# Data Science Assignment eCommerce Transactions

## Objective

To Build a Lookalike Model that takes a user's information as input and recommends 3 similar customers based on their profile and transaction history.

## Dataset Description

Files Description:

1. Customers.csv
   - CustomerID: Unique identifier for each customer.
   - CustomerName: Name of the customer.
   - Region: Continent where the customer resides.
   - SignupDate: Date when the customer signed up.
2. Products.csv
   - ProductID: Unique identifier for each product.
   - ProductName: Name of the product.
   - Category: Product category.
   - Price: Product price in USD.
3. Transactions.csv
   - TransactionID: Unique identifier for each transaction.
   - CustomerID: ID of the customer who made the transaction.
   - ProductID: ID of the product sold.
   - TransactionDate: Date of the transaction.
   - Quantity: Quantity of the product purchased.
   - TotalValue: Total value of the transaction.
   - Price: Price of the product sold.

## Procedure

### 1. Importing Libraries

The necessary libraries are imported to handle various tasks:

- **Pandas**: Used for data manipulation and handling.
- **NumPy**: Used for numerical operations.
- **Matplotlib and Seaborn**: For visualization, though these are not used in the final code snippet.
- **Scikit-learn**: For machine learning and clustering, particularly KMeans, NearestNeighbors, StandardScaler, etc.
- **Datetime**: For working with dates.

## 2.Data Collection

The first step is collecting the data. The code assumes that the data is available in CSV format for three tables:

- **Customers.csv**: Contains customer-specific information such as CustomerID, SignupDate, and Region.
- **Products.csv**: Contains details about the products sold.
- **Transactions.csv**: Contains transactional data such as TransactionID, CustomerID, TotalValue, and TransactionDate.

## 3.Feature Engineering

a. Datetime Conversion

The columns SignupDate and TransactionDate are converted into datetime objects for easier date-based calculations.

b. Customer Aggregation

Aggregated transactional data at the customer level:

**TotalValue**: The total spending of each customer.
**Quantity**: The total quantity of items bought by each customer

we merge this aggregated data back with the customer-level information:

c. Recency, Frequency, and Monetary (RFM) Calculation

Next, the Recency, Frequency, and Monetary values (RFM) are calculated:

- **Recency**: The number of days since the most recent transaction for each customer.
- **Frequency**: The number of transactions each customer has made.
- **Monetary**: The total value of all transactions for each customer.

d. Normalization of RFM Values

To standardize the data, the RFM values are normalized using StandardScaler

# 4. Feature Engineering

Features are prepared for machine learning. For customer segmentation, the Region column is transformed using LabelEncoder to convert categorical values into numeric form, enabling its use in machine learning models.

# 5. Lookalike Model (Nearest Neighbors)

To find customers with similar behavior, we use the **Nearest Neighbors** algorithm. Here, the task is to find customers that are similar based on the features: Region, Recency, Frequency, and Monetary.

a. Nearest Neighbors

A `NearestNeighbors` model is created to find the nearest neighbors based on Euclidean distance.

b. Finding Lookalikes

The lookalikes are found for the first 20 customers.

# 6. Saving File

The lookalikes are saved in a CSV file, so they can be easily reviewed later.

| CustomerID | Lookalikes |
|---|---|
| C0001 | C0152-0.18,C0107-0.21,C0048-0.53 |
| C0002 | C0142-0.37,C0146-0.45,C0159-0.50 |
| C0003 | C0052-0.17,C0192-0.50,C0120-0.53 |
| C0004 | C0012-0.46,C0113-0.47,C0099-0.74 |
| C0005 | C0186-0.22,C0177-0.53,C0140-0.54 |
| C0006 | C0158-0.45,C0168-0.51,C0187-0.51 |
| C0007 | C0027-0.82,C0043-0.87,C0040-0.92 |
| C0008 | C0109-0.98,C0098-1.10,C0068-1.30 |
| C0009 | C0198-0.47,C0121-0.50,C0063-0.56 |

Table: Lookalikes for different customers

# Conclusion

- The **Nearest Neighbors** algorithm helps find customers who are most similar to others, which is the core idea behind creating a "lookalike" model.
- Feature engineering (like transforming categorical columns and normalizing numerical ones) is crucial to ensure that the model works correctly.
- The data is preprocessed and aggregated into meaningful metrics (RFM), which are used as inputs to the model.
- Finally, the results are saved to a CSV, which can be used to identify marketing opportunities or personalized customer engagement strategies.