RangeIndex: 4260904 entries, 0 to 4260903

1. 'cc\_num\_x' – CC Number
2. 'acct\_num' – Customer Account Number
3. 'trans\_num' – Customer Transaction Number
4. 'unix\_time' – Customer Transaction Time
5. 'category' – Category of Purchase
6. 'amt' – Transaction Amount made by the customer
7. 'is\_fraud' – Whether the transaction is fraud or not (0 if the transaction is fraud, 1 if the transaction is fraud)
8. 'merchant' – Merchant name
9. 'merch\_lat' – Merchant Latitude Location
10. 'merch\_long' – Merchant Longitude Location
11. 'ssn' – Customer SSN Number
12. 'first' – Customer First Name
13. 'last' – Customer Last Name
14. 'gender' – Customer Gender
15. 'street' – Customer Street Address
16. 'city' - Customer City
17. 'state' – Customer State
18. 'zip' – Customer Zipcode
19. 'lat' – Customer Latitude Location
20. 'long' – Customer Longitude Location
21. 'city\_pop' – Customer City population
22. 'job' – Customer Job description
23. 'dob' -Customer Date of Birth
24. 'trans\_date' – Customer Transaction Date
25. ‘seasons’ – Seasons in an year

Data columns (total 14 columns):

# Column Dtype

--- ------ -----

0 category int64

1 amt int64

2 is\_fraud int64

3 merch\_lat float64

4 merch\_long float64

5 ssn object

6 gender int64

7 city object

8 city\_pop int64

9 job object

10 dob datetime64[ns]

11 trans\_date datetime64[ns]

12 age int64

13 seasons int64

Pre-Processing

DataFrame is loaded from a CSV file and its information and missing values are checked. The 'dob' column is converted to datetime format, and the age is calculated based on the current date. The 'amt' and 'age' columns are then converted to the appropriate data types. Next, the categorical variable 'category' is mapped to numeric values using a predefined mapping dictionary. The 'trans\_date' column is converted to datetime format. Several unnecessary columns such as 'cc\_num\_x', 'acct\_num', and 'merchant' are dropped from the DataFrame to remove redundant information. The code also performs label encoding on the 'gender' column using the LabelEncoder from scikit-learn library. This step assigns numeric labels to the categorical gender variable for modeling purposes.

Exploratory Data Analysis

After the data preparation steps, exploratory data analysis is conducted to gain insights into the dataset. A scatter plot is created to visualize the relationship between age and transaction amount. For features with a small number of unique values (less than 25), the code prints the unique values. This step helps in identifying any potential issues or anomalies within the data.

Several cities stand out for having significant transaction volumes among the 726 cities in the dataset. Notably, the top cities with much larger transaction volumes than others include Brooklyn, Denver, Austin, Houston, and Dallas. With regard to the 'shopping\_pos' category, Brooklyn has the most transactions, indicating a significant customer presence. The category 'shopping\_pos' has the most transactions out of the 14 available transaction types, demonstrating a strong preference for retail purchases. Contrarily, the analysis shows that customers typically spend less on transactions involving travel, identifying a potential area for additional research and focused marketing efforts.

Interesting insights are revealed by further research of gender-based transaction patterns. According to the research, men spend a significant amount of money on gas and other transportation-related expenses, underscoring the significance of this category in their spending patterns. Females, on the other hand, show a stronger propensity to engage in food shopping activities, suggesting their attention to home necessities. Seasonal patterns in transaction activity also offer important insights. It is noteworthy that the data shows that the Summer season saw the most transactions, mostly fueled by purchases in the Food & Dining category. This finding implies that people have a stronger a tendency to go out to eat or engage in food-related activities in the summer. On the other hand, Autumn shows the lowest number of transactions, suggesting that overall consumer spending may slow down during this season.

Modelling

After cleaning and preparing the data, the next step in the CRISP-DM methodology is the modeling phase, where it focuses on applying the K-means clustering algorithm to segment the data based on age and transaction amount. To determine the optimal number of clusters for K-means algorithm, we evaluate the sum of squared error (SSE) for different values of K. SSE measure the within clusters variation and helps identify the number of clusters that best explain the data’s underlying structure.