CASSION · FORMATION

Statistiques appliquées aux programmes

Incertitude, comparaison et signification statistique, enseignées autour des affirmations que porte un rapport de programme — et non selon une séquence de tests tirée d'un manuel.

Formateur	Pierre Robentz Cassion
Niveau	Intermédiaire
Heures apprenant	24 h
Enseignée en	Python · R
Mise à jour	28 juillet 2026
Secteurs	Santé publique · Nutrition · EAH · Éducation · Protection
Normes et méthodologies	Enquête SMART · Enquête démographique et de santé (EDS) · Définitions d'indicateurs de l'UNICEF · Critères d'évaluation du CAD de l'OCDE

Consulter et exécuter la formation en ligne sur data-analysis.cassion.dev/fr/cours/applied-statistics-for-programmes

Ce que vous saurez faire

— Lire la distribution d'un indicateur avant d'y appliquer le moindre calcul, et dire ce que sa moyenne dissimule

— Poser un intervalle de confiance sur une proportion et formuler le résultat sous forme d'intervalle dans une phrase de rapport

— Comparer deux groupes avec le bon test, énoncer ses conditions d'application et les vérifier sur les données

— Distinguer la signification statistique de la portée opérationnelle, et rapporter une taille d'effet à côté de chaque valeur p

— Reconnaître quand l'unité d'analyse n'est pas la ligne, et corriger la taille d'échantillon en conséquence

— Conduire une famille de comparaisons sans fabriquer les faux positifs qu'elle invite

Prérequis

Aucun. Cette formation ne suppose aucune expérience préalable de la programmation.

Sommaire

À propos de cette formation

Le module 4 a passé six cours à renvoyer à celui-ci. Chaque fois qu'une leçon disait « posez un intervalle là-dessus avant d'écrire la phrase », elle pointait ici — et les écarts qu'elle a différés sont les exemples travaillés, déjà calculés sur des fichiers que vous avez déjà nettoyés.

Ce n'est pas un manuel de statistiques décoré de données de programme. Il est organisé autour des quatre affirmations qu'un rapport de programme porte réellement : *voici le niveau*, *ce groupe diffère de celui-là*, *ceci a changé*, et *ceci a causé cela*. Chaque unité en prend une et établit ce qu'il en coûte de la formuler honnêtement.

Trois résultats structurent le cours, tous calculés à partir de fichiers commités.

Le dénombrement moyen d'E. coli dans les ménages testés vaut 29,6 UFC/100 mL et la médiane vaut 0. Plus de la moitié des ménages testés n'ont aucune contamination détectable et la moyenne décrit une queue de distribution qui monte à 608. Un rapport citant la moyenne n'a décrit aucun ménage de l'échantillon.

Deux écarts du module 4, dont l'un est réel et l'autre non. Les garçons sont en retard d'âge à 39,2 % contre 33,8 % pour les filles — un écart de 5,5 points dont l'intervalle va de −0,4 à +11,3, si bien que la réponse honnête est que cette enquête ne peut pas trancher. Les cas avec handicap signalé aboutissent à 26,7 % contre 46,2 % — un écart de 19,4 points dont l'intervalle va de −26,1 à −12,8, qu'aucune lecture raisonnable ne rend nul.

1 200 élèves font 24 écoles. Aucune des vingt-quatre écoles n'a à la fois des élèves avec et sans cantine, si bien qu'une comparaison traitant les enfants comme des observations indépendantes a une taille d'échantillon effective de 24 et une erreur type 1,8 fois trop petite.

Python et R tout du long. Vous devriez avoir suivi les modules 3 et 4 — ce cours suppose que vous savez défendre un dénominateur, et il consacre son temps à ce qu'il faut faire une fois que vous en avez un.

Première partie

Regardez-la d'abord

CHAPITRE 1

Une moyenne de 29,6 et une médiane de zéro

Leçon 1 sur 8 · 180 min

Le dénombrement moyen d'E. coli dans les ménages testés vaut 29,6 UFC/100 mL. La médiane vaut 0. Plus de la moitié de l'échantillon n'a aucune contamination détectable, et la moyenne décrit une queue qui monte à 608.

1.1 Calculez le résumé en dernier

```python
import pandas as pd

wash = pd.read_csv("wash-household-survey-2024.v1.csv")
ecoli = wash["ecoli_cfu_100ml"].dropna()

print(ecoli.describe().round(1))
```

```r
library(dplyr)

wash |> filter(!is.na(ecoli_cfu_100ml)) |>
  summarise(n = n(), mean = mean(ecoli_cfu_100ml),
            median = median(ecoli_cfu_100ml), sd = sd(ecoli_cfu_100ml))
```

	Valeur
n	802
Moyenne	**29,6**
Médiane	**0,0**
Écart-type	85,6
Maximum	608

La moyenne vaut 29,6 et la médiane 0. Ces deux nombres décrivent les mêmes 802 ménages, et un seul des deux en décrit un.

Plus de la moitié des ménages testés n'ont aucune E. coli détectable. La moyenne est ce qu'on obtient en moyennant un grand groupe de zéros avec un petit groupe de très grands nombres, et elle atterrit dans une zone où presque personne ne se trouve.

1.2 Regardez la forme

```python
bins = [-1, 0, 10, 100, 1000]
labels = ["0 (none detected)", "1-10", "11-100", ">100"]
print(pd.cut(ecoli, bins, labels=labels).value_counts().reindex(labels))
print(f"\nskew: {ecoli.skew():.2f}")
```

```r
wash |> filter(!is.na(ecoli_cfu_100ml)) |>
  count(band = cut(ecoli_cfu_100ml, c(-1, 0, 10, 100, Inf)))
```

Bande	Ménages	Part
0 — rien de détecté	430	**53,6 %**
1–10	173	21,6 %
11–100	139	17,3 %
>100	60	7,5 %

Asymétrie +4,04. Une distribution symétrique a une asymétrie proche de zéro ; au-delà d'environ +1, la moyenne et la médiane répondent à des questions différentes.

C'est pourquoi le cours EAH rapportait des classes de risque et non une moyenne. Les classes ne sont pas un choix de présentation — ce sont le seul résumé qui survit à une distribution de cette forme.

1.3 Trois formes et ce que chacune exige

Passez les mêmes trois nombres sur quatre indicateurs de quatre cours et le motif saute aux yeux.

```
import numpy as np

def profile(series, name):
    s = series.dropna()
    return {
        "indicator": name, "n": len(s),
        "mean": round(s.mean(), 1), "median": round(s.median(), 1),
        "skew": round(s.skew(), 2),
    }

points = pd.read_csv("water-point-monitoring-2024.v1.csv")
protection = pd.read_csv("protection-referrals-2024.v1.csv")

print(pd.DataFrame([
    profile(wash["ecoli_cfu_100ml"], "E. coli, CFU/100mL"),
    profile(points["days_since_breakdown"], "days a water point is down"),
    profile(protection["days_to_first_service"], "days to first service"),
    profile(wash["litres_per_person_day"], "litres per person per day"),
]))
```

```
# One function, four indicators, four different answers about which summary
# to report.
```

Indicateur	n	Moyenne	Médiane	Asymétrie
E. coli, UFC/100 mL	802	29,6	**0,0**	+4,04
Jours d'arrêt d'un point d'eau	709	125,7	56,0	+1,64
Jours jusqu'au premier service	728	10,1	9,0	+0,52
Litres par personne et par jour	2 403	24,2	23,7	**+8,54**

Le délai jusqu'au premier service est presque symétrique — moyenne 10,1, médiane 9,0, asymétrie +0,52. Une moyenne est un résumé honnête et un écart-type signifie quelque chose.

Les jours d'arrêt d'un point d'eau sont fortement asymétriques à droite — une moyenne de 125,7 contre une médiane de 56, parce qu'une poignée de points abandonnés sont en panne depuis plus de 600 jours. Rapportez la médiane et les quartiles.

Les litres par personne et par jour paraissent presque symétriques et portent une asymétrie de +8,54. C'est cette combinaison qui est intéressante.

1.4 Quand l'asymétrie est l'erreur de données

```python
litres = wash["litres_per_person_day"].dropna()
print(f"median {litres.median():.1f}, p90 {litres.quantile(0.90):.1f}, "
      f"max {litres.max():.1f}")
print(f"above 80: {(litres > 80).sum()} households")
```

```r
wash |> summarise(median = median(litres_per_person_day),
                  p90 = quantile(litres_per_person_day, 0.9),
                  max = max(litres_per_person_day))
```

Médiane 23,7, quatre-vingt-dixième centile 33,7, maximum 277,6. **La statistique d'asymétrie ne décrit pas la population ; elle détecte onze mauvaises lignes.**

Ces onze sont les ménages dont la consommation *totale* a été saisie dans une colonne par personne — le défaut qu'enseignait le cours EAH. Il arrive ici par l'autre bout : **une asymétrie très en décalage avec l'écart interquartile est un signal de qualité des données avant d'être un constat de distribution.**

```python
clean = litres[litres <= 80]
print(f"after removing 11 rows: skew {clean.skew():.2f}, "
      f"mean {clean.mean():.1f}, median {clean.median():.1f}")
```

```r
# Recompute after the correction and see whether the shape was real.
```

Onze lignes sur 2 403 et l'asymétrie passe de +8,54 à +0,06. Moyenne 23,5, médiane 23,6 — une distribution symétrique depuis le début, portant une forme qui appartenait entièrement à une erreur d'unité.

