

Dates, âges et périodes de rapportage

L'âge en mois révolus calculé correctement plutôt qu'en divisant des jours, des périodes de rapportage qui survivent à un mois partiel, et l'arithmétique de frontière dont dépend un standard de croissance.

Pierre Robentz Cassion

Leçon 7 sur 8

Python pour les données de programme — Produire et transmettre

Ce que couvre cette leçon

- Pourquoi cela mérite une leçon entière
- Analyser la date, une fois, explicitement
- L'âge en mois révolus
- Les périodes de rapportage
- Rééchantillonner une série temporelle
- Les durées
- Les fuseaux horaires, brièvement
- Ce qui vient ensuite

Pourquoi cela mérite une leçon entière

- L'arithmétique des dates paraît être la partie facile et produit certaines des erreurs les plus coûteuses du secteur.

Analyser la date, une fois, explicitement — En Python

```
import pandas as pd

muac = pd.read_csv(path)
muac["screening_date"] = pd.to_datetime(
    muac["screening_date"], format="%Y-%m-%d", errors="raise"
)
```

Analyser la date, une fois, explicitement

- **Énoncez le format.** Un format inféré peut changer d'un fichier à l'autre selon la composition des valeurs — la même...
- **Utilisez** `errors="raise"`. L'alternative, `errors="coerce"`, transforme toute date illisible en valeur manquante :...

L'âge en mois révolus — En Python

Faux assez souvent pour que cela compte.

```
age_months = ((visit_date - date_of_birth).dt.days / 30.4375).astype(int)
```

L'âge en mois révolus — En Python

```
def age_in_months(dob: pd.Series, visit: pd.Series) -> pd.Series:
    months = (visit.dt.year - dob.dt.year) * 12 + (visit.dt.month - dob.dt.month)
    # Le quantieme n'est pas atteint : le dernier mois n'est pas revolu.
    return months - (visit.dt.day < dob.dt.day).astype(int)
```

L'âge en mois révolus — En Python

```
dob    = pd.to_datetime(["2022-03-15", "2022-03-16", "2022-02-28"])
visit  = pd.to_datetime(["2024-03-15", "2024-03-15", "2024-03-01"])

age_in_months(pd.Series(dob), pd.Series(visit)).tolist()    # [24, 23, 24]
```

L'âge en mois révolus

- Pourquoi un mois compte ici
- Les standards de croissance de l'OMS changent de mesure exactement à 24 mois — En dessous de 24 mois, l'enfant est. . .
- Les tranches d'âge sont l'autre endroit — 0-5, 6-23, 24-59 mois sont les tranches nutritionnelles standard, et un. . .

L'âge en mois révolus — En Python

```
BANDS = [0, 6, 24, 60]
LABELS = ["0-5m", "6-23m", "24-59m"]

muac["age_band"] = pd.cut(
    muac["age_months"], bins=BANDS, labels=LABELS, right=False, include_lowest=True
)
```

L'âge en mois révolus — En Python

```
check = muac.loc[muac["age_months"].isin([5, 6, 23, 24, 59, 60]),  
                ["age_months", "age_band"]].drop_duplicates().sort_values("age_months")  
print(check)
```

Les périodes de rapportage — En Python

```
muac["month"] = muac["screening_date"].dt.to_period("M")  
  
by_month = muac.groupby("month").size()
```

Les périodes de rapportage — Exemple

2024-01	207
2024-02	391
...	
2024-11	375
2024-12	156

Les périodes de rapportage — En Python

```
current = pd.Period("2024-06", freq="M")  
print(current - 1)                # 2024-05  
print(current.start_time, current.end_time)
```

- Les périodes partielles sont le piège

Les périodes de rapportage — En Python

```
first, last = muac["screening_date"].min(), muac["screening_date"].max()
print(first.date(), last.date())

muac["partial_period"] = muac["month"].isin(
    [first.to_period("M"), last.to_period("M")]
)
```

Les périodes de rapportage

- Étiquetez-les plutôt que de les supprimer — Un lecteur qui voit décembre s'effondrer sur un graphique conclut que le. . .
- Aligner deux sources sur des périodes

Les périodes de rapportage — En Python

```
epi = pd.read_csv(path, parse_dates=["period"])
epi["month"] = epi["period"].dt.to_period("M")
print(epi["month"].nunique())          # 12
```

Rééchantillonner une série temporelle — En Python

```
daily = muac.set_index("screening_date")
weekly = daily.resample("W")["child_id"].size()
monthly = daily.resample("MS")["child_id"].size()      # MS = debut de mois
```

```
referrals["days_to_service"] = (  
    referrals["service_date"] - referrals["referral_date"]  
) .dt.days
```

```
hours = (referrals["service_date"] - referrals["referral_date"]) / pd.Timedelta(hours=1)
referrals["same_day"] = hours < 24
```

```
impossible = referrals["days_to_service"] < 0  
print(f"service saisi avant le referencement : {impossible.sum()}")
```

Les fuseaux horaires, brièvement — En Python

```
submissions["submitted_at"] = pd.to_datetime(  
    submissions["submitted_at"], utc=True  
) .dt.tz_convert("America/Port-au-Prince")  
  
submissions["submission_date"] = submissions["submitted_at"].dt.date
```

Ce qui vient ensuite

- Chaque colonne dit désormais ce qu'elle signifie, y compris celles faites de dates.

Lire la leçon complète, avec le code exécutable

[https://data-analysis.cassion.dev/fr/cours/python-for-programme-data/
python-07-dates-ages-periods](https://data-analysis.cassion.dev/fr/cours/python-for-programme-data/python-07-dates-ages-periods)