**Credit Scoring Engine for Aave V2 Wallets (with Machine Learning)**

**This project computes a credit score (0–1000) for each DeFi wallet based on its historical interactions with the Aave V2 protocol. The scoring logic is powered by unsupervised machine learning (KMeans clustering) to identify wallet behavior patterns.**

**Objective**

**Build a transparent, extensible ML-based credit scoring system that:**

* **Analyzes raw DeFi transactions**
* **Extracts behavioural wallet features**
* **Scores users with a clustering-based method**

**Dataset**

* **File: user-wallet-transactions.json.zip**
* **Content: 100K transaction-level records from Aave V2**
* **Fields: wallet address, timestamp, action (deposit, borrow, repay, redeem, liquidation), token info**

**Architecture**

**Raw JSON Zip File**

**⬇️**

**Unzip + Load JSON**

**⬇️**

**Feature Engineering (per wallet)**

**⬇️**

**Normalize Features (StandardScaler)**

**⬇️**

**KMeans Clustering (5 clusters)**

**⬇️**

**Map cluster → credit score**

**⬇️**

**Save results as wallet\_credit\_scores.csv**

**🔍 Feature Engineering**

**For each wallet, we compute:**

| **Feature** | **Description** |
| --- | --- |
| **num\_deposits** | **Number of deposits** |
| **total\_deposit\_usd** | **Total value of deposits in USD** |
| **num\_borrows** | **Number of borrows** |
| **total\_borrow\_usd** | **Value of borrows in USD** |
| **num\_repays** | **Number of repayments** |
| **total\_repay\_usd** | **Value of repayments in USD** |
| **repay\_to\_borrow\_ratio** | **Repaid / Borrowed ratio** |
| **num\_liquidations** | **Liquidation events count** |
| **num\_redeems** | **Redeems made** |
| **total\_redeem\_usd** | **Redeem amount in USD** |
| **avg\_txn\_interval\_days** | **Avg time between transactions** |
| **active\_days** | **Unique number of active days** |
| **asset\_diversity** | **Number of distinct assets used** |
| **total\_tx\_count** | **Total transaction count** |

**Machine Learning Model**

**We use KMeans clustering to group wallet behaviors into 5 clusters. Features are normalized using StandardScaler. Each cluster is then mapped to a fixed credit score:**

| **Cluster ID** | **Assigned Credit Score** |
| --- | --- |
| **0** | **200** |
| **1** | **400** |
| **2** | **600** |
| **3** | **800** |
| **4** | **1000** |

**This creates a fully automated, data-driven wallet scoring system.**

**Output**

* **wallet\_credit\_scores.csv contains:**
  + **All extracted features**
  + **cluster column**
  + **credit\_score column (scaled 0–1000)**

**📂 Files**

* **wallet\_credit\_score.py – Main script (unzips, extracts features, clusters, scores)**
* **wallet\_credit\_scores.csv – Output scores for all wallets**
* **README.md – Documentation of method and pipeline**
* **analysis.md – Behavioural summary by score range**