Calculez l'asymétrie, puis décidez si c'est un constat ou un bogue. Les deux sont courants et se distinguent en regardant les valeurs extrêmes, non en regardant la statistique.

1.5 Ce qu'il faut rapporter pour chaque forme

Forme	Rapportez	Ne rapportez pas
À peu près symétrique	Moyenne et écart-type	—
Asymétrique à droite	Médiane et écart interquartile	Une moyenne sans la médiane à côté
Gonflée en zéros	**La part à zéro**, puis la distribution du reste	Une moyenne, tout court
Proportion bornée	La proportion et son intervalle	Un écart-type

E. coli est gonflée en zéros, ce qui est un cas distinct de la simple asymétrie : 53,6 % de l'échantillon se trouve exactement sur une valeur, et aucun résumé continu ne décrit cela. Le rapport en deux parties — *combien sont à zéro, et parmi les autres, à quel point* — est le seul honnête.

```
E. coli at point of collection, 802 households tested

  No detectable E. coli        53.6%   430 households
  Among the 372 with any detected:
```

```
  median                    13 CFU/100 mL
  interquartile range       5 to 49
  maximum                   608

Mean over all 802 households is 29.6 CFU/100 mL. It is not reported as a
summary because 53.6% of the sample sits at zero and the mean falls in a
range occupied by almost no household.
```

1.6 L'histogramme qu'il faut toujours tracer

```python
import matplotlib.pyplot as plt

fig, axes = plt.subplots(1, 2, figsize=(9, 3))
axes[0].hist(ecoli, bins=40)
axes[0].set_title("E. coli, raw")
axes[1].hist(np.log1p(ecoli), bins=40)
axes[1].set_title("log(1 + E. coli)")
plt.tight_layout()
```

```r
hist(wash$ecoli_cfu_100ml, breaks = 40)
hist(log1p(wash$ecoli_cfu_100ml), breaks = 40)
```

Tracez-le avant de résumer, à chaque fois, et ne le mettez pas dans le rapport. L'histogramme est un instrument pour vous, non un constat pour le lecteur — son rôle est de vous dire quel résumé est honnête, et une fois qu'il l'a fait, le résumé entre et l'histogramme reste dehors.

Deux minutes d'observation auraient évité chaque erreur de cette leçon, et c'est tout l'argument en faveur de cette habitude.

1.7 La suite

Vous savez maintenant quel nombre unique décrit un indicateur. La leçon suivante porte sur l'ampleur de son erreur possible, et sur la notation qui la transporte dans une phrase sur laquelle un lecteur peut agir.

CHAPITRE 2

Une proportion de zéro avec un intervalle jusqu'à 24 %

Leçon 2 sur 8 · 180 min

Douze enfants, aucun en retard d'âge. L'intervalle du manuel dit 0 % à 0 %. Le bon dit 0 % à 24,3 %, et l'écart entre ces deux réponses décide si vous agiriez sur cette cellule.

2.1 Pourquoi une estimation ponctuelle n'est pas un résultat

Chaque proportion du module 4 a été calculée sur un échantillon — de ménages, de cas, d'élèves. Un autre échantillon de la même population aurait produit un nombre différent, et un intervalle de confiance dit de combien.

```python
import pandas as pd
import numpy as np

wash = pd.read_csv("wash-household-survey-2024.v1.csv")
open_defecation = wash["sanitation_facility"].eq("open-defecation")
k, n = open_defecation.sum(), len(wash)
print(f"{k}/{n} = {k / n:.1%}")
```

```r
library(dplyr)
wash |> summarise(k = sum(sanitation_facility == "open-defecation"), n = n())
```

319 sur 2 403 — 13,3 %. L'intervalle là-dessus est étroit, parce que l'échantillon est grand. Sur des cellules plus petites il ne l'est pas, et tout l'intérêt de le calculer est qu'on ne peut pas savoir dans quel cas on se trouve en regardant le pourcentage.

2.2 Wilson, non Wald

La formule que la plupart des gens apprennent est l'intervalle de Wald : `p ± 1,96 × sqrt(p(1-p)/n)`. Elle est simple, c'est celle qu'un manuel montre en premier, et elle échoue précisément là où vous en avez besoin.

```python
def wald(k, n, z=1.96):
    p = k / n
    se = np.sqrt(p * (1 - p) / n)
    return p - z * se, p + z * se

def wilson(k, n, z=1.96):
    p = k / n
    denom = 1 + z**2 / n
    centre = (p + z**2 / (2 * n)) / denom
    half = z * np.sqrt(p * (1 - p) / n + z**2 / (4 * n**2)) / denom
    return centre - half, centre + half

for k, n in [(0, 12), (1, 20), (3, 11)]:
    print(f"{k}/{n} = {k/n:.1%}")
    print(f"   Wald   [{wald(k, n)[0]:.1%}, {wald(k, n)[1]:.1%}]")
    print(f"   Wilson [{wilson(k, n)[0]:.1%}, {wilson(k, n)[1]:.1%}]")
```

```r
# R has this built in and it is the Wilson interval by default.
prop.test(0, 12)$conf.int
binom.test(1, 20)$conf.int
```

Cellule	Wald	Wilson
0 sur 12	**[0,0 %, 0,0 %]**	[0,0 %, 24,3 %]
1 sur 20	**[−4,6 %, 14,6 %]**	[0,9 %, 23,6 %]
3 sur 11	[1,0 %, 53,6 %]	[9,7 %, 56,6 %]

Wald affirme qu'une cellule sans événement n'a aucune incertitude, et il produit des probabilités négatives. Les deux sont absurdes et les deux apparaissent dans de vrais rapports, parce que c'est la formule dont on se souvient.

Employez Wilson, ou `prop.test` en R et `statsmodels.stats.proportion.proportion_confint(method="wi` en Python. Cela ne coûte rien et cela ne casse pas aux bornes.

2.3 Lisez la largeur, pas seulement les bornes

```python
cases = [
    ("Open defecation", 319, 2403),
    ("Cholera case fatality", 38, 974),
    ("Cholera CFR, Nord district", 24, 381),
    ("Referral completion, disability reported", 54, 202),
    ("Over-age, grade 6", 67, 132),
]
for name, k, n in cases:
    lo, hi = wilson(k, n)
    print(f"{name:42} {k/n:6.1%}  [{lo:.1%}, {hi:.1%}]  width {hi-lo:.1%}")
```

```r
# Same five, and the width is the column to read.
```

Estimation	Valeur	Intervalle à 95 %	Largeur
Défécation à l'air libre	13,3 %	12,0–14,7 %	**2,7 pts**
Létalité du choléra	3,9 %	2,9–5,3 %	2,5 pts
Létalité, district Nord	6,3 %	4,3–9,2 %	4,9 pts
Aboutissement, handicap signalé	26,7 %	21,1–33,2 %	12,1 pts
Retard d'âge, sixième année	**50,8 %**	**42,3–59,1 %**	**16,8 pts**

Le retard d'âge en sixième vaut 50,8 % et pourrait se situer entre 42 % et 59 %. Le cours d'éducation a rapporté ce nombre à une décimale près. La décimale n'est pas fausse, elle est simplement sans objet — le second chiffre de 50,8 ne porte aucune information à n=132.

Deux choses gouvernent la largeur et une seule dépend de vous. La taille d'échantillon, décidée par le plan ; et la proximité de la proportion à 50 %, décidée par le monde. Une proportion proche de 50 % a l'intervalle le plus large possible, et les 6,3 % de Nord sont plus serrés que les 50,8 % de sixième malgré un n plus grand.

2.4 L'écrire dans une phrase

C'est ce sur quoi porte l'ossature de ce cours, et c'est un problème de rédaction autant que de statistique.

Pas ceci : « 50,8 % des élèves de sixième sont en retard d'âge. »

Ni ceci : « 50,8 % (IC à 95 % 42,3–59,1) des élèves de sixième sont en retard d'âge. » Correct, et un lecteur saute la parenthèse.

Ceci : « Entre quatre et six élèves de sixième sur dix sont en retard d'âge (50,8 %, IC à 95 % 42,3–59,1, n=132). »

Mettez l'intervalle en mots d'abord, puis donnez les nombres. Les mots sont ce que lit un non-analyste et ils portent l'incertitude ; les nombres sont ce que vérifie un analyste.

```
def sentence(label, k, n):
    lo, hi = wilson(k, n)
    return (f"{label}: {k/n:.1%} (95% CI {lo:.1%} to {hi:.1%}, n={n})")

print(sentence("Grade 6 over-age", 67, 132))
```

```
# Generate the string in code so the number and its interval cannot drift apart.
```

2.5 Quand l'intervalle change la décision

Le test de l'utilité d'un intervalle est de savoir si une valeur quelconque qu'il contient mènerait ailleurs.

```
lo, hi = wilson(38, 974)
threshold = 0.01                        # Sphere: cholera CFR below 1%
print(f"CFR {38/974:.1%}, interval [{lo:.1%}, {hi:.1%}]")
print(f"entire interval above the {threshold:.0%} threshold: {lo > threshold}")
```

```
prop.test(38, 974)$conf.int
```

La létalité du choléra vaut 3,9 % avec un intervalle de 2,9 % à 5,3 %, et le seuil Sphère vaut 1 %. Toute valeur de l'intervalle est au-dessus du seuil, si bien que la conclusion — cette réponse échoue au standard — ne dépend pas de l'endroit où se situe la vérité dans l'intervalle.

