

La tournée que personne n'a conduite

2 629 visites faites sur 2 904 dues. Nord-Ouest se lit comme le meilleur des trois districts à 76,4 % et sa réponse est incertaine à treize points près, contre deux et demi pour le district qui a presque tout visité.

Pierre Robentz Cassion

Leçon 7 sur 8

Analyse EAH — Ce que le suivi n'a pas vu

Ce que couvre cette leçon

- Le dénominateur que personne n'a interrogé
- La couverture n'est pas répartie uniformément, et c'est tout le problème
- Encadrez la réponse
- L'absence est-elle aléatoire ?
- Ce qu'il ne faut pas faire
- Rapportez la couverture à côté de chaque taux
- La suite

Le dénominateur que personne n'a interrogé — En Python

```
import pandas as pd

points = pd.read_csv("water-point-monitoring-2024.v1.csv", parse_dates=["visit_date"])

due = points["water_point_id"].nunique() * 12
made = len(points)
print(f"due {due:},, made {made:},, coverage {made / due:.1%}")
```

Le dénominateur que personne n'a interrogé — En R

```
library(dplyr)

points |> summarise(
  due = n_distinct(water_point_id) * 12,
  made = n(),
  coverage = n() / (n_distinct(water_point_id) * 12)
)
```

Le dénominateur que personne n'a interrogé

- 2 629 sur 2 904 — une couverture de 90,5 % — Deux cent soixante-quinze tournées n'ont jamais été conduites, et le...

La couverture n'est pas répartie uniformément, et c'est tout le problème — En Python

```
coverage = points.groupby("district").agg(  
    points=("water_point_id", "nunique"),  
    visits=("water_point_id", "size"),  
)  
coverage["due"] = coverage["points"] * 12  
coverage["coverage"] = coverage["visits"] / coverage["due"]  
print(coverage.round(3))
```

La couverture n'est pas répartie uniformément, et c'est tout le problème — En R

```
points |>  
  summarise(points = n_distinct(water_point_id), visits = n(), .by = district) |>  
  mutate(due = points * 12, coverage = visits / due)
```

La couverture n'est pas répartie uniformément, et c'est tout le problème

District	Points	Visites faites	Dues	Couverture
Sud-Est	79	913	948	96,3 %
Centre	79	882	948	93,0 %
Nord-Ouest	84	834	1 008	82,7 %

La couverture n'est pas répartie uniformément, et c'est tout le problème — En Python

```
monthly = points.pivot_table(  
    index=points["visit_date"].dt.month, columns="district",  
    values="water_point_id", aggfunc="size",  
)  
print(monthly)
```

La couverture n'est pas répartie uniformément, et c'est tout le problème — En R

```
points |> count(district, month = lubridate::month(visit_date)) |>  
  tidyr::pivot_wider(names_from = district, values_from = n)
```

Encadrez la réponse

- **Borne haute** : chaque visite manquée aurait trouvé un point en service. C'est le taux observé, qui suppose déjà que. . .
- **Borne basse** : chaque visite manquée aurait trouvé un point en panne.

Encadrez la réponse — En Python

```
WORKING = {"functional", "partially-functional"}
ok = points["functional_status"].isin(WORKING)

bounds = points.assign(ok=ok).groupby("district").agg(
    ok=("ok", "sum"), made=("ok", "size"), points=("water_point_id", "nunique"),
)
bounds["due"] = bounds["points"] * 12
bounds["observed"] = bounds["ok"] / bounds["made"]
bounds["lower"] = bounds["ok"] / bounds["due"]
bounds["band"] = bounds["observed"] - bounds["lower"]
print((bounds[["observed", "lower", "band"]] * 100).round(1))
```

Encadrez la réponse — En R

```
points |>
  mutate(ok = functional_status %in% c("functional", "partially-functional")) |>
  summarise(ok = sum(ok), made = n(), pts = n_distinct(water_point_id),
            .by = district) |>
  mutate(due = pts * 12, observed = ok / made, lower = ok / due)
```

Encadrez la réponse

District	Observé	Borne basse	Fourchette
Nord-Ouest	76,4 %	63,2 %	13,2
Centre	75,7 %	70,5 %	5,3
Sud-Est	71,4 %	68,8 %	2,6

Encadrez la réponse

- Nord-Ouest est le meilleur district sur le chiffre observé et le seul dont le classement n'est pas sûr — Sa fourchette. . .
- Un nombre calculé sur 96 % de son dénominateur et un nombre calculé sur 83 % ne sont pas comparables, et rien dans le tableau ne le dira si vous ne l'y mettez pas

L'absence est-elle aléatoire? — En Python

```
observed = points.pivot_table(
    index="water_point_id", columns="round", values="functional_status",
    aggfunc="first",
)
followed_by_gap = []
for point, row in observed.iterrows():
    for r in range(1, 12):
        if pd.isna(row.get(r)):
            continue
        broken = row[r] not in WORKING
        followed_by_gap.append((pd.isna(row.get(r + 1)), broken))

check = pd.DataFrame(followed_by_gap, columns=["gap_next", "broken"])
print(check.groupby("gap_next")["broken"].agg(["mean", "size"]).round(3))
```

L'absence est-elle aléatoire? — En R

Same shape: was the point broken at the visit immediately before a missed round?

L'absence est-elle aléatoire ?

- 26,7 % des visites suivies d'une tournée manquée ont trouvé un point en panne, contre 24,7 % des visites suivies d'une. . .

Ce qu'il ne faut pas faire

- Reporter la dernière observation — Remplir chaque tournée manquante par l'état précédent du point
- Imputer par la moyenne du district — Suppose que les points manqués ressemblent aux points visités du même district, ce. . .
- Retirer le district — Ôte du rapport le district le moins bien couvert, qui est presque toujours aussi le plus. . .

Ce qu'il ne faut pas faire — En Python

```
print("Reported: 76.4% (observed), 63.2% to 76.4% (bounds), coverage 82.7%")  
print("Not reported: a single imputed number that hides which of these it is")
```

Ce qu'il ne faut pas faire — En R

The band and the coverage are the finding. Neither is a caveat.

Rapportez la couverture à côté de chaque taux — Exemple

Water point functionality by district, 2024

District	Functionality	Coverage	Range if unvisited were broken
Sud-Est	71.4%	96.3%	68.8 - 71.4
Centre	75.7%	93.0%	70.5 - 75.7
Nord-Ouest	76.4%	82.7%	63.2 - 76.4

275 of 2,904 scheduled visits were not made, concentrated in Nord-Ouest during June to September when roads are impassable. Districts are not ranked here: Nord-Ouest's range overlaps both others.

- Vous disposez maintenant de tous les nombres EAH que ce cours peut produire et d'un jeu de réserves attaché à chacun.

Lire la leçon complète, avec le code exécutable

```
https://data-analysis.cassion.dev/fr/cours/wash-analysis/  
wash-07-the-round-nobody-drove
```