Groupe G3

Data Mining et Analyse de Patterns

Projet SDID 2025–2026

Matricules :

23618 – 23605 – 23635

13 février 2026

Table des matières

1 Introduction

Ce rapport présente le travail réalisé par le groupe G3 dans le cadre du projet sdid-energy-anomaly-2025-2026. Notre mission consiste à analyser les données historiques de consommation électrique afin d'identifier les comportements normaux, qui serviront de référence pour la détection d'anomalies effectuée par le groupe G4.

Le pipeline développé s'appuie sur des techniques avancées de **data mining non supervisé**, combinant réduction de dimension par Analyse en Composantes Principales (ACP) et clustering par l'algorithme DBSCAN.

2 Objectif du groupe G3

Le groupe G3 a pour objectif d'identifier les **comportements normaux de consommation électrique** à partir des données historiques stockées dans PostgreSQL, afin de fournir une base fiable pour la détection d'anomalies réalisée par le groupe G4.

Plus spécifiquement, nos objectifs sont :

- **Extraction** : Récupérer les données historiques depuis la base PostgreSQL
- **Prétraitement** : Normaliser les variables électriques pour les rendre comparables
- **Réduction dimensionnelle** : Projeter les données dans un espace à 2 dimensions via l'ACP
- **Clustering** : Identifier automatiquement les profils de consommation avec DBSCAN
- **Artefacts** : Générer des modèles réutilisables pour les autres groupes

3 Pré-requis

Les éléments suivants doivent être disponibles avant l'exécution du pipeline G3.

3.1 Infrastructure

- Base PostgreSQL active et accessible (mise en place par le groupe G2)
- Table `power_consumption` contenant au minimum 100 000 observations
- Variables requises : `global_active_power_kw`, `voltage_v`, `global_intensity_a`

3.2 Environnement logiciel

- Python 3.9 ou supérieur
- Bibliothèques : `pandas`, `numpy`, `scikit-learn`, `matplotlib`, `psycopg2`
- Environnement virtuel Python recommandé

4 Structure du projet G3

Le travail du groupe G3 est organisé selon la structure modulaire suivante.

```
G3_data_mining/
          artifacts/                # Modeles et parametres sauvegardes
                clusters.json
                dbscan_params.json
                pca.pkl
                scaler.pkl
          data_access/              # Acces aux donnees PostgreSQL
                __init__.py
                fetch_data.py
          preprocessing/            # Normalisation des donnees
                __init__.py
                scaling.py
          modeling/                 # Modeles ACP et DBSCAN
                __init__.py
                pca.py
                clustering.py
          visualization/            # Graphiques et exports
                __init__.py
                plot.py
                pca_clusters.png
          main.py                   # Point d'entree du pipeline
          README.md                # Documentation
```

Cette architecture permet une séparation claire des responsabilités et facilite la maintenance du code.

5 Méthodologie

5.1 Variables utilisées

Notre analyse porte sur trois variables électriques clés :

Variable	Description	Unité
`global_active_power_kw`	Puissance active totale	kW
`voltage_v`	Tension électrique	V
`global_intensity_a`	Intensité du courant	A

Table 1 – Variables électriques analysées

Les observations contenant des valeurs nulles sont automatiquement exclues de l'analyse pour garantir la cohérence des résultats.

5.2 Accès aux données

Ce module assure la lecture des données historiques depuis la base PostgreSQL mise en place par le groupe G2.

Le module `fetch_data.py` réalise les opérations suivantes :

— Établissement de la connexion à PostgreSQL via le module `common.db`
— Exécution d'une requête SQL filtrée excluant les valeurs nulles
— Limitation à 100 000 observations pour optimiser les performances
— Chargement des données dans un DataFrame Pandas

La requête SQL utilisée garantit la qualité des données en éliminant les observations incomplètes dès l'extraction.

5.3 Prétraitement et normalisation

La normalisation est essentielle pour comparer des variables ayant des échelles différentes (kW, V, A).

5.3.1 Choix du RobustScaler

Nous avons opté pour le `RobustScaler` plutôt que le `StandardScaler` pour les raisons suivantes :

— **Robustesse aux outliers** : Utilise la médiane et l'intervalle interquartile (IQR)
— **Préservation des anomalies** : Ne déforme pas excessivement les valeurs extrêmes
— **Stabilité** : Moins sensible aux valeurs aberrantes que la moyenne/écart-type

La transformation appliquée est :

$$X_{\text{scaled}} = \frac{X - \text{médiane}(X)}{\text{IQR}(X)}$$

Le modèle `scaler.pkl` est sauvegardé pour permettre la transformation de nouvelles données selon les mêmes paramètres.

5.4 Réduction de dimension par ACP

L'Analyse en Composantes Principales (ACP) permet de projeter les 3 variables dans un espace à 2 dimensions tout en conservant l'essentiel de la variance.