C'est là qu'on peut agir sur une estimation ponctuelle : quand tout l'intervalle dit la même chose. Là où il enjambe le seuil, le rapport honnête dit que les données ne tranchent pas, et la leçon suivante porte sur un écart où c'est exactement ce qui se produit.

2.6 Trois intervalles que ce cours ne calculera pas ainsi

Une proportion issue d'un échantillon en grappes. Le cours sur les enquêtes a établi qu'un plan en grappes élargit l'intervalle du facteur d'effet de plan, et la formule ci-dessus suppose un tirage aléatoire simple. L'appliquer à des données en grappes sous-estime la largeur, parfois d'un facteur deux.

Une médiane ou une moyenne asymétrique. Les intervalles présentés ici sont pour des proportions. Une médiane exige un bootstrap ou une méthode de rangs, et une moyenne sur la distribution d'E. coli de la leçon 1 n'a besoin ni de l'un ni de l'autre — elle a besoin d'un autre résumé.

Un effectif sans dénominateur. « 142 cas ont été signalés » n'a pas d'intervalle, car ce n'est l'estimation de rien. C'est un décompte de ce qui a été enregistré, et le cours de protection a consacré une leçon à pourquoi.

2.7 Rapportez-le comme une fourchette

```
Over-age enrolment, grade 6

  50.8%   95% CI 42.3 to 59.1, n = 132

  Between four and six students in ten. The interval is wide because grade 6
  holds 132 students; a difference of five points against another grade would
  not be distinguishable at this sample size.
```

La dernière phrase est la plus utile. Elle dit d'avance au lecteur quelles comparaisons ce nombre peut porter, ce qui l'empêche de faire celle qu'il ne peut pas.

2.8 La suite

Un intervalle dit à quel point un nombre est incertain. La leçon suivante place deux nombres côte à côte et demande si l'écart entre eux est réel — sur deux écarts laissés ouverts par le module 4, dont les réponses sont opposées.

Deuxième partie

Comparer deux groupes

CHAPITRE 3

Deux écarts, deux réponses

Leçon 3 sur 8 · 180 min

Un écart de 5,5 points avec un intervalle de −0,4 à +11,3, et un écart de 19,4 points avec un intervalle de −26,1 à −12,8. Même test, même code, conclusions opposées — et le module 4 avait laissé les deux ouverts exprès.

3.1 Les deux questions que le module 4 a reportées

Le cours d'éducation a trouvé les garçons plus souvent en retard d'âge que les filles et a refusé d'en faire un constat. Le cours de protection a trouvé un écart d'aboutissement selon le handicap et l'a qualifié de résultat le plus important du jeu de données. Les deux disaient « calculez d'abord l'intervalle ». Voici ce calcul.

```python
import pandas as pd
import numpy as np
from statsmodels.stats.proportion import proportions_ztest, confint_proportions_2indep

enrolment = pd.read_csv("school-enrolment-2024.v1.csv")
clean = enrolment[(enrolment["age_years"] <= 20)
                  & (enrolment["school_year"] == 2024)
                  & enrolment["grade"].between(1, 6)]
clean = clean.assign(over_age=clean["age_years"] > clean["grade"] + 5)

boys = clean[clean["sex"] == "m"]["over_age"]
girls = clean[clean["sex"] == "f"]["over_age"]
print(f"boys  {boys.sum()}/{len(boys)} = {boys.mean():.1%}")
print(f"girls {girls.sum()}/{len(girls)} = {girls.mean():.1%}")
```

```r
library(dplyr)

enrolment |>
  filter(age_years <= 20, school_year == 2024, between(grade, 1, 6)) |>
  mutate(over_age = age_years > grade + 5) |>
  summarise(k = sum(over_age), n = n(), .by = sex)
```

3.2 Écart numéro un : celui qui n'y est pas

Groupe	Retard d'âge	n	Intervalle à 95 %
Garçons	39,2 %	510	35,1–43,5 %
Filles	33,8 %	542	29,9–37,8 %

Les deux intervalles se recouvrent. **Un recouvrement d'intervalles est un indice et non un test** — ce qu'il faut calculer, c'est un intervalle sur la *différence*.

```python
count = np.array([boys.sum(), girls.sum()])
nobs = np.array([len(boys), len(girls)])
```

```
stat, pvalue = proportions_ztest(count, nobs)
low, high = confint_proportions_2indep(count[0], nobs[0], count[1], nobs[1])
print(f"difference {boys.mean() - girls.mean():+.1%}")
print(f"95% CI [{low:+.1%}, {high:+.1%}]")
print(f"z = {stat:.2f}, p = {pvalue:.3f}")
```

```
prop.test(c(200, 183), c(510, 542))
```

Différence +5,5 points, IC à 95 % −0,4 à +11,3, p = 0,066.

L'intervalle contient zéro. **La réponse honnête n'est donc pas « il n'y a pas d'écart » — c'est « cette enquête ne peut pas vous dire s'il y a un écart, et s'il y en a un il se situe quelque part entre des filles légèrement plus mal loties et des garçons plus mal lotis de onze points ».**

C'est une phrase réellement utile et ce n'est pas la même chose qu'un résultat nul.

3.3 Écart numéro deux : celui qui y est

```
protection = pd.read_csv("protection-referrals-2024.v1.csv")
consenting = protection[protection["consent_to_refer"]]
reached = (consenting["referral_accepted"]
           & consenting["days_to_first_service"].notna())

disability = consenting["disability_reported"].isin([True, "true", "Yes"])
k = np.array([reached[disability].sum(), reached[~disability].sum()])
n = np.array([disability.sum(), (~disability).sum()])

stat, pvalue = proportions_ztest(k, n)
low, high = confint_proportions_2indep(k[0], n[0], k[1], n[1])
print(f"{k[0]}/{n[0]} = {k[0]/n[0]:.1%}   {k[1]}/{n[1]} = {k[1]/n[1]:.1%}")
print(f"difference {k[0]/n[0] - k[1]/n[1]:+.1%}, CI [{low:+.1%}, {high:+.1%}]")
print(f"z = {stat:.2f}, p = {pvalue:.2e}")
```

```
prop.test(c(54, 663), c(202, 1436))
```

Groupe	Aboutissement	n
Handicap signalé	26,7 %	202
Non signalé	46,2 %	1 436

Différence −19,4 points, IC à 95 % −26,1 à −12,8, p < 0,001.

Toute valeur de cet intervalle est un écart substantiel. Le plus petit écart compatible avec les données vaut 12,8 points, ce qui reste assez grand pour agir. C'est ce qui en fait un constat plutôt qu'un indice.

3.4 Les deux résultats côte à côte

```
results = pd.DataFrame([
    {"comparison": "over-age, boys vs girls", "diff": 5.5,
     "ci": "-0.4 to +11.3", "n": "510 vs 542", "p": 0.066},
    {"comparison": "completion, disability", "diff": -19.4,
     "ci": "-26.1 to -12.8", "n": "202 vs 1436", "p": 0.0000002},
```

```
])
print(results)
```

```
# Two rows. The conclusion column is the analyst's, not the test's.
```

Remarquez que le résultat significatif a le plus petit échantillon dans l'un des bras. 202 cas ont produit une réponse tranchée et 510 élèves non, parce que la taille d'effet est presque quatre fois plus grande. **Taille d'échantillon et taille d'effet s'échangent**, et c'est leur combinaison qui décide si une comparaison se tranche.

3.5 Choisir le test

Trois tests couvrent presque tout ce qu'un rapport de programme compare, et le choix se fait d'après la nature du résultat plutôt que d'après ce qui paraît sophistiqué.

Comparaison	Test	Dans ce cours
Deux proportions	Test z à deux échantillons, ou `prop.test`	Les deux écarts ci-dessus
Deux moyennes	Test t de Welch	Présence et cantine, leçon 6
Une variable catégorielle contre une autre	Khi-deux	Motif de fermeture par distri

```
from scipy import stats

points = pd.read_csv("water-point-monitoring-2024.v1.csv")
table = pd.crosstab(points["admin2"], points["functional_status"])
chi2, p, dof, expected = stats.chi2_contingency(table)
print(f"chi-square {chi2:.1f}, dof {dof}, p = {p:.4f}")
print(f"smallest expected cell: {expected.min():.1f}")
```

```
chisq.test(table(points$admin2, points$functional_status))
```

3.6 Vérifiez les hypothèses, et dites que vous l'avez fait

Chaque test suppose des choses, et deux d'entre elles se vérifient en une ligne.

Indépendance. Tous les tests ci-dessus supposent que chaque ligne est une observation indépendante. C'est l'hypothèse la plus souvent violée et la plus difficile à voir — la leçon 6 porte entièrement sur un cas où elle tombe.

Effectifs attendus pour le khi-deux. Le test devient peu fiable quand une cellule attendue passe sous cinq environ. Affichez `expected.min()` à chaque fois ; s'il est petit, regroupez des catégories ou employez le test exact de Fisher.

