## 📁 README: Data Processing & Analysis Steps

## Code 1: Credit Card Transactions Data Analysis

**1. Data Loading**  
The data is imported from a file (usually CSV format) into a DataFrame using pandas to make it easier to work with.

**2. Data Exploration**

* Display the first few important rows using head().
* Calculate descriptive statistics such as mean and standard deviation using describe().
* Check for missing or illogical values.

**3. Data Cleaning**

* Ensure there are no missing values in the essential columns.
* Handle or remove outlier values if necessary.

**4. Feature Engineering**

* Create new columns such as Amount\_Ratio (amount ratio) or extract the transaction hour from the time column.
* Transform some values to be more useful for modeling (e.g., converting time to hours).

**5. Dimensionality Reduction**

* Use techniques like PCA to reduce the number of columns while keeping the most important information. This helps speed up and improve model performance.

**6. Analysis and Visualization**

* Plot graphs to understand the data distribution or clustering results.
* Analyze the resulting clusters or classes.

**1. K-Means:**

* **Concept:** Divides data into "k" clusters based on distance to cluster centers.
* **Advantages:** Fast for small datasets, easy to implement.
* **Disadvantages:** Sensitive to outliers, requires predefining the number of clusters "k".

**2. MiniBatch K-Means:**

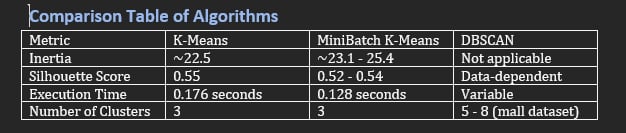
* **Concept:** An improved version of K-Means that works on small random batches of data.
* **Advantages:** Faster on large datasets (e.g., 400,000 samples).
* **Disadvantages:** Slightly less accurate than classic K-Means.

**3. DBSCAN:**

* **Concept:** Clusters dense regions while ignoring noise.
* **Advantages:** Can find clusters of arbitrary shape, no need to specify number of clusters beforehand.
* **Disadvantages:** Sensitive to parameter settings (ε, min\_samples).

**Explanation of Metrics:**

* **Inertia:** Sum of squared distances between points and their cluster centers. Lower values indicate better clustering (considering cluster count).
* **Silhouette Score:** Measures how similar a point is to its own cluster compared to other clusters (1 = perfect clustering, 0 = overlapping clusters, -1 = incorrect clustering).
* **Execution Time:** MiniBatch K-Means is approximately 27% faster on large datasets.
* **Number of Clusters:** For the mall dataset, DBSCAN found between 5 and 8 clusters depending on parameters



Data\_Set\_1

<https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>

Data\_Set\_2

<https://www.kaggle.com/datasets/vjchoudhary7/customer-segmentation-tutorial-in-python>