TD n°1 : Processus global de la qualité des données

Criminalité à Cambridge

Organisation du travail et livrables

Éléments obligatoires	Éléments optionnels (bonus)
☐ Environnement Python isolé (ex. `venv`)	☐ Versionnement avec `git` (commits réguliers)
☐ Fichier `requirements.txt`	☐ Tests automatisés (`pytest`)
☐ Fichier `README.md`	☐ Journalisation des traitements (`logging`)
☐ Structuration du projet (src/ data/ ...)	
☐ Justification des décisions de traitement	
Le projet doit pouvoir être exécuté par un tiers à partir des instructions fournies.	*Les éléments optionnels ne sont pas obligatoires mais seront valorisés.*

Voir le fichier `orga_bonnes_pratiques.pdf` *pour plus de détails.*

Contexte

Vous êtes consultant pour la ville de Cambridge, au sein du service « Sécurité ». Le service de police vous confie des données de crimes déclarés entre 2009 et 2024, issues de systèmes opérationnels hétérogènes, destinées à produire des indicateurs d'aide à la décision.

Avant toute exploitation, un audit de qualité des données est nécessaire afin d'évaluer leur fiabilité et de définir des règles de traitement adaptées.

1 Profilage et exploration

- Chargez le dataset `crime_reports_broken.csv`
- Afficher :
  - nombre de lignes
  - type des colonnes
  - nombre de valeurs manquantes par colonnes
- Identifier au moins 3 problèmes de qualité
- Établissez un dictionnaire des données pour chaque variable sélectionnée sous forme de tableau (nom, type, définition, exemple, ...)

2 Audit de la qualité

- Définissez des **fonctions de calcul d'indicateurs de qualité** permettant de mesurer (en %), pour un dataset donné :
  - la **complétude** de `File Number`, `Crime` et `Neighborhood`,
  - l'**unicité** de `File Number` (proportion de valeurs uniques),
  - le **taux de doublons exacts** (lignes strictement identiques),
  - le **taux de dates invalides** dans `Date of Report` (dates non parsables),
  - le **taux d'incohérences temporelles** : `Date of Report` antérieure au début de `Crime Date Time`,
  - le **taux de valeurs non conformes** dans `Reporting Area`.
- Implémentez ensuite une **fonction principale** regroupant l'ensemble de ces indicateurs et retournant une structure synthétique (ex. dictionnaire ou `Series` Pandas).
- Appliquez cette fonction au dataset initial.
- Proposez des **seuils** d'acceptation pour au moins 3 indicateurs (ex : complétude ≥ 95%).

3 Traitement

- Créez un dataframe de travail en faisant une copie de l'original
- Implémentez des règles de traitement pour les cas suivants :
  - **ID unique** en doublon
  - **Crime** null
  - **Date of Report** invalide
  - **Date of Report** avant **Crime Date Time**
  - **Reporting Area** invalide
  - **Neighborhood** invalide

Le service de police vous fournit la liste suivante comme référentiel officiel des quartiers :

```
VALID_NEIGHBORHOODS = {
    "Cambridgeport",
    "East Cambridge",
    "Mid-Cambridge",
    "North Cambridge",
    "Riverside",
    "Area 4",
    "West Cambridge",
    "Peabody",
    "Inman/Harrington",
    "Highlands",
    "Agassiz",
    "MIT",
    "Strawberry Hill",
}
```

- **Enrichissement (préparation cartographie)** : créez une nouvelle colonne `reporting_area_group` à partir de `Reporting Area` en extrayant le groupe de centaines (ex : 602 → 6 ; 1109 → 11).
- Vérifiez que la nouvelle colonne ne contient pas de valeurs aberrantes (ex : négatives, très élevées) et normalisez-les si nécessaire.
- Exporter le dataset nettoyé dans un nouveau fichier CSV `crime_reports_clean.csv` .

4 Monitoring de la qualité des données

- Recalculer au moins 3 indicateurs de qualité après traitement.
- Comparer les résultats avant / après nettoyage.
- Identifier les indicateurs dont l'évolution est significative.

5 Cartographie : choroplèthe des crimes par quartier

Le maire de Cambridge souhaite disposer d'une visualisation simple de la répartition des crimes par quartier, afin de communiquer sur les zones les plus touchées. Les données ne doivent pas permettre d'identifier une adresse précise : le niveau d'agrégation retenu est donc le **quartier**.

- **Agrégation des crimes**
  - Calculez le **nombre de crimes par quartier** à partir du dataset nettoyé `crime_reports_clean.csv` .
  - Vérifiez que la somme des crimes agrégés correspond bien au nombre de lignes du dataset (hors valeurs manquantes).
- **Référentiel géographique**
  - Récupérez le fichier `BOUNDARY_CDDNeighborhoods.geojson` représentant les quartiers de Cambridge.
  - Identifiez les colonnes représentant le nom du quartier et le code du quartier.
- **Jointure par code (référentiel)**
  - Réalisez une jointure entre les crimes agrégés et le GeoJSON.
  - Vérifiez que la jointure ne laisse pas de quartiers « orphelins » (sans crimes ou sans polygone).
- **Production de la carte**
  - Produisez une **carte choroplèthe** du nombre de crimes par quartier (du vert au rouge).
  - Ajoutez une légende et un tooltip (au survol) affichant le nom du quartier et le nombre de crimes.
  - (Bonus) Exportez la carte en HTML (`map.html`) pour diffusion.

Questions :

- Pourquoi le terme « quartier le plus dangereux » peut-il être trompeur avec un indicateur en volume brut ?