Égalité des variances pour le test t. Ne la vérifiez pas — **employez toujours le test t de Welch**, qui ne la suppose pas. `scipy.stats.ttest_ind(..., equal_var=False)` et le comportement par défaut de `t.test` en R. La version qui suppose des variances égales n'apporte rien et tombe dès que les groupes diffèrent par leur dispersion.

```
girls_rate, boys_rate = girls.mean(), boys.mean()
print("assumptions checked:")
print(f"  independence: one row per student, no student in two rows -- verified")
print(f"  sample sizes: {len(boys)} and {len(girls)}, both well above 30")
```

```
print(f"  expected counts: smallest is {min(len(boys), len(girls)) * min(girls_rate, 1 -
    boys_rate):.0f}")


# Write the check into the script, not into your memory of having done it.
```

3.7 Rapportez la différence, non les deux nombres

```
Over-age enrolment by sex, primary, 2024

  Boys    39.2%   n = 510
  Girls   33.8%   n = 542
  Difference +5.5 points, 95% CI -0.4 to +11.3, p = 0.066

  The interval includes zero. This survey cannot establish whether over-age
  enrolment differs by sex; if it does, the gap is between 0 and 11 points
  and boys are the disadvantaged group. Not reported as a finding.

Referral completion by disability status, 1,638 consenting cases

  Disability reported     26.7%   n = 202
  Not reported            46.2%   n = 1,436
  Difference -19.4 points, 95% CI -26.1 to -12.8, p < 0.001

  The smallest gap consistent with the data is 12.8 points. Reported as a
  finding, and the pathway lesson locates it at referral-making and
  acceptance rather than at consent.
```

Les deux blocs mettent la différence et son intervalle en titre, avec les deux valeurs de groupe en dessous. Cet ordre est délibéré : la différence est l'affirmation, et les deux proportions en sont la preuve.

3.8 La suite

L'un de ces écarts est statistiquement significatif. La leçon suivante demande si c'est la même chose qu'être important — et trouve une comparaison où un p inférieur à 0,001 décrit une différence sur laquelle aucun programme n'agirait.

CHAPITRE 4

p = 0,023, et un tiers d'une journée

Leçon 4 sur 8 · 180 min

Les garçons sont présents à 88,77 % et les filles à 88,21 %. Sur 68 267 pointages, c'est significatif à p = 0,023. C'est aussi une différence d'un tiers de journée scolaire par enfant et par trimestre, sur laquelle aucun programme n'agirait et aucun ne devrait.

4.1 Un résultat significatif sur lequel personne ne devrait agir

```python
import pandas as pd
import numpy as np
from statsmodels.stats.proportion import proportions_ztest

attendance = pd.read_csv("school-attendance-2024.v1.csv")
enrolment = pd.read_csv("school-enrolment-2024.v1.csv")
current = enrolment[enrolment["school_year"] == 2024]

MARKS = {"true": True, "Y": True, "false": False, "N": False}
marked = attendance[attendance["present"].isin(MARKS)].copy()
marked["attended"] = marked["present"].map(MARKS)
marked = marked.merge(current[["student_id", "sex"]], on="student_id")

by_sex = marked.groupby("sex")["attended"].agg(["sum", "size"])
print(by_sex)

stat, p = proportions_ztest(by_sex["sum"], by_sex["size"])
print(f"z = {stat:.2f}, p = {p:.4f}")
```

```r
library(dplyr)

marked |> summarise(k = sum(attended), n = n(), .by = sex)
prop.test(c(29655, 30749), c(33408, 34859))
```

Groupe	Présences	Pointages	Taux
Garçons	29 655	33 408	**88,77 %**
Filles	30 749	34 859	**88,21 %**

Différence 0,56 point de pourcentage. z = 2,28, p = 0,023.

Selon la convention qu'emploie tout rapport, c'est une différence significative de présence selon le sexe. Elle passerait une relecture. Elle n'est aussi, opérationnellement, rien du tout.

4.2 Convertissez-la dans l'unité de la décision

```python
term_days = marked["attendance_date"].nunique()
gap = 0.0056
print(f"{term_days} school days in the term")
print(f"0.56 points = {gap * term_days:.2f} days per student per term")
```

```
# The statistic is in percentage points. The decision is in days.
```

Soixante jours de classe, donc 0,56 % vaut 0,34 jour — environ un tiers de journée par enfant et par trimestre.

Aucune intervention sur la présence n'est conçue à cette résolution. Aucun directeur d'école ne peut agir dessus. Aucune ligne budgétaire ne bouge. **Le résultat est réel, reproductible, statistiquement significatif et opérationnellement vide**, et ces quatre choses sont parfaitement compatibles.

Convertissez toujours un effet dans l'unité où se prend la décision. Les points de pourcentage sont l'unité de l'analyste ; les journées, les enfants, les cas et les francs sont celle du programme.

4.3 Pourquoi p est devenu petit : c'était le n

Le p répond à « à quel point ces données seraient surprenantes s'il n'y avait aucune différence », et la surprise croît avec la taille d'échantillon à effet constant.

```
for scale in (0.05, 0.2, 1.0):
    k = (by_sex["sum"] * scale).round().astype(int)
    n = (by_sex["size"] * scale).round().astype(int)
    stat, p = proportions_ztest(k, n)
    print(f"n = {n.sum():>6,}  difference {k[1]/n[1] - k[0]/n[0]:+.2%}  p = {p:.3f}")
```

```
# Same proportions, smaller n, and the p-value walks back across 0.05.
```

Échantillon	Différence	p
5 % des pointages (n=3 413)	+0,62 %	0,570
20 % (n=13 654)	+0,55 %	0,314
100 % (n=68 267)	**+0,56 %**	**0,023**

L'effet n'a jamais changé. Seul l'échantillon a changé. Un p est une affirmation sur la force de la preuve, non sur l'ampleur, et avec un échantillon assez grand toute différence devient significative.

Un p seul ne peut donc jamais vous dire si quelque chose compte. Il vous dit si vous pouvez écarter zéro, et zéro est rarement la comparaison intéressante.

4.4 Rapportez une taille d'effet à côté de chaque p

Trois mesures d'effet couvrent presque tout ce dont un rapport de programme a besoin.

Mesure	Pour	Se lit comme
Différence en points de pourcentage	Deux proportions	La réponse directe
Rapport de risques	Deux proportions, événements rares	Combien de fois plus probable
d de Cohen	Deux moyennes	Écarts-types de séparation

```
p_boys, p_girls = by_sex["sum"] / by_sex["size"]
print(f"difference: {p_boys - p_girls:+.2%} points")
print(f"risk ratio: {p_boys / p_girls:.3f}")
```

```
# Two lines. There is no excuse for a p-value with no effect size beside it.
```

Rapport de risques 1,006. Les garçons sont présents 0,6 % plus souvent en termes relatifs — une autre façon de dire le même rien.

Comparez à l'écart selon le handicap de la leçon précédente : une différence de 19,4 points et un rapport de risques de 0,58, c'est-à-dire que les cas signalant un handicap aboutissent à un peu plus de la moitié du taux. **Les deux mêmes nombres, calculés de la même façon, décrivent un constat dans un cas et du bruit dans l'autre**, et seule la taille d'effet les distingue.

4.5 Fixez le seuil avant de tester

La défense contre les deux erreurs — courir après un rien significatif, et écarter un quelque chose non significatif — consiste à décider d'avance quelle ampleur de différence changerait ce que vous faites.

```
MEANINGFUL = 0.03      # 3 points of attendance, about 1.8 days a term

observed = p_boys - p_girls
print(f"observed {observed:+.2%}, threshold {MEANINGFUL:.0%}")
print(f"large enough to act on: {abs(observed) >= MEANINGFUL}")
```

```
# Write the threshold in the analysis plan, not after seeing the result.
```

Trois points de présence font environ 1,8 jour par trimestre, c'est-à-dire l'échelle à laquelle un programme de suivi serait conçu. Les 0,56 observés en valent le cinquième.

Écrire `MEANINGFUL` avant de lancer le test est ce qui empêche le seuil de se déplacer là où le résultat est tombé. C'est la même discipline que le tableau des facteurs de confusion du cours d'épidémiologie : **décidez la règle avant de voir le nombre auquel elle s'appliquera.**

4.6 Les quatre cas

```
cases = pd.DataFrame([
    ("significant, large",     "report it",                    "disability gap, -19.4 pts"
    ),
    ("significant, small",     "report the effect size, say it is small", "attendance by
    sex, 0.56 pts"),
    ("not significant, large", "report the interval; underpowered", "over-age by sex, +5.5
     pts"),
    ("not significant, small", "report as no evidence of a difference", "-"),
], columns=["case", "what to do", "example in this course"])
print(cases)
```

```
# Four quadrants, and only one of them is "publish the p-value".
```

La troisième ligne est celle qu'on traite mal. L'écart de retard d'âge selon le sexe valait 5,5 points avec p = 0,066, et l'instinct est de rapporter « pas de différence ». Le rapport correct dit que l'étude n'a pas pu trancher une différence d'une ampleur qui aurait compté — ce qui est une affirmation sur l'échantillon, et un argument pour en prendre un plus grand plutôt que pour fermer la question.

4.7 Rapportez-le en entier

```
Attendance by sex, February to April 2024

  Boys    88.77%   29,655 of 33,408 marks
  Girls   88.21%   30,749 of 34,859 marks

  Difference +0.56 percentage points (p = 0.023, risk ratio 1.006).

  Equivalent to 0.34 school days per child per term. Below the 3-point
  threshold set in the analysis plan and not reported as a programme
  finding. The p-value is small because the analysis has 68,267 marks, not
  because the difference is large.
```

La dernière phrase est ce qui rend le bloc honnête. Un lecteur qui ne voit que « p = 0,023 » supposera que la différence compte, et la phrase qui dit pourquoi elle ne compte pas tient sur une ligne.

