C'est le solde de deux forces opposées, et laquelle. . .

D'où viennent les effets de plan quand vous n'en avez pas

- **Le tour précédent de la même enquête**, même indicateur, même zone. Presque toujours disponible et presque toujours. . .
- **Un rapport d'enquête publié** pour le même indicateur dans un contexte comparable. Les rapports EDS tabulent les. . .
- **Une convention**. Les enquêtes SMART supposent couramment 1,5 pour l'anthropométrie ; les évaluations humanitaires. . .
- **1,0** n'est jamais une hypothèse défendable pour une enquête en grappes, et un protocole qui l'emploie a. . .

```
summary = pd.DataFrame({
    "outcome": ["Food insecure", "Improved water source"],
    "estimate": ["29.1%", "69.8%"],
    "ci_95": ["25.5-32.7%", "67.0-72.5%"],
    "deff": [1.60, 0.91],
    "n": [996, 996],
    "n_effective": [623, 1098],
})
```

```
tibble::tribble(  
  ~outcome,          ~estimate, ~deff, ~n_effective,  
  "Food insecure",   "29.1%",  1.60,      623,  
  "Improved water source", "69.8%", 0.91,     1098  
)
```

- Vous avez un effet de plan, donc une erreur type, donc de quoi cesser de calculer tout cela à la main.

Lire la leçon complète, avec le code exécutable

`https://data-analysis.cassion.dev/fr/cours/survey-analysis-sampling/
survey-04-design-effect`