5.4.1 Principe de l'ACP

L'ACP recherche les directions de variance maximale dans les données. Elle transforme les variables originales corrélées en nouvelles variables non corrélées (composantes principales).

5.4.2 Paramètres utilisés

— **Nombre de composantes** : 2 (pour visualisation en 2D)
— **Seed aléatoire** : 42 (pour reproductibilité)

5.4.3 Interprétation

Les deux premières composantes principales capturent les patterns dominants de consommation électrique. La projection 2D facilite :

— La visualisation des groupes de comportements
— L'identification des observations atypiques
— L'analyse de la structure des données

Le modèle `pca.pkl` est sauvegardé pour transformer de nouvelles observations dans le même espace réduit.

5.5 Clustering DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) est un algorithme de clustering basé sur la densité, particulièrement adapté pour identifier des formes arbitraires et détecter les outliers.

5.5.1 Principe de DBSCAN

Contrairement à K-means qui requiert de spécifier le nombre de clusters à l'avance, DBSCAN :

— Identifie automatiquement le nombre de clusters
— Détecte les points de bruit (outliers, label = -1)
— Forme des clusters de densité variable
— Ne nécessite pas de forme sphérique des clusters

5.5.2 Hyperparamètres

Les paramètres utilisés sont :

Paramètre	Valeur	Description
`eps`	0.5	Rayon de voisinage
`min_samples`	10	Nombre minimum de points pour un cluster

TABLE 2 – Hyperparamètres DBSCAN

Ces valeurs ont été choisies empiriquement pour identifier les groupes de consommation normale tout en isolant les comportements atypiques.

6 Résultats

6.1 Visualisation

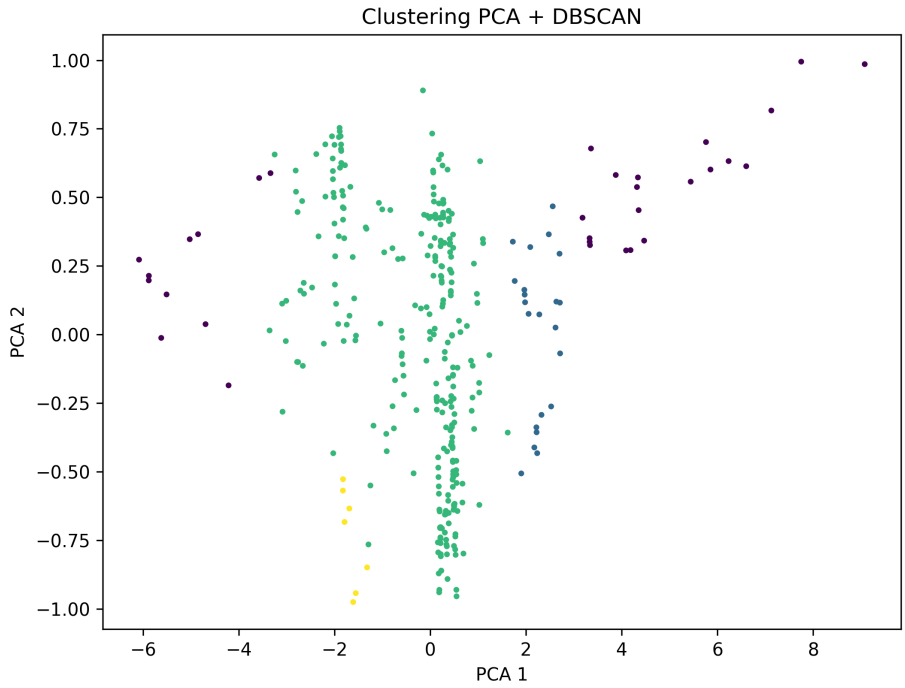

FIGURE 1 – Projection ACP des données avec clustering DBSCAN. Chaque couleur représente un cluster identifié. Les points violets (label = -1) correspondent aux outliers.

La Figure 1 montre la distribution spatiale des observations dans l'espace des deux premières composantes principales. On observe :

— Un **cluster central dominant** (vert, label = 1) : comportement normal majoritaire
— Des **clusters secondaires** (bleu et jaune) : variations de consommation spécifiques
— Des **outliers** (violet, label = -1) : comportements atypiques ou anomalies potentielles

6.2 Distribution des clusters

L'analyse a identifié les groupes suivants :

Label	Effectif	Interprétation
-1	31	Outliers / Comportements atypiques
0	22	Cluster secondaire 1
1	284	Cluster principal (consommation normale)
2	7	Cluster secondaire 2
Total	**344**	

TABLE 3 – Répartition des observations par cluster

6.3 Interprétation

— **Cluster 1 (284 observations, 82.6%)** : Représente le profil de consommation électrique normal et dominant. Ce groupe servira de référence principale pour le groupe G4.