4.8 La suite

Chaque test jusqu'ici portait sur une comparaison. La leçon suivante en lance vingt-trois d'un coup, sur un effet qui n'existe pas, et compte combien reviennent « significatives ».

Troisième partie

Où les comparaisons cèdent

CHAPITRE 5

Vingt-quatre tests, deux constats, aucun effet

Leçon 5 sur 8 · 180 min

Le registre d'inscription tire le sexe indépendamment de tout le reste, si bien que l'écart de retard d'âge selon le sexe vaut exactement zéro par construction. Testez-le école par école et deux des vingt-quatre ressortent significatives, ce qui est à peu près ce que le hasard vous devait.

5.1 Une comparaison dont on connaît la réponse

Les deux leçons précédentes testaient une comparaison à la fois. Une vraie analyse s'arrête rarement à une seule : un relecteur demande le même indicateur par district, par année, par école, par statut de déplacement, et l'analyste lance chacun.

Cette leçon en lance vingt-quatre sur une comparaison dont la vraie réponse est connue, parce que le jeu de données a été produit avec le sexe tiré indépendamment de l'âge, de l'année, du redoublement et de l'école. **La vraie différence de retard d'âge selon le sexe vaut exactement zéro**, et tout ce qu'un test y trouve est un artefact d'échantillonnage.

```python
import pandas as pd
import numpy as np
from statsmodels.stats.proportion import proportions_ztest

enrolment = pd.read_csv("school-enrolment-2024.v1.csv")
clean = enrolment[(enrolment["school_year"] == 2024)
                  & (enrolment["age_years"] <= 20)
                  & enrolment["grade"].between(1, 6)]
clean = clean.assign(over_age=clean["age_years"] > clean["grade"] + 5)

results = []
for school, group in clean.groupby("school_id"):
    counts = group.groupby("sex")["over_age"].agg(["sum", "size"])
    if counts["size"].min() < 5:
        continue
    stat, p = proportions_ztest(counts["sum"], counts["size"])
    results.append({"school": school, "p": p,
                    "boys": counts.loc["m", "sum"] / counts.loc["m", "size"],
                    "girls": counts.loc["f", "sum"] / counts.loc["f", "size"]})

tests = pd.DataFrame(results).sort_values("p")
print(f"{len(tests)} schools tested, {(tests['p'] < 0.05).sum()} significant")
```

```r
library(dplyr)

clean |>
  summarise(k = sum(over_age), n = n(), .by = c(school_id, sex)) |>
  tidyr::pivot_wider(names_from = sex, values_from = c(k, n)) |>
  filter(n_m >= 5, n_f >= 5) |>
  rowwise() |>
  mutate(p = prop.test(c(k_m, k_f), c(n_m, n_f))$p.value)
```

École	Garçons	Filles	p
SCH23	62,5 % (10/16)	25,9 % (7/27)	**0,018**
SCH06	39,1 % (9/23)	12,5 % (3/24)	**0,036**
SCH21	43,5 % (10/23)	25,8 % (8/31)	0,173
SCH03	50,0 % (11/22)	30,8 % (8/26)	0,175
SCH04	27,3 % (3/11)	55,6 % (5/9)	0,199

Vingt-quatre écoles testées, deux significatives à p < 0,05. Les deux seraient écrites. SCH23 en particulier paraît frappante — des garçons en retard d'âge à plus du double du taux des filles, dans une école de 43 enfants.

Aucune des deux n'est réelle. Le générateur n'a jamais laissé le sexe toucher quoi que ce soit.

5.2 Deux, c'est ce que le hasard vous devait

```python
n_tests = len(tests)
print(f"tests run: {n_tests}")
print(f"expected significant if nothing is real: {0.05 * n_tests:.1f}")
print(f"chance of at least one: {1 - 0.95 ** n_tests:.1%}")
```

```r
# 1 - 0.95^24. The arithmetic is one line and it is the whole lesson.
```

Faux positifs attendus : 1,2. Observés : 2. La probabilité d'obtenir *au moins un* résultat significatif à partir de vingt-quatre hypothèses nulles vraies vaut **70,8 %** — le résultat surprenant aurait donc été de n'en trouver aucun.

Un seuil de 0,05 est une promesse portant sur un test. Il dit que si rien ne se passe, vous serez trompé une fois sur vingt. Lancez vingt tests et être trompé une fois est le cas attendu, non l'accident.

C'est pourquoi « nous avons regardé par école et deux écoles se détachent » n'est pas un constat. C'est une description de ce que font vingt-quatre tests.

5.3 Bonferroni : divisez le seuil par le nombre de tests

```python
BONFERRONI = 0.05 / n_tests
print(f"corrected threshold: {BONFERRONI:.5f}")
print(f"surviving: {(tests['p'] < BONFERRONI).sum()}")
```

```r
p.adjust(tests$p, method = "bonferroni") < 0.05
```

Seuil 0,05/24 = 0,00208. Survivants : aucun. Le plus petit p des vingt-quatre vaut 0,018, un ordre de grandeur plus loin.

La correction est grossière — elle suppose le pire cas, que tous les tests sont indépendants et toutes les hypothèses nulles vraies — et elle est grossière du bon côté. **C'est le bon choix par défaut quand vous n'avez pas de raison d'en préférer un autre**, et elle tient sur une ligne.

5.4 Benjamini-Hochberg, quand vingt-quatre devient deux cents

Bonferroni contrôle la probabilité d'*un seul* faux positif, ce qui est strict. Quand une analyse de dépistage lance des centaines de tests et s'attend à quelques effets réels parmi eux, contrôler la *proportion* de fausses découvertes est la garantie la plus utile.

```python
from statsmodels.stats.multitest import multipletests

reject, adjusted, _, _ = multipletests(tests["p"], alpha=0.05, method="fdr_bh")
print(f"BH survivors: {reject.sum()}")
```

```r
p.adjust(tests$p, method = "BH") < 0.05
```

Benjamini-Hochberg ne retient lui non plus aucune des vingt-quatre, ce qui est la bonne réponse ici — il n'y a rien à trouver. Son avantage apparaît quand il y a quelque chose : sur deux cents tests dont vingt effets réels, Bonferroni en manquera la plupart et BH non.

Employez Bonferroni pour une petite famille confirmatoire ; BH pour un large dépistage exploratoire. Les deux tiennent sur une ligne, et le choix importe beaucoup moins que le fait d'en faire un.

5.5 La correction ne détruit pas les constats réels

La crainte qui empêche de corriger est que cela enterre quelque chose de vrai. Testez un effet réel de la même façon et regardez ce qui se produit.

```python
previous = enrolment[enrolment["school_year"] == 2023].set_index("student_id")
clean = clean.assign(last_year=clean["student_id"].map(previous["end_of_year_status"]))
returning = clean[clean["last_year"].isin(["repeated", "promoted"])]

survivors = []
for school, group in returning.groupby("school_id"):
    counts = group.groupby("last_year")["over_age"].agg(["sum", "size"])
    if len(counts) < 2 or counts["size"].min() < 5:
        continue
    stat, p = proportions_ztest(counts["sum"], counts["size"])
    survivors.append(p)

survivors = np.array(survivors)
print(f"{len(survivors)} schools, {(survivors < 0.05).sum()} significant, "
      f"{(survivors < 0.05 / len(survivors)).sum()} survive Bonferroni")
```

```r
# Same loop, different comparison. The generator made this one real.
```

Comparaison	Tests	Significatifs à 0,05	Survivent à Bonferroni
Retard d'âge selon le **sexe** (aucun effet réel)	24	2	**0**
Retard d'âge selon le **redoublement de 2023** (réel)	14	14	**9**

L'écart de 94,5 % contre 30,6 % du cours d'éducation survit à la correction dans neuf des quatorze écoles où il peut être testé. Un écart qui n'existe pas ne survit nulle part. La correction n'est pas un impôt sur les constats ; c'est ce qui les sépare des deux écoles qui allaient de toute façon paraître intéressantes.

5.6 Comptez les tests, y compris ceux que vous n'avez pas rapportés

Le nombre qui entre dans la correction est le nombre de tests que vous avez *lancés*, non celui que vous avez *écrit*. Quatre sortes de tests échappent régulièrement au compte :

Les sous-groupes regardés puis abandonnés. Découper par district, ne rien voir, et découper par année ensuite fait deux familles de tests, non une.

Les résultats essayés tour à tour. Tester le retard d'âge, puis la présence, puis l'achèvement contre le même regroupement fait trois tests.

Les seuils déplacés. « Retard d'âge » à année + 2 plutôt qu'à année + 1 est un second test de la même idée.

L'analyse que quelqu'un d'autre a lancée sur les mêmes données. Rarement connue, et à demander quand un relecteur arrive avec un résultat frappant sur un sous-groupe.

```
plan = {
    "outcomes": ["over_age"],
    "groupings": ["sex"],
    "subgroups": ["school_id"],
}
n_planned = 1 * 1 * 24
print(f"tests declared in the analysis plan: {n_planned}")


# Write the number down before running anything. It cannot be recovered after.
```

Déclarez la famille avant de la lancer. Après coup, le nombre de tests est affaire de mémoire et d'intérêt personnel, et les deux le réduisent.

