

The 212 cases that are not a failure

88.5% of these cases consented to a referral. The 212 that did not are outside the performance denominator, because a person declining a referral is exercising a right rather than revealing a service gap — and counting them as failures does two wrong things at once.

Pierre Robentz Cassion

Lesson 3 of 8

Protection and GBV Data — The pathway

What this lesson covers

- The pathway has four gates
- Consent is not a performance measure
- Report all three numbers
- Where consent is not the gate
- The contradictions to resolve first
- What comes next

The pathway has four gates — In Python

```
import pandas as pd

referrals = pd.read_csv("protection-referrals-2024.v1.csv")

gates = {
    "cases recorded": len(referrals),
    "consented to referral": referrals["consent_to_refer"].sum(),
    "referral made": (referrals["consent_to_refer"] &
                      referrals["referral_made"]).sum(),
    "referral accepted": (referrals["referral_made"] &
                          referrals["referral_accepted"]).sum(),
    "reached a service": (referrals["referral_accepted"] &
                          referrals["days_to_first_service"].notna()).sum(),
}

for name, count in gates.items():
    print(f"{name:24} {count:>5}")
```

The pathway has four gates — In R

```
library(dplyr)

referrals |> summarise(
  cases = n(),
  consented = sum(consent_to_refer),
  made = sum(consent_to_refer & referral_made),
  accepted = sum(referral_made & referral_accepted),
  reached = sum(referral_accepted & !is.na(days_to_first_service))
)
```

The pathway has four gates

Gate	Cases	Share of the previous gate
Cases recorded	1,850	—
Consented to referral	1,638	88.5%
Referral made	1,142	69.7%
Referral accepted	757	66.3%
Reached a service	717	94.7%

Consent is not a performance measure

- As failures — End-to-end completion becomes $717 / 1,850 = 38.8\%$
- As exclusions — Completion becomes $717 / 1,638 = 43.8\%$ on cases that consented
- As a finding in their own right — 11.5% declined, and *why* is a service design question worth asking separately

Consent is not a performance measure — In Python

```
consenting = referrals["consent_to_refer"]
reached = referrals["referral_accepted"] & referrals["days_to_first_service"].notna()

print(f"on all cases:          {reached.sum() / len(referrals):.1%}")
print(f"on consenting cases: {reached.sum() / consenting.sum():.1%}")
print(f"declined:             {(~consenting).sum()} ({(~consenting).mean():.1%}")
```

Consent is not a performance measure — In R

```
referrals |> summarise(  
  all_cases = mean(referral_accepted & !is.na(days_to_first_service)),  
  consenting = sum(referral_accepted & !is.na(days_to_first_service)) / sum(consent_to_refer)  
)
```

Consent is not a performance measure

- Counting a non-consenting case as a pathway failure does two wrong things at once — It misstates performance by five. . .
- Both matter and the second matters more — An indicator that treats declining as failure creates pressure on caseworkers. . .

Report all three numbers — Example

Referral pathway, 1,850 cases

Consented to referral	88.5%	1,638	
Declined	11.5%	212	reported separately
Reached a service, of consenting	43.8%	717	
Reached a service, of all cases	38.8%		for reference only

Report all three numbers

- Publish the consent rate as its own line — A falling consent rate is a signal about trust in the service, and it is. . .

Where consent is not the gate

- Child protection cases involving a young child — operate under a best-interests determination rather than the child's. . .
- Life-threatening emergencies — proceed to a life-saving referral without waiting for consent to a data transfer

Where consent is not the gate — In Python

```
by_category = referrals.groupby("case_category")["consent_to_refer"].agg(  
    ["mean", "size"]  
)  
print((by_category * [100, 1]).round(1))
```

Where consent is not the gate — In R

```
referrals |> summarise(consent = mean(consent_to_refer), n = n(),  
                        .by = case_category)
```

Where consent is not the gate

- Check whether the consent rate differs by category before applying one rule to all of them — Where it does, the...

The contradictions to resolve first — In Python

```
impossible = (referrals["days_to_first_service"].notna() &
              ~referrals["referral_accepted"])
no_referral = impossible & ~referrals["referral_made"]
no_consent = impossible & ~referrals["consent_to_refer"]

print(f"time to service but referral not accepted: {impossible.sum()}")
print(f"  of which no referral was made at all:    {no_referral.sum()}")
print(f"  of which there was no consent:          {no_consent.sum()}")
```

The contradictions to resolve first — In R

```
referrals |> filter(!is.na(days_to_first_service), !referral_accepted) |>  
  count(referral_made, consent_to_refer)
```

The contradictions to resolve first

- Eleven cases record a time to first service on a referral that was never accepted — Six of them show no referral made. . .

What comes next

- The pathway loses cases at every gate.

Read the full lesson, with runnable code

[https://data-analysis.cassion.dev/courses/protection-and-gbv-data/
prot-03-the-consent-gate](https://data-analysis.cassion.dev/courses/protection-and-gbv-data/prot-03-the-consent-gate)