$p = 0.023$, and a third of one day

Boys attend 88.77% and girls 88.21%. On 68,267 attendance marks that is significant at $p = 0.023$. It is also a difference of one third of one school day per child per term, which no programme would act on and none should.

Pierre Robentz Cassion

Lesson 4 of 8

Applied Statistics for Programmes — Comparing two groups

What this lesson covers

- A significant result nobody should act on
- Convert it into the unit of the decision
- Why p got small: it was the n
- Report an effect size beside every p-value
- Set the threshold before you test
- The four cases
- Report it whole
- What comes next

A significant result nobody should act on — In Python (cont.)

```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)
```

A significant result nobody should act on — In Python (cont.)

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

A significant result nobody should act on — In R

```r
library(dplyr)

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

A significant result nobody should act on

Group	**Attended**	**Marks**	**Rate**
Boys	29,655	33,408	**88.77%**
Girls	30,749	34,859	**88.21%**

A significant result nobody should act on

- Difference 0.56 percentage points. $z = 2.28$, $p = 0.023$

Convert it into the unit of the decision — In Python

```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")
```

Convert it into the unit of the decision — In R

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

Convert it into the unit of the decision

- Sixty school days, so 0.56% is 0.34 days — about one third of one day per child per term
- Always convert an effect into the unit the decision is made in — Percentage points are the analyst's unit; days,. . .

Why p got small: it was the n — In Python

```
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}")
```

Why p got small: it was the n — In R

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

Why p got small: it was the n

Sample	Difference	p
5% of the marks (n=3,413)	+0.62%	0.570
20% (n=13,654)	+0.55%	0.314
100% (n=68,267)	**+0.56%**	**0.023**

Why p got small: it was the n

- The effect never changed. Only the sample did — A p-value is a statement about evidence, not about magnitude, and with...
- So a p-value alone can never tell you whether something matters — It tells you whether you can rule out zero, and zero...

Report an effect size beside every p-value

Measure	For	Read as
Difference in percentage points	Two proportions	The direct answer
Risk ratio	Two proportions, rare outcomes	How many times more likely
Cohen's d	Two means	Standard deviations of separation

Report an effect size beside every p-value — In Python

```python
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}")
```

Report an effect size beside every p-value — In R

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

Report an effect size beside every p-value

- Risk ratio 1.006 — Boys attend 0.6% more often in relative terms — a way of saying the same nothing

Set the threshold before you test — In Python

```
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}")
```

Set the threshold before you test — In R

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

Set the threshold before you test

- Three points of attendance is about 1.8 days a term — which is the scale at which a follow-up programme would be. . .

The four cases — In Python

```
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)
```

The four cases — In R

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

The four cases

- The third row is the one that gets mishandled — The over-age sex gap was 5.5 points with $p = 0.066$, and the instinct is...

Report it whole — Example

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.

Report it whole

- The last sentence is what makes the block honest — A reader who sees only "p = 0.023" will assume the difference...

What comes next

- Every test so far has been one comparison.

Read the full lesson, with runnable code

https://data-analysis.cassion.dev/courses/applied-statistics-for-programmes/stat-04-significant-and-irrelevant