5.7 Rapportez-le en entier

```
Over-age enrolment by sex, school-level analysis

  24 schools tested (both sexes with at least 5 students in grades 1-6).
  2 schools significant at p < 0.05: SCH23 (p = 0.018), SCH06 (p = 0.036).

  With 24 tests, 1.2 significant results are expected by chance alone and
  the probability of at least one is 71%. Neither school survives the
  Bonferroni threshold of 0.0021, and neither is reported as a finding.

  For comparison, over-age by 2023 repetition is significant in all 14
  schools testable and survives Bonferroni in 9 of them.
```

La ligne de comparaison est ce qui rend le bloc convaincant plutôt que défensif. Elle montre la même procédure trouvant quelque chose quand il y a quelque chose à trouver, ce qu'un responsable de programme a besoin de voir avant d'accepter que deux écoles frappantes soient du bruit.

5.8 La suite

Tous les tests jusqu'ici ont supposé qu'une ligne était une observation indépendante. La leçon suivante prend une comparaison où cette hypothèse tombe — une cantine scolaire attribuée par école, testée sur des élèves — et trouve une statistique t presque deux fois plus grande que l'honnête.

CHAPITRE 6

Mille deux cents élèves, vingt-quatre décisions

Leçon 6 sur 8 · 180 min

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.

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

```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())
```

```r
library(dplyr)

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

Pas une école sur les vingt-quatre n'a à la fois des élèves nourris et non nourris. Quinze écoles font tourner le programme et neuf non, et chaque enfant d'une école partage son statut.

Le registre compte 1 202 lignes pour 1 200 élèves, et il faut trancher cela d'abord : deux enfants ont été transférés sans être radiés, si bien que chacun figure sous les deux écoles avec des statuts de cantine opposés. Joindre avant de dédoublonner leur donne deux lignes de présence chacun et place le même enfant des deux côtés de la comparaison.

Cette seule ligne décide de toute l'analyse. **Le programme a été attribué à 24 écoles, donc l'analyse a 24 observations indépendantes et non 1 200** — et les 1 200 élèves sont 24 groupes d'une cinquantaine d'enfants qui partagent un directeur, un bassin de recrutement, une route et un calendrier scolaire.

Faites cette vérification avant chaque comparaison. Elle tient sur une ligne, et c'est la différence entre un t de 5,41 et un t de 3,11.

6.2 Les deux analyses

```python
attendance = pd.read_csv("school-attendance-2024.v1.csv")
```

```python
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))
```

```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.

```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}")
```

```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.

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 comme indépendantes ne biaise pas l'estimation, cela sous-estime son incertitude, et chaque p et chaque intervalle bâtis dessus sont trop étroits.

6.3 Pourquoi le rapport vaut 1,8

Le multiplicateur n'est pas arbitraire — il vient de la part de variation qui se situe *entre* les écoles plutôt qu'à l'intérieur.

```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}")
```

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

Coefficient de corrélation intra-classe 0,060, grappe moyenne 50, effet de plan 3,94. Six pour cent de la variation de présence se situe entre écoles, ce qui paraît négligeable et ne l'est pas : avec cinquante enfants par école, cela multiplie la variance par près de quatre.

Taille d'échantillon efficace 304, non 1 200. L'erreur-type croît comme la racine de l'effet de plan, et la racine de 3,94 vaut 1,99 — ce qui donne le 1,8 observé plus haut.

C'est le même effet de plan que le cours sur les enquêtes appliquait à l'échantillonnage en grappes. **L'arithmétique se moque de savoir si les grappes viennent d'un plan de sondage ou de la façon dont un programme a été déployé** — une cantine attribuée par école est un plan en grappes, que quelqu'un l'ait appelé ainsi ou non.

6.4 Trois façons de bien faire

Agréger à l'unité d'attribution. Calculez un nombre par école, testez les 24. Simple, transparent, et c'est ce que fait le tableau ci-dessus. Le coût est qu'une école de 90 enfants compte autant qu'une de 20.

```
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%}
     ")
```

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

Employer un modèle mixte avec une constante aléatoire par école. Il garde chaque élève, estime explicitement la variance inter-écoles et gère des tailles d'école inégales. `statsmodels.formula.api.mixedlm` en Python, `lme4::lmer` en R.

Employer des erreurs-types robustes aux grappes. Elles gardent le modèle au niveau élève et corrigent l'erreur-type du regroupement, ce qui est l'approche standard en économétrie et tient dans un argument de `statsmodels`. Il leur faut un nombre raisonnable de grappes — 24 est bas, et en dessous d'une trentaine la correction devient elle-même peu fiable.

Avec 24 grappes, agrégez. Le modèle mixte apporte de la précision quand les grappes sont nombreuses et de tailles inégales ; ici il produirait une réponse voisine avec plus de machinerie et plus d'hypothèses.

6.5 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 école.** Les deux sont des réponses correctes à des questions différentes, et rapporter le chiffre au niveau élève comme « la corrélation entre cantine et présence » attribue à des enfants une relation qui est celle des écoles.

```
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}")
```

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

Les points d'eau. La fonctionnalité est attribuée par point et mesurée par visite. Le cours EAH comptait un point une fois et non ses douze visites, ce qui est cette règle appliquée avant qu'elle ait un nom.

Les cas orientés. Un travailleur social suivant quarante cas est un travailleur social, et comparer des résultats entre travailleurs sur 1 600 cas a pour unité le nombre de travailleurs, quel qu'il soit.

Les mesures répétées sur le même ménage. Trois passages d'enquête sur un ménage font trois lignes et un ménage, et la même correction s'applique.

6.6 Lisez le rapport comme une affirmation sur la généralisation

Il y a une raison pour que l'analyse honnête paraisse plus faible, et ce n'est pas une subtilité technique.

Avec 24 écoles, « la cantine augmente-t-elle la présence » est répondu par quinze écoles qui en ont une contre neuf qui n'en ont pas. Tout ce qui diffère entre ces deux ensembles d'écoles — où elles sont, qui les dirige, pourquoi elles ont été choisies pour le programme — voyage avec la comparaison, et 1 200 élèves n'y changent rien.

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 obtiendrait si 1 200 enfants avaient chacun été affectés indépendamment à des repas scolaires, ce qui aurait appris bien davantage et aurait été un autre programme.

6.7 Rapportez-le en entier

```
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 l'intervalle honnête ; cela ne rend pas la comparaison causale, et le cours d'épidémiologie a établi ce qui le fait.

6.8 La suite

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.

Quatrième partie

Ce que le rapport affirme

CHAPITRE 7

r = 0,039, et tout le monde attendait 0,5

Leçon 7 sur 8 · 180 min

La présence et la littératie de fin d'année corrèlent à 0,039 sur 659 élèves — un intervalle de −0,04 à +0,12, ce qui est aussi proche de rien qu'un jeu de données peut l'être. La corrélation que tout le monde suppose présente dans le registre n'y est pas, et le rapporter est le constat.

7.1 La corrélation qui n'y est pas

```python
import pandas as pd
import numpy as np

assessment = pd.read_csv("learning-assessment-2024.v1.csv")
attendance = pd.read_csv("school-attendance-2024.v1.csv")
roster = pd.read_csv("school-roster-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)
rate = marked.groupby("student_id")["attended"].mean().rename("attendance")

endline = (assessment[assessment["assessment_round"] == "endline"]
           .pivot_table(index="student_id", columns="domain", values="raw_score"))
joined = endline.join(rate, how="inner").join(roster.set_index("student_id"))

print(joined[["literacy", "numeracy", "attendance"]].corr().round(3))
```

```r
library(dplyr)

joined |> select(literacy, numeracy, attendance) |> cor(use = "complete.obs")
```

Paire	r	n
Présence et littératie de fin d'année	**0,039**	659
Présence et numératie de fin d'année	**−0,006**	659
Littératie et numératie	0,849	659

La présence et la littératie corrèlent à 0,039. Chaque document de programme éducatif suppose cette relation et la plupart en citent un chiffre. Dans ce registre elle est absente.

C'est un résultat, et il lui faut un intervalle avant d'en être un.

7.2 Un intervalle sur une corrélation

```python
def fisher_ci(r, n, z=1.96):
    zr = np.arctanh(r)
```

```
    se = 1 / np.sqrt(n - 3)
    return np.tanh(zr - z * se), np.tanh(zr + z * se)

for label, r, n in [("attendance vs literacy", 0.039, 659),
                    ("literacy vs numeracy", 0.849, 659)]:
    lo, hi = fisher_ci(r, n)
    print(f"{label:24} r = {r:+.3f}  95% CI [{lo:+.3f}, {hi:+.3f}]")
```

```
cor.test(joined$attendance, joined$literacy)
```

r = 0,039, IC à 95 % −0,037 à +0,115. L'intervalle contient zéro et toute valeur qu'il contient est négligeable ; ce n'est donc pas un résultat nul par manque de puissance — 659 élèves suffisent à dire que si une relation existe, elle est trop petite pour compter.

Comparez le même intervalle sur la seconde paire : **r = 0,849, IC 0,826 à 0,869**, où l'intervalle entier est grand.

Une corrélation sans intervalle est le même défaut qu'une proportion sans le sien, et `cor.test` le donne gratuitement.

7.3 La corrélation qui y est et ne dit rien

La littératie et la numératie corrèlent à 0,849, ce qui est le plus grand r de ce cours et le moins intéressant.

C'est un enfant, un jour d'épreuve, deux copies de quarante items. Un enfant malade, à jeun, incapable de lire les consignes, ou simplement bon élève, obtient des scores voisins aux deux. La corrélation est une propriété de la mesure, non une découverte sur la littératie et la numératie.

Demandez ce qu'il faudrait pour que la corrélation soit nulle. Ici la réponse est « le même enfant devrait réussir indépendamment deux copies passées à une demi-heure d'intervalle », ce que personne ne croit — donc un r élevé ne porte aucune information.

