

Deux coefficients traversent le seuil et un le retraverse

La strate rurale isolée pèse 11 % des ménages et un tiers des entretiens. Ajustez le modèle sans pondération et l'élevage ressemble à un facteur de risque significatif ; pondérez-le et il cesse de l'être, tandis que la taille du ménage et le statut de retourné se mettent à l'être.

Pierre Robentz Cassion

Leçon 7 sur 8

Régression appliquée aux données de programme — Le plan et l'ajustement

Ce que couvre cette leçon

- Le plan sous lequel l'échantillon a été tiré
- Ce que les poids font à une estimation
- Le même modèle, pondéré et non
- Poids, grappes et strates sont trois choses distinctes
- Quand laisser les poids de côté
- Rapportez-le en entier
- La suite

Le plan sous lequel l'échantillon a été tiré — En Python

```
import pandas as pd

survey = pd.read_csv("household-survey-2025.v1.csv")
frame = pd.read_csv("household-survey-frame-2025.v1.csv")

print(frame.groupby("stratum")["households"].sum())
print(survey.groupby("stratum").size())
```

Le plan sous lequel l'échantillon a été tiré — En R

```
library(dplyr)
frame |> summarise(households = sum(households), .by = stratum)
survey |> count(stratum)
```

Le plan sous lequel l'échantillon a été tiré

Strate	Ménages du sondage	Part	Entretiens	Part
Rurale accessible	28 958	51,3 %	337	33,8 %
Urbaine	21 270	37,7 %	316	31,7 %
Rurale isolée	6 200	11,0 %	343	34,4 %

Le plan sous lequel l'échantillon a été tiré

- La strate rurale isolée pèse 11,0 % de la population et 34,4 % de l'échantillon — C'est un choix de plan délibéré — le...

Ce que les poids font à une estimation — En Python

```
stratum_households = frame.groupby("stratum")["households"].sum()
base_weight = stratum_households / (25 * 14)
interviewed = frame[frame["selected"]].set_index("ea_id")["households_interviewed"]

survey["weight"] = (survey["stratum"].map(base_weight)
                    * 14 / survey["ea_id"].map(interviewed))
assert abs(survey["weight"].sum() - frame["households"].sum()) < 1

unweighted = survey["food_insecure"].mean()
weighted = np.average(survey["food_insecure"], weights=survey["weight"])
print(f"unweighted {unweighted:.1%}    weighted {weighted:.1%}")
```

Ce que les poids font à une estimation — En R

```
survey <- survey |>
  mutate(weight = base_weight[stratum] * 14 / households_interviewed)

c(unweighted = mean(survey$food_insecure),
  weighted   = weighted.mean(survey$food_insecure, survey$weight))
```

Ce que les poids font à une estimation

- 32,8 % sans pondération, 29,1 % avec — La strate rurale isolée a l'insécurité alimentaire la plus élevée (46,4 %), et. . .
- Une régression ajustée sur ce fichier sans poids hérite exactement de ce problème — et cela ne se corrige pas en. . .

Le même modèle, pondéré et non — En Python

```
import statsmodels.api as sm
import statsmodels.formula.api as smf

FORMULA = ("food_insecure ~ improved_water_source + displacement_status"
           " + main_livelihood + household_size")

plain = smf.glm(FORMULA, data=d, family=sm.families.Binomial()).fit(
    cov_type="cluster", cov_kwds={"groups": d["ea_id"]})

weighted = smf.glm(FORMULA, data=d, family=sm.families.Binomial(),
                    freq_weights=d["weight"]).fit(
    cov_type="cluster", cov_kwds={"groups": d["ea_id"]})
```

Le même modèle, pondéré et non — En R

```
library(survey)

design <- svydesign(ids = ~ea_id, strata = ~stratum, weights = ~weight,
                  data = survey, nest = TRUE)
svyglm(food_insecure ~ improved_water_source + displacement_status +
        main_livelihood + household_size, design = design,
        family = quasibinomial())
```

Le même modèle, pondéré et non

Terme	RC non pondéré	RC pondéré	Bouge ?
Source d'eau améliorée	0,75 (0,55–1,03)	0,74 (0,51–1,08)	—
Déplacement : retourné	0,56 (0,25–1,26)	0,39 (0,15–0,97)	devient significatif
Moyen d'existence : agriculture	1,55 (1,03–2,34)	1,69 (1,04–2,75)	—
Moyen d'existence : élevage	1,51 (1,04–2,18)	1,26 (0,73–2,18)	cesse d'être significatif
Moyen d'existence : salarié	0,41 (0,22–0,77)	0,45 (0,24–0,87)	—
Taille du ménage	1,04 (0,98–1,11)	1,08 (1,01–1,16)	devient significatif

- Trois coefficients changent de côté du seuil, dans les deux sens — La pondération n'est pas une correction qui. . .
- Les moyens d'existence par l'élevage sont concentrés dans la strate isolée suréchantillonnée — Sans pondération, ils. . .