— **Clusters 0 et 2 (29 observations, 8.4%)** : Variantes du comportement normal, possiblement liées à des périodes spécifiques (week-end, heures creuses, etc.).

— **Outliers (31 observations, 9.0%)** : Points jugés trop éloignés des zones denses. Ils nécessitent une analyse approfondie par le groupe G4 pour déterminer s'il s'agit d'anomalies réelles ou de cas particuliers légitimes.

7 Artefacts générés

Les artefacts suivants sont produits automatiquement dans le dossier `artifacts/` et sont essentiels pour les groupes suivants.

7.1 Liste des fichiers

Fichier	Contenu et utilité
`scaler.pkl`	Modèle RobustScaler entraîné. Permet de normaliser de nouvelles données selon les mêmes paramètres (médiane, IQR) que les données d'entraînement.
`pca.pkl`	Modèle ACP entraîné. Permet de projeter de nouvelles observations dans l'espace réduit à 2 dimensions défini lors de l'entraînement.
`dbscan_params.json`	Hyperparamètres DBSCAN (`eps=0.5`, `min_samples=10`). Permet de reproduire exactement le clustering ou d'ajuster les paramètres si nécessaire.
`clusters.json`	Distribution des observations par cluster. Fournit des statistiques sur la répartition des comportements identifiés.

TABLE 4 – Description des artefacts générés

7.2 Utilisation par les autres groupes

Ces artefacts sont utilisés par :

— **Groupe G4 (détection d'anomalies)** : Utilise `scaler.pkl` et `pca.pkl` pour transformer les nouvelles observations temps réel, puis compare leur position aux clusters normaux identifiés.

— **Groupes d'analyse** : Peuvent consulter `clusters.json` pour comprendre la distribution des comportements normaux.

8 Exécution du pipeline

Le pipeline G3 s'exécute de manière autonome une fois les pré-requis satisfaits.

8.1 Commande de lancement

```
python -m G3_data_mining.main
```

8.2 Flux d'exécution

Le script `main.py` orchestre les étapes suivantes :

1. **Extraction** : `fetch_historical_data()` récupère les données depuis PostgreSQL
2. **Normalisation** : `scale_data()` applique le RobustScaler
3. **ACP** : `apply_pca()` réduit les dimensions à 2
4. **Clustering** : `run_dbscan()` identifie les clusters
5. **Visualisation** : `plot_clusters()` génère le graphique
6. **Sauvegarde** : Tous les artefacts sont écrits dans `artifacts/`

8.3 Temps d'exécution estimé

Sur un ordinateur standard avec 100 000 observations :
— Extraction : 5–10 secondes
— Normalisation : <1 seconde
— ACP : <1 seconde
— DBSCAN : 2–5 secondes
— Visualisation : 1–2 secondes
— **Total : environ 10–20 secondes**

9 Justification des choix méthodologiques

9.1 Choix de RobustScaler

Le RobustScaler a été préféré au StandardScaler car il est moins sensible aux valeurs extrêmes, ce qui est essentiel pour préserver les anomalies potentielles que le groupe G4 devra détecter.

9.2 Réduction à 2 composantes principales

Le choix de 2 composantes permet :
— Une visualisation directe et interprétable des résultats
— Une transmission efficace des patterns au groupe G4
— Un compromis optimal entre simplification et conservation de l'information

9.3 Hyperparamètres DBSCAN

Les valeurs `eps=0.5` et `min_samples=10` ont été sélectionnées après tests empiriques pour :

— Identifier le cluster principal de consommation normale (82.6%)

— Isoler suffisamment d'outliers (9%) pour le groupe G4

— Éviter un sur-clustering qui fragmenterait le comportement normal

9.4 Échantillonnage à 100 000 observations

Cette limite garantit :

— Des temps d'exécution raisonnables (10-20 secondes)

— Une représentativité suffisante des comportements

— Une reproductibilité facile lors des démonstrations

10 Conclusion

Le groupe G3 fournit une modélisation claire et robuste des comportements normaux de consommation électrique, constituant une base solide pour la détection d'anomalies.

Notre pipeline a permis d'identifier avec succès :

— Un profil de consommation normal dominant (82.6% des observations)

— Des variations légitimes de comportement (8.4%)

— Des cas potentiellement anomaux (9.0%)

Les artefacts générés (`scaler.pkl`, `pca.pkl`, `dbscan_params.json`, `clusters.json`) sont prêts à être utilisés par le groupe G4 pour la détection d'anomalies en temps réel.

La méthodologie employée, combinant normalisation robuste, réduction dimensionnelle et clustering basé sur la densité, s'est révélée efficace pour analyser les patterns de consommation électrique sans supervision préalable.