Une corrélation n'est informative que si son absence était plausible, et c'est un jugement sur le monde plutôt que sur les données.

7.4 Élevez-la au carré avant de la décrire

```
for label, r in [("attendance vs literacy", 0.039),
                 ("attendance vs literacy, school level", 0.199),
                 ("literacy vs numeracy", 0.849)]:
    print(f"{label:38} r = {r:+.3f}   r-squared = {r**2:.3f}")
```

```
# r^2 is the share of variance, and it is much smaller than r looks.
```

Paire	r	r^2	Se lit
Présence et littératie	0,039	**0,002**	Deux dixièmes de un pour cent
Présence et littératie, niveau école	0,199	0,040	Quatre pour cent
Littératie et numératie	0,849	0,721	Soixante-douze pour cent

Un r de 0,199 sonne comme une relation et explique quatre pour cent de la variation. Élever au carré est la défense la moins chère contre la surlecture d'une corrélation moyenne, et le nombre à mettre dans un rapport est généralement r^2 plutôt que r.

7.5 Pearson, Spearman, et onze mauvaises lignes

La leçon 1 a trouvé onze ménages dont la consommation *totale* avait été saisie dans une colonne par personne. Regardez ce que ces onze lignes font à une corrélation.

```python
wash = pd.read_csv("wash-household-survey-2024.v1.csv")
pair = wash.dropna(subset=["round_trip_minutes", "litres_per_person_day"])
clean = pair[pair["litres_per_person_day"] <= 80]

for label, frame in [("all rows", pair), ("11 rows removed", clean)]:
    p = frame["round_trip_minutes"].corr(frame["litres_per_person_day"])
    s = frame["round_trip_minutes"].corr(frame["litres_per_person_day"],
                                         method="spearman")
    print(f"{label:18} n = {len(frame):5}  Pearson {p:+.3f}  Spearman {s:+.3f}")
```

```r
cor(pair$round_trip_minutes, pair$litres_per_person_day)                    # Pearson
cor(pair$round_trip_minutes, pair$litres_per_person_day, method = "spearman")
```

Lignes	n	Pearson	Spearman
Toutes	2 403	**−0,185**	−0,285
Onze retirées	2 392	**−0,293**	−0,287

Onze lignes sur 2 403 amputent Pearson de plus d'un tiers. Spearman n'a pas bougé.

Pearson emploie les valeurs, si bien qu'un ménage enregistré à 277 litres par personne tire fort sur le coefficient. Spearman emploie les rangs, si bien qu'une mauvaise valeur qui reste la plus grande ne change rien.

Rapportez Spearman quand l'une des deux variables est asymétrique ou porte des valeurs extrêmes que vous n'avez pas résolues, et Pearson quand les deux sont à peu près symétriques et propres. Un grand écart entre les deux est un signal pour aller regarder les extrêmes — le même usage qu'avait la statistique d'asymétrie à la leçon 1.

La réponse corrigée est celle qui sert : des trajets plus longs vont avec moins d'eau consommée, r = −0,29 sur 2 392 ménages, ce qui est la relation que le cours EAH prédisait et que le fichier non corrigé sous-estimait.

7.6 Corrélés parce qu'autre chose a bougé les deux

```python
prices = pd.read_csv("market-prices-2024.v1.csv")
wide = prices.pivot_table(index="period", columns="commodity",
                          values="price_htg", aggfunc="mean").sort_index()

print(f"beans and maize, levels:           {wide['beans-black'].corr(wide['maize']):.3f}"
    )
changes = wide.diff().dropna()
print(f"beans and maize, month-on-month:   {changes['beans-black'].corr(changes['maize'])
    :.3f}")
```

```r
cor(wide$`beans-black`, wide$maize)
cor(diff(wide$`beans-black`), diff(wide$maize))
```

Le haricot et le maïs corrèlent à 0,986 en niveaux et à 0,876 en variations mensuelles. Aucun des deux nombres ne signifie que le prix du haricot déplace celui du maïs. Les deux séries portent la même inflation annuelle et le même choc de soudure, et la corrélation mesure le choc.

Corréler deux séries dans le temps donnera presque toujours un grand nombre, parce que presque tout suit une tendance. Différencier retire la tendance partagée et ce qui survit est le choc partagé — qui est une cause commune, et toujours pas un lien causal entre les deux produits.

Le cours d'épidémiologie faisait ce point avec la confusion. **Le coefficient de corrélation n'a aucun moyen de l'exprimer**, et c'est pourquoi la phrase en dessous doit le faire.

7.7 Et le regroupement de la leçon précédente

```
by_school = joined.groupby("school_id")[["attendance", "literacy"]].mean()
print(f"student level  r = {joined['attendance'].corr(joined['literacy']):.3f}  n = {len(
    joined)}")
print(f"school level   r = {by_school['attendance'].corr(by_school['literacy']):.3f}  n =
    {len(by_school)}")
```

```
# Same two variables, two units of analysis, two answers.
```

0,039 sur 659 élèves et 0,199 sur 24 écoles. Le chiffre au niveau école a un intervalle de −0,22 à +0,56 sur 24 points, donc il n'établit rien non plus — mais il est cinq fois plus grand, et un analyste qui aurait agrégé d'abord et rapporté r sans n aurait écrit une autre phrase.

Dites entre quelles unités porte la corrélation, à chaque fois. « La présence corrèle avec la littératie à 0,20 » est illisible sans savoir si les lignes sont des enfants ou des écoles.

7.8 Écrivez la phrase

Le coefficient est un nombre et la phrase en dessous est l'endroit où l'analyse est honnête ou ne l'est pas.

Pas ceci : « La présence est corrélée aux résultats de littératie. » Vrai au niveau école, faux au niveau élève, et infalsifiable tel quel.

Ni ceci : « Il n'y a aucune relation entre présence et littératie. » Plus fort que ce que portent les données ; l'intervalle atteint +0,12.

Ceci : « Sur 659 élèves, la présence du trimestre février-avril ne montre aucune association avec la littératie de fin d'année (r = 0,04, IC à 95 % −0,04 à +0,12). Le registre ne peut pas soutenir l'hypothèse que la présence porte l'apprentissage dans cette cohorte. Notez que la présence varie peu — la moitié centrale des élèves se situe entre 86,7 % et 97,8 % — de sorte que cette analyse détecte mal une relation aux faibles présences, là où on l'attendrait. »

La troisième phrase est celle à copier. Elle énonce le constat, l'intervalle, ce qu'il signifie pour le programme, et la raison pour laquelle l'analyse pourrait avoir manqué quelque chose de réel — et elle tient en quatre lignes.

7.9 Rapportez-le en entier

```
Attendance and learning outcomes, endline 2024
```

```
Attendance vs endline literacy   r = 0.04   95% CI -0.04 to +0.12   n = 659
Attendance vs endline numeracy   r = -0.01  95% CI -0.08 to +0.07   n = 659

Between students within the term. No association at student level.
Aggregated to the 24 schools, r = 0.20 (95% CI -0.22 to +0.56), which is
also consistent with no relationship.

Attendance is compressed: the interquartile range is 86.7% to 97.8%, so
students with the low attendance that a learning effect would show up at
are rare in this cohort. This is a null result about the range observed,
not about attendance in general.
```

Le dernier paragraphe est ce qui empêche de surlire le résultat nul, et c'est le pendant de la phrase « sous-puissant, non nul » de la leçon 4. Une corrélation nulle sur une plage étroite ne dit rien de ce qui se passe en dehors.

7.10 La suite

La dernière leçon assemble tout cela en une section statistique de rapport — ce qui y entre, ce qui reste en annexe, et ce qu'un relecteur demandera et que ce cours a déjà calculé.

CHAPITRE 8

La section statistique qu'un relecteur ne peut pas démonter

Leçon 8 sur 8 · 180 min

Sept leçons ont produit onze nombres. Celle-ci les met dans un rapport — ce qui va dans le corps, ce qui va en annexe, ce qu'un relecteur demandera, et les quatre phrases qui répondent à chaque question avant qu'elle soit posée.

8.1 Ce que fait réellement un relecteur

Un relecteur lisant un rapport de programme fait quatre choses, dans cet ordre, et chacune trouve sa réponse dans quelque chose que ce cours a calculé.

Le relecteur demande	Répondu par
« À quel point êtes-vous sûr de ce nombre ? »	L'intervalle — leçon 2
« Cette différence est-elle réelle ? »	Le test et ses hypothèses — leçon 3
« Est-ce que cela compte ? »	La taille d'effet dans l'unité de la décision — leçon 4
« Combien de choses avez-vous regardées ? »	Le nombre de tests et la correction — leçon 5

L'ordre n'est pas celui dans lequel vous les avez calculées. L'intervalle vient en premier parce que c'est le seul qu'un lecteur puisse confronter à sa propre connaissance du programme, et le nombre de tests vient en dernier parce que c'est celui qu'on oublie de demander.

8.2 Les quatre phrases

Toute affirmation statistique d'un rapport peut s'écrire en quatre phrases, et la structure ci-dessous est ce que répète le bloc « rapportez-le en entier » à la fin de chaque leçon.

