

Mille deux cents élèves, vingt-quatre décisions

La cantine a été attribuée par école, non par enfant. Testez-la sur 1 200 élèves et t vaut 5,41 ; testez-la sur les 24 écoles réellement attribuées et t vaut 3,11. L'effet a la même taille. Une seule des deux erreurs-types est honnête.

Pierre Robentz Cassion

Leçon 6 sur 8

Statistiques appliquées aux programmes — Où les comparaisons cèdent

Ce que couvre cette leçon

- Vérifiez qui a été attribué avant de vérifier qui a été mesuré
- Les deux analyses
- Pourquoi le rapport vaut 1,8
- Trois façons de bien faire
- Le même piège dans trois autres cours
- Lisez le rapport comme une affirmation sur la généralisation
- Rapportez-le en entier
- La suite

Vérifiez qui a été attribué avant de vérifier qui a été mesuré — En Python

```
import pandas as pd
import numpy as np
from scipy import stats

roster = pd.read_csv("school-roster-2024.v1.csv")
print(f"roster rows {len(roster)}, students {roster['student_id'].nunique()}")

# Two students were transferred and never de-registered, so they appear twice
# with different schools. Keep the later row: that is the school they moved to.
roster = roster.drop_duplicates("student_id", keep="last")

mixed = roster.groupby("school_id")["feeding_programme"].nunique()
print(f"schools with both fed and unfed students: {(mixed > 1).sum()}")
print(roster.groupby("school_id")["feeding_programme"].first().value_counts())
```

Vérifiez qui a été attribué avant de vérifier qui a été mesuré — En R

```
library(dplyr)
```

```
roster |> summarise(variants = n_distinct(feeding_programme), .by = school_id) |>  
  count(variants)
```

Vérifiez qui a été attribué avant de vérifier qui a été mesuré

- Pas une école sur les vingt-quatre n'a à la fois des élèves nourris et non nourris — Quinze écoles font tourner le...
- Le registre compte 1 202 lignes pour 1 200 élèves — et il faut trancher cela d'abord : deux enfants ont été transférés...
- Faites cette vérification avant chaque comparaison — Elle tient sur une ligne, et c'est la différence entre un t de...

Les deux analyses — En Python

```
attendance = pd.read_csv("school-attendance-2024.v1.csv")
MARKS = {"true": True, "Y": True, "false": False, "N": False}
marked = attendance[attendance["present"].isin(MARKS)].copy()
marked["attended"] = marked["present"].map(MARKS)

per_student = (marked.groupby("student_id")["attended"].mean()
               .rename("rate").reset_index()
               .merge(roster, on="student_id"))

fed = per_student[per_student["feeding_programme"]]["rate"]
unfed = per_student[~per_student["feeding_programme"]]["rate"]
print(stats.ttest_ind(fed, unfed, equal_var=False))
```

Les deux analyses — En R

```
per_student |>  
  t.test(rate ~ feeding_programme, data = _)
```

- Au niveau élève : 90,14 % contre 85,21 %, différence 4,93 points, $t = 5,41$ sur 1 200 élèves

Les deux analyses — En Python

```
per_school = (per_student.groupby(["school_id", "feeding_programme"])["rate"]
               .mean().reset_index())

fed_s = per_school[per_school["feeding_programme"]]["rate"]
unfed_s = per_school[~per_school["feeding_programme"]]["rate"]
result = stats.ttest_ind(fed_s, unfed_s, equal_var=False)
print(f"n = {len(fed_s)} fed, {len(unfed_s)} unfed")
print(f"difference {fed_s.mean() - unfed_s.mean():+.2%}, t = {result.statistic:.2f}")
```

Les deux analyses — En R

```
per_school |> t.test(rate ~ feeding_programme, data = _)
```

- Au niveau école : 90,25 % contre 85,04 %, différence 5,21 points, $t = 3,11$ sur 24 écoles

Les deux analyses

Analyse	n	Différence	Erreur-type	t
Niveau élève	1 200	+4,93 pts	0,91 pt	5,41
Niveau école	24	+5,21 pts	1,68 pt	3,11

- L'effet a à peine bougé. L'erreur-type a presque doublé — C'est tout le phénomène : traiter des observations en grappes. . .

