Projet Data Analytics Spark + MLlib + Sécurité (version GitHub)

Objectif

En binôme, les étudiants doivent produire un **pipeline Spark complet**, depuis la collecte des données jusqu'à l'analyse avancée avec MLlib, en incluant la **sécurisation des données** et les **agrégations pour reporting**.
L'ensemble du projet doit être **hébergé sur GitHub**, avec un **README détaillé** expliquant toutes les étapes, les choix et les interprétations.

Workflow intégré du projet

1. **Choix des datasets**
    - Récupérer **deux datasets volumineux** (≥100 000 lignes chacun) sur un sujet concret (transport, santé, finance, consommation…).
    - Exemples :
        - NYC Taxi & Limousine Trip Data
        - Open Food Facts
        - Base DVF – Transactions immobilières France
    - Justifier le choix des datasets et le problème métier que vous souhaitez explorer.
2. **Nettoyage et préparation**
    - Traiter les valeurs manquantes et formats inconsistants.
    - Préparer les colonnes pour réaliser la **jointure**.
    - Justifier vos transformations.
3. **Jointure des datasets**
    - Réaliser la jointure sur des clés pertinentes.

- Justifier le choix des clés et la pertinence de la jointure.

4. **Sécurisation des données**
    - Identifier les colonnes sensibles.
    - Si aucune donnée sensible réelle n'est présente → **simuler des colonnes sensibles** (ex : ID clients, emails).
    - Appliquer des techniques de sécurisation :
        - Hashage (SHA-256) pour pseudonymiser les identifiants
        - Masquage partiel pour colonnes confidentielles
    - Justifier vos choix et expliquer comment la sécurité est assurée.
5. **Agrégations et export pour reporting**
    - Calculer des **indicateurs clés** (moyenne, somme, top N, taux, segmentation, etc.) pour la direction.
    - Exporter le (s) **DataFrame (s) sécurisé (s) et agrégé (s)**.
    - Justifier le choix des agrégations et leur utilité pour le reporting.
6. **Analyse avancée avec MLlib**
    - Choisir un algorithme adapté :
        - Clustering (KMeans)
        - Régression (Linear Regression, Decision Tree)
        - Classification (Logistic Regression, Random Forest)
    - Former le modèle, l'évaluer et interpréter les résultats.
    - Justifier le choix du modèle et l'interprétation des résultats.
7. **Visualisation et interprétation**
    - Créer des graphiques synthétiques pour illustrer vos insights.
    - Les visualisations doivent être intégrées dans le notebook et expliquées.

Livrables GitHub

- **Notebook Spark** complet : ingestion, nettoyage, jointure, sécurisation, agrégations, MLlib, visualisations
- **DataFrames exportés** sécurisés et agrégés
- **README.md** détaillé :
    - Sujet et datasets choisis
    - Nettoyage et jointure
    - Sécurisation des données
    - Agrégations et insights pour la direction
    - Choix et interprétation du modèle MLlib
    - Visualisations et interprétation
- **Lien GitHub** à inclure en bas du notebook.

Inspirez-vous de ce projet pour le format du README et de l'organisation : https://github.com/Esther-Wend/Package_R_sisepls

Grille de notation – Projet Data Analytics Spark + MLlib + Sécurité

Catégories	Critères	Points	Commentaires
1. Collecte et choix des datasets	Qualité et pertinence des datasets choisis (≥100 000 lignes chacun), justification du choix par rapport à la problématique	2	Les datasets doivent être cohérents et pertinents pour le problème métier
2. Nettoyage et préparation des données	Traitement des valeurs manquantes, formats corrects, préparation pour jointure	2	Les transformations doivent être logiques et justifiées
3. Jointure des datasets	Jointure correcte sur clés pertinentes, justification de la logique de jointure	2	La jointure doit enrichir les données de manière cohérente
4. Sécurisation des données	Identification des données sensibles ou simulation de colonnes sensibles, application correcte du hashage, pseudonymisation ou masquage	3	La sécurisation doit être fonctionnelle et documentée
5. Agrégations et export pour reporting	Calcul d'indicateurs clés pertinents pour la direction, export des DataFrames sécurisés et agrégés	3	Les agrégations doivent être pertinentes et correctement exécutées
6. Analyse MLlib	Choix approprié du modèle (clustering, classification, régression), mise en œuvre correcte et interprétation des résultats	3	Le modèle doit correspondre au problème et être justifié
7. Visualisations et interprétation	Graphiques clairs et explicatifs, interprétation des insights	2	Visualisations compréhensibles et pertinentes pour le reporting
8. Documentation GitHub / README	README complet : explications des choix, descriptions des étapes, interprétation des résultats, lien GitHub présent	3	La documentation doit permettre à un lecteur externe de comprendre le projet et ses résultats