Un : le nombre et son intervalle. « L'aboutissement des orientations parmi les cas signalant un handicap vaut 26,7 % (IC à 95 % 21,1 à 33,2, n = 202). »

Deux : la comparaison et son test. « C'est 19,4 points en dessous des cas sans handicap signalé (46,2 %, n = 1 436 ; IC à 95 % sur la différence −26,1 à −12,8, test z à deux échantillons, $p < 0{,}001$). »

Trois : l'ampleur, dans l'unité de la décision. « Rapportée à la charge de cas 2024, cette différence vaut 39 cas qui auraient atteint un service si l'aboutissement avait égalé celui du reste de la charge. »

Quatre : ce que l'analyse ne peut pas dire. « La comparaison est observationnelle. Les cas ne sont pas randomisés et l'écart peut refléter la façon dont les cas signalant un handicap entrent dans le système plutôt que la façon dont ils y sont traités. »

```
import pandas as pd
import numpy as np

protection = pd.read_csv("protection-referrals-2024.v1.csv")
consenting = protection[protection["consent_to_refer"]]
```

```
disability = consenting["disability_reported"].isin([True, "true", "Yes"])
reached = consenting["referral_accepted"] & consenting["days_to_first_service"].notna()

k, n = reached[disability].sum(), disability.sum()
comparator = reached[~disability].mean()
print(f"cases short of the comparator rate: {comparator * n - k:.0f}")
```

```
# Sentence three is arithmetic, and it is the sentence a manager reads.
```

La troisième phrase est celle que les analystes sautent et dont les responsables ont besoin. Des points de pourcentage ne commandent rien ; trente-neuf cas si.

8.3 Corps, annexe et script

Trois endroits, et mettre un nombre au mauvais est l'échec le plus courant d'une section statistique.

Le corps porte l'affirmation. L'estimation, son intervalle, la comparaison, l'effet en unités de programme, et la limite. Cinq lignes par constat, pas plus.

L'annexe porte la machinerie. Noms des tests, statistiques, degrés de liberté, vérifications des hypothèses, nombre de comparaisons lancées, correction appliquée, et les analyses de sous-groupes qui n'ont rien trouvé.

Le script porte la reproduction. Chaque nombre des deux, produit par du code qui tourne à partir du fichier versionné — c'est pourquoi chaque bloc « rapportez-le en entier » de ce cours est imprimé par le script même qui l'a calculé.

```
def claim(label, k, n, digits=1):
    p = k / n
    z = 1.96
    denom = 1 + z**2 / n
    centre = (p + z**2 / (2 * n)) / denom
    half = z * np.sqrt(p * (1 - p) / n + z**2 / (4 * n**2)) / denom
    return (f"{label}: {p:.{digits}%} "
            f"(95% CI {centre - half:.{digits}%} to {centre + half:.{digits}%}, n={n})")

print(claim("Referral completion, disability reported", 54, 202))
```

```
# Generate the sentence, do not type it. A typed interval drifts from its number.
```

Un nombre saisi à la main dans un document est un nombre qui sera faux après la prochaine correction de données. Produire la phrase relève de la même discipline que produire les figures depuis le jeu de données.

8.4 Le tableau d'annexe

```
annex = pd.DataFrame([
    {"comparison": "Completion by disability status", "test": "two-sample z",
     "statistic": "z = -5.21", "p": "<0.001", "n": "202 vs 1,436",
     "effect": "-19.4 pts (CI -26.1 to -12.8)"},
    {"comparison": "Over-age by sex", "test": "two-sample z",
     "statistic": "z = 1.84", "p": "0.066", "n": "510 vs 542",
     "effect": "+5.5 pts (CI -0.4 to +11.3)"},
    {"comparison": "Attendance by sex", "test": "two-sample z",
```

```
     "statistic": "z = 2.28", "p": "0.023", "n": "68,267 marks",
     "effect": "+0.56 pts, risk ratio 1.006"},
    {"comparison": "Attendance by school feeding", "test": "Welch t, school level",
     "statistic": "t = 3.11, df 12.7", "p": "0.008", "n": "15 vs 9 schools",
     "effect": "+5.2 pts (CI 1.6 to 8.8)"},
])
print(annex.to_string(index=False))


# One row per test run, including the ones that found nothing.
```

Comparaison	Test	Statistique	p	Effet
Aboutissement selon le handicap	z à deux échantillons	z = −5,21	<0,001	−19,4 pts
Retard d'âge selon le sexe	z à deux échantillons	z = 1,84	0,066	+5,5 pts
Présence selon le sexe	z à deux échantillons	z = 2,28	0,023	+0,56 pt, RR 1,006
Présence selon la cantine	t de Welch, niveau école	t = 3,11, ddl 12,7	0,008	+5,2 pts
Retard d'âge par sexe, 24 écoles	z ×24	2 sur 24 à p<0,05	—	0 survit à Bonferroni

Trois de ces cinq lignes sont en annexe et non dans le corps, parce qu'un écart non significatif, une significativité sans portée et une famille de tests nuls sont tous des choses qu'un relecteur doit pouvoir retrouver, et qu'aucune n'est un constat de programme.

Lister les tests qui n'ont rien trouvé est ce qui rend crédibles ceux qui ont trouvé quelque chose. Une annexe ne contenant que des tests réussis dit au relecteur que le compte a été arrangé.

8.5 Le paragraphe de limites, écrit depuis le cours

Quatre limites reviennent dans chaque analyse de cette plateforme, et chacune a une forme d'une ligne qui est honnête plutôt que défensive.

Comparaison observationnelle. « Les groupes n'ont pas été randomisés ; les différences peuvent refléter qui entre dans chaque groupe plutôt que ce qui leur arrive. »

Regroupement en grappes. « Le programme est attribué par école, donc l'analyse emploie 24 unités indépendantes. Des statistiques au niveau élève surestimeraient la précision d'un facteur 1,8. »

Comparaisons multiples. « Vingt-quatre tests au niveau école ont été lancés ; 1,2 résultat significatif est attendu par hasard et 2 ont été observés, dont aucun ne survit à la correction. »

Restriction de l'étendue. « La présence de cette cohorte va de 86,7 % à 97,8 % pour la moitié centrale, si bien que cette analyse ne peut rien dire des élèves à faible présence, là où un effet serait attendu. »

Chacune tient en une phrase et chacune devance une question précise. Un paragraphe de limites qui dit « les données comportent des limites » ne devance rien et se lit comme ayant quelque chose à cacher.

8.6 Que faire quand le nombre refuse de trancher

Trois des résultats de ce cours sont réellement non tranchés, et chacun appelle une action différente.

Résultat	Pourquoi il ne tranche pas	Que faire
Retard d'âge par sexe, p = 0,066	Sous-puissant pour un écart de 5 points	Le dire ; regrouper avec le passage suiv
Présence et littératie, r = 0,04	Étendue restreinte	Rapporter le résultat nul et nommer la
Cantine et présence, 24 grappes	Peu de grappes, observationnel	Rapporter l'intervalle ; ne pas revendiq

« Les données ne peuvent pas trancher » est un constat sur lequel un programme peut agir, parce que l'action consiste à collecter autrement. Ce n'est pas la même chose que n'avoir rien à rapporter, et la version qui met un programme en difficulté est celle qui rapporte quand même un nombre.

8.7 Rapportez-le en entier

```
Statistical methods

  Proportions are reported with Wilson 95% confidence intervals. Group
  comparisons use two-sample z-tests for proportions and Welch's t-test for
  means. Where a programme is assigned above the level of the observation,
  the analysis is conducted at the level of assignment.

  Analyses were specified before the data were examined. Where a family of
  comparisons was run, the number of tests is stated with the result and a
  Bonferroni correction applied.

  All figures are generated from the committed dataset files by the scripts
  in this annex; no number in this report is typed by hand.

  Findings reported in the body: 2. Comparisons run in total: 29.
```

La dernière ligne est celle qui change la façon dont la section est lue. Deux constats sur vingt-neuf comparaisons est un rapport défendable énoncé ouvertement ; deux constats sans dénominateur est une affirmation qu'un relecteur n'a aucun moyen de peser.

8.8 La suite

Les deux cours restants du module 5 poussent plus loin. La **régression** remplace la discipline « une comparaison à la fois » de ce cours par un modèle qui ajuste sur plusieurs choses en même temps — et hérite de tous les problèmes vus ici, y compris les grappes et les comparaisons multiples. L'**évaluation d'impact** s'attaque directement à la quatrième phrase : quels protocoles permettent d'écrire *ceci a causé cela* plutôt que *ces deux groupes diffèrent*.

Mais la discipline est complète ici. Chaque nombre d'un rapport de programme peut porter un intervalle, une taille d'effet, un nombre de tests et une limite, et le faire coûte quatre phrases. **Les cours qui suivent rendent les nombres meilleurs ; celui-ci les rend honnêtes**, et aucune dose du premier ne remplace le second.

À propos de cette édition

Ce PDF est produit à partir de la même source que la version web de la formation ; les deux ne peuvent donc pas diverger. L'édition en ligne comporte le code exécutable, les jeux de données téléchargeables et les corrections publiées depuis la production de ce fichier.

> Tous les jeux de données cités dans cette formation sont synthétiques ou entièrement anonymisés. Aucun enregistrement ne renvoie à une personne réelle.

Cette formation est publiée en anglais et en français. L'édition française est une adaptation professionnelle reprenant la terminologie employée par l'UNICEF, l'OMS, le PAM, OCHA et les ministères nationaux — et non une traduction littérale.

Consulter et exécuter la formation en ligne sur data-analysis.cassion.dev/fr/cours/applied-statistics-for-programmes

Généré le 28 juillet 2026.