Poids, grappes et strates sont trois choses distinctes

Élément du plan	Corrige	Si vous l'omettez
Poids	L'estimation	Les estimations décrivent l'échantillon, non la population
Grappes	L'erreur-type	Erreurs-types trop petites, comme à la leçon précédente
Strates	L'erreur-type	Erreurs-types un peu trop grandes — l'erreur conservatrice

Poids, grappes et strates sont trois choses distinctes

- Les poids changent le coefficient ; les grappes et les strates changent son intervalle
— C'est la même distinction. . .

Poids, grappes et strates sont trois choses distinctes — En Python

```
print(f"clusters: {d['ea_id'].nunique()} enumeration areas")
print(f"strata:    {d['stratum'].nunique()}")
print(f"weights:  {d['weight'].min():.1f} to {d['weight'].max():.1f}")
```

Poids, grappes et strates sont trois choses distinctes — En R

survey::svydesign wants all three; a glm with weights= handles only one.

Poids, grappes et strates sont trois choses distinctes

- `glm(..., weights=)` n'est pas une régression pondérée par le plan — Elle applique les poids et calcule les...

Quand laisser les poids de côté

- Une question intra-strate — « Parmi les ménages ruraux isolés, une source d'eau améliorée va-t-elle avec moins. . . »
- Un modèle dont les covariables contiennent tout ce dont les poids sont faits — Si la strate et la probabilité de. . .

Quand laisser les poids de côté — En Python

```
print(f"unweighted {plain.params['main_livelihood[T.livestock]':.3f}")  
print(f"weighted   {weighted.params['main_livelihood[T.livestock]':.3f}")
```

Quand laisser les poids de côté — En R

A large gap between the two is evidence the design is not in the model.

Quand laisser les poids de côté

- Un grand écart entre l'ajustement pondéré et le non pondéré est en soi un diagnostic — et c'est la vérification à faire. . .

Rapportez-le en entier — Exemple (suite)

Food insecurity, household survey 2025

976 households with complete covariates (of 996 interviewed),
75 enumeration areas, 3 strata.

Survey-weighted logistic regression; weights sum to 56,428 households,
which is the frame total. Standard errors account for clustering by
enumeration area and for stratification.

Weighted prevalence 29.1% (unweighted 32.8%)

Adjusted odds ratios (weighted):

Improved water source	0.74	95% CI 0.51 to 1.08
Returnee household	0.39	95% CI 0.15 to 0.97
Farming livelihood	1.69	95% CI 1.04 to 2.75
Livestock livelihood	1.26	95% CI 0.73 to 2.18
Salaried livelihood	0.45	95% CI 0.24 to 0.87

Rapportez-le en entier — Exemple (suite)

Household size, per person 1.08 95% CI 1.01 to 1.16

The unweighted model makes livestock livelihoods significant (OR 1.51, 95% CI 1.04 to 2.18) and household size not. The rural remote stratum is 11% of households and 34% of interviews, and livestock is concentrated there; the unweighted result is an artefact of the sampling design.

Odds ratios are reported because food insecurity is 29% and the ratios are moderate; the weighted risk difference for salaried livelihoods is -14.2 points against the observed mix of livelihoods, which is the number in the summary.

- Le paragraphe qui nomme l'artefact est ce qu'un relecteur vérifiera en premier — et l'écrire vous-même vaut mieux que. . .

- Chaque modèle jusqu'ici a été ajusté puis lu.

Lire la leçon complète, avec le code exécutable

[https://data-analysis.cassion.dev/fr/cours/regression-for-programmes/
reg-07-weighting-the-model](https://data-analysis.cassion.dev/fr/cours/regression-for-programmes/reg-07-weighting-the-model)