# **Decision Patterns: Analyzing Personal Finance Decisions Through Behavioral Clustering**

## **Project Overview**

This project aims to uncover hidden behavioral patterns in personal finance decisions through clustering techniques. By analyzing transaction data, we group users into clusters based on their spending behavior, such as rent, grocery, and entertainment expenses. This allows us to understand the financial behavior of different user groups and potentially make recommendations for financial planning.

### **Key Technologies:**

* **Data Cleaning & Feature Engineering**: Using Pandas and Numpy to preprocess and engineer features from raw transaction data.
* **Unsupervised Learning**: Implemented **KMeans clustering** to categorize users based on their spending habits.
* **Data Visualization**: Leveraged **Matplotlib** and **Seaborn** to create compelling visualizations for better understanding of clusters.
* **Streamlit App**: Built an optional interactive web app for users to predict their cluster based on input financial data.

## **Project Structure**

finance\_behavior\_clustering/

│

├── data/

│ └── transactions.csv # (dataset)

│

├── notebooks/

│ └── finance\_clustering.ipynb # (main analysis notebook)

│

├── app/

│ └── app.py # (optional Streamlit app)

│

├── README.md

│

├── requirements.txt

│

└── visuals/

└── cluster\_plot.png

└── spending\_patterns.png

## **Requirements**

Ensure you have the necessary dependencies by installing them via the requirements.txt file:

pandas

numpy

scikit-learn

matplotlib

seaborn

plotly

streamlit

## **Dataset**

The project uses a **Personal Finance Dataset** containing transaction data for individuals. The dataset can either be obtained from sources like [Kaggle](https://www.kaggle.com/datasets/ruchi798/million-dollar-personal-finance-dataset) or simulated with simple sample data. The columns in the dataset include:

* **user\_id**: Unique identifier for each user
* **transaction\_date**: Date of the transaction
* **transaction\_amount**: Amount spent on the transaction
* **transaction\_category**: Category of the transaction (e.g., Rent, Grocery, Entertainment)

Example of transactions.csv:

user\_id,transaction\_date,transaction\_amount,transaction\_category

1,2023-01-02,1200,Rent

1,2023-01-03,150,Grocery

1,2023-01-10,40,Entertainment

2,2023-01-04,1000,Rent

2,2023-01-07,200,Investment

2,2023-01-09,300,Grocery

...

## **Data Cleaning + Feature Engineering**

In the **finance\_clustering.ipynb** notebook, the following preprocessing steps are performed:

* Parsing the transaction date
* Aggregating data by user to compute the total spent, average transaction, and transaction count
* Pivoting the data to calculate the percentage of spending in different categories (e.g., Rent, Grocery, Entertainment)

df['transaction\_date'] = pd.to\_datetime(df['transaction\_date'])

df['month'] = df['transaction\_date'].dt.to\_period('M')

## **Clustering**

KMeans clustering is applied to identify patterns in financial behaviors:

* The dataset is normalized using StandardScaler.
* The **KMeans algorithm** is used to group users into clusters based on their financial spending patterns.

kmeans = KMeans(n\_clusters=4, random\_state=42)

clusters = kmeans.fit\_predict(X\_scaled)

## **Data Visualization**

Using **PCA (Principal Component Analysis)**, we reduce the dataset's dimensions to two for visualization, enabling us to plot the users in a 2D space:

pca = PCA(n\_components=2)

components = pca.fit\_transform(X\_scaled)

A scatter plot is created to visualize the different clusters:

sns.scatterplot(data=final\_df, x='pca\_1', y='pca\_2', hue='cluster', palette='Set2')

## **Insights**

After clustering, you can explore the characteristics of each cluster to derive insights:

* **Cluster 0**: Big spenders, heavy entertainment expenditure
* **Cluster 1**: Careful savers, high rent percentage

for c in final\_df['cluster'].unique():

display(final\_df[final\_df['cluster'] == c].describe())

## **Streamlit App (Optional)**

For a more interactive experience, we built a **Streamlit** app that allows users to input their financial details and predict which cluster they belong to. This gives users a personalized understanding of their financial behavior based on their spending.

import streamlit as st

# Prediction code for Streamlit app

if st.button('Predict My Cluster'):

X = scaler.transform([[total\_spent, avg\_transaction, transaction\_count, rent\_pct, grocery\_pct, entertainment\_pct]])

cluster = model.predict(X)

st.success(f'You belong to Cluster {cluster[0]}!')

## **README.md**

The README.md provides a complete guide for the project, explaining the data, methods used, and how to run the project, with detailed descriptions of the folder structure.

# Decision Patterns: Analyzing Personal Finance Decisions Through Behavioral Clustering

...