Pourquoi le rapport vaut 1,8 — En Python

```
groups = [g["rate"].values for _, g in per_student.groupby("school_id")]
k, N = len(groups), len(per_student)
grand = per_student["rate"].mean()

msb = sum(len(g) * (g.mean() - grand) ** 2 for g in groups) / (k - 1)
msw = sum(((g - g.mean()) ** 2).sum() for g in groups) / (N - k)
n0 = (N - sum(len(g) ** 2 for g in groups) / N) / (k - 1)

icc = (msb - msw) / (msb + (n0 - 1) * msw)
deff = 1 + (n0 - 1) * icc
print(f"ICC {icc:.3f}, average cluster {n0:.0f}, design effect {deff:.2f}")
print(f"effective sample size {N / deff:.0f} of {N}")
```

Pourquoi le rapport vaut 1,8 — En R

lme4::lmer(rate ~ 1 + (1 | school_id)) gives the same variance components.

Pourquoi le rapport vaut 1,8

- Coefficient de corrélation intra-classe 0,060, grappe moyenne 50, effet de plan 3,94 — Six pour cent de la variation de...
- Taille d'échantillon efficace 304, non 1 200 — L'erreur-type croît comme la racine de l'effet de plan, et la racine de...

- Agréger à l'unité d'attribution — Calculez un nombre par école, testez les 24

Trois façons de bien faire — En Python

```
sizes = per_student.groupby("school_id").size().rename("students")
sized = per_school.merge(sizes, on="school_id")
fed_rows = sized[sized["feeding_programme"]]
print(f"unweighted {fed_rows['rate'].mean():.2%}, "
      f"weighted by roster {np.average(fed_rows['rate'], weights=fed_rows['students']):.2%}")
```

Trois façons de bien faire — En R

Weighting is a judgement about what the average school means, not a fix.

Trois façons de bien faire

- Employer un modèle mixte avec une constante aléatoire par école — Il garde chaque élève, estime explicitement la...
- Employer des erreurs-types robustes aux grappes — Elles gardent le modèle au niveau élève et corrigent l'erreur-type du...
- Avec 24 grappes, agrégez — Le modèle mixte apporte de la précision quand les grappes sont nombreuses et de tailles...

Le même piège dans trois autres cours

- La corrélation — La corrélation point-bisériale entre cantine et présence vaut 0,161 au niveau élève et 0,588 au niveau...

Le même piège dans trois autres cours — En Python

```
print(f"student level r = {np.corrcoef(per_student['feeding_programme'], per_student['rate'])  
      [0,1]:.3f}")  
print(f"school level r = {np.corrcoef(per_school['feeding_programme'], per_school['rate'])  
      [0,1]:.3f}")
```

Le même piège dans trois autres cours — En R

```
cor(as.numeric(per_student$feeding_programme), per_student$rate)  
cor(as.numeric(per_school$feeding_programme), per_school$rate)
```

Le même piège dans trois autres cours

- Les points d'eau — La fonctionnalité est attribuée par point et mesurée par visite
- Les cas orientés — Un travailleur social suivant quarante cas est un travailleur social, et comparer des résultats. . .
- Les mesures répétées sur le même ménage — Trois passages d'enquête sur un ménage font trois lignes et un ménage, et la. . .

Lisez le rapport comme une affirmation sur la généralisation

- Avec 24 écoles, « la cantine augmente-t-elle la présence » est répondu par quinze écoles qui en ont une contre neuf qui. . .
- Le t de 5,41 au niveau élève est une affirmation sur une étude qui n'a jamais eu lieu — C'est la réponse qu'on. . .

Rapportez-le en entier — Exemple

Attendance and school feeding, February to April 2024

Schools with feeding	90.25%	15 schools
Schools without	85.04%	9 schools

Difference +5.21 points, 95% CI 1.6 to 8.8, $t = 3.11$, Welch df 12.7

The programme is assigned by school: no school has both fed and unfed students, so the analysis has 24 independent units and not 1,200. The student-level comparison gives the same effect with $t = 5.41$; that statistic is not reported because it treats 50 children in one school as 50 independent observations (ICC 0.060, design effect 3.94, effective sample size 304).

The comparison is observational. Schools were not randomised into the programme and the difference should not be read as the effect of feeding alone.

- Le dernier paragraphe coûte deux lignes et c'est celui qu'un relecteur cherchera — Bien choisir l'unité d'analyse rend. . .

- La leçon suivante porte sur la corrélation — là où le même problème de grappes fait passer un r de 0,16 à 0,59, et où c'est la phrase écrite sous le nombre qui fait l'essentiel des dégâts.

Lire la leçon complète, avec le code exécutable

`https://data-analysis.cassion.dev/fr/cours/
applied-statistics-for-programmes/stat-06-the-unit-of-analysis`