

Required Files:

- `Superstore.csv` (The dataset containing sales, profit, and discount information).

Libraries Used:

- `pandas`
- `random`
- `sklearn.preprocessing.LabelEncoder`
- `sklearn.cluster.KMeans`
- `matplotlib.pyplot`



Code Workflow

1. Data Preparation and Encoding

- **Loading:** The `Superstore.csv` file is loaded.
- **Cleaning:** Rows with missing values are removed (`dropna()`).
- **Encoding:** All categorical (`object`) columns are converted into numerical data using `LabelEncoder` so they can be processed by the K-Means and Genetic Algorithms.

2. Genetic Algorithm (GA) for Profit Optimization

The `GeneticAlgorithm` class is initialized to search for the best solution:

- **Objective:** Maximize total `Profit`.
- **Variables (Chromosome):**
 - `discount`: A float between 0 and 1.
 - `price`: A float between the minimum and maximum observed `Sales` values.
- **Fitness Function:** Calculates the sum of `Profit` for all data points where the actual discount is less than or equal to the candidate `discount` AND the actual sales value is greater than or equal to the candidate `price`.
- **Parameters:**
 - Population Size: 30
 - Generations: 60
 - Mutation Rate: 0.15

The output is the `best_solution` found, indicating the optimal price threshold and discount limit.

3. K-Means Clustering

- **Clustering:** K-Means is applied to the fully processed and encoded dataset with **3 clusters** (`n_clusters=3`).

- **Assignment:** The resulting cluster labels are added back to the dataset in a new column called 'Cluster'.

4. Visualization

- A scatter plot is generated to visualize the clustering results.
- **X-axis:** Sales
- **Y-axis:** Profit
- **Color:** The assigned Cluster label, allowing for visual inspection of the segmentation in the Sales-Profit space.