**E-COMMERCE USER BEHAVIOR ANALYSIS REPORT**

**Content**

[1. Project Background and Objectives 3](#_Toc7415)

[1.1 Background 3](#_Toc9000)

[1.2 Objectives 3](#_Toc21815)

[2. Data Sources and Preprocessing 4](#_Toc12145)

[2.1 Data Sources 4](#_Toc28636)

[2.2 Data Cleaning and Preprocessing 4](#_Toc26618)

[Data Cleaning and Preprocessing 4](#_Toc12660)

[3. Methodology and Technical Implementation 4](#_Toc26042)

[3.1 Technical Framework 4](#_Toc25657)

[3.2 Core Algorithms and Models 4](#_Toc28278)

[4. Results Analysis and Visualization 5](#_Toc30594)

[4.1 Analytical Results 5](#_Toc31299)

[4.2 Visualization Outputs 5](#_Toc6336)

[5. Summary and Recommendations 5](#_Toc1977)

[5.1 Project Summary 5](#_Toc24429)

[5.2 Suggestions For Improvement 5](#_Toc7844)

[6.  Submission Materials Checklist 6](#_Toc12666)

[7. References (Optional) 6](#_Toc23867)

[8. Appendix 6](#_Toc27678)

[8.1 Code Structure Documentation 6](#_Toc3120)

[8.2 Dataset Description 7](#_Toc31801)

# 1. Project Background and Objectives

## 1.1 Background

In the competitive e-commerce landscape, understanding customer behavior is crucial for businesses to thrive. By analyzing transaction data, companies can identify different customer segments, understand their purchasing patterns, and tailor marketing strategies effectively. This project focuses on leveraging customer transaction data to perform **Recency, Frequency, Monetary (RFM) analysis** and **K-Means clustering** to segment customers. The insights gained can help in developing targeted marketing campaigns, improving customer retention, and maximizing customer lifetime value, which are essential for sustainable business growth.

## 1.2 Objectives

The core objectives of this project, as implemented in the Python notebook, are:

* To load and preprocess a large e-commerce customer transaction dataset.
* To calculate RFM (Recency, Frequency, Monetary) values for each customer.
* To perform customer segmentation based on RFM scores using rule-based methods.
* To apply the K-Means clustering algorithm on RFM metrics to identify distinct customer segments in a more data-driven manner.
* To analyze and interpret the characteristics of these customer segments.
* To visualize the RFM distributions and customer segments.
* To propose targeted marketing strategies for each identified customer segment to enhance engagement and profitability.

# 2. Data Sources and Preprocessing

# 2.1 Data Sources

The dataset used is the **Taobao User Behavior public dataset**, which includes approximately 100 million records. Due to performance and resource constraints, a **sample of 100,000 rows** was used for this project. The dataset includes:

* user\_id: Unique ID of each user
* item\_id: Unique ID of each item
* item\_category: Category ID of the item
* behavior\_type: User action (click, cart, purchase, collect)
* timestamp: Unix timestamp of the behavior

# 2.2 Data Cleaning and Preprocessing

Handling missing values/outliers

Data format standardization (e.g., timestamp conversion)

Feature engineering (e.g., constructing RFM metrics)

### Handling Missing Values and Outliers

To ensure data quality, the dataset was examined for missing values using the isnull().sum() method. No missing values were detected in any of the five columns (user\_id, item\_id, item\_category, behavior\_type, timestamp), so no rows were removed at this stage.

Outliers were handled by analyzing the distribution of purchase frequencies. Specifically, after calculating the number of purchases (frequency) per user, the 99th percentile was used as a threshold to identify extremely high-frequency users. These users were considered statistical outliers and were excluded from further analysis to avoid skewing the clustering results. This filtering was done using the following logic:

outliers = rfm\_data['frequency'].quantile(0.99)

rfm\_data = rfm\_data[rfm\_data['frequency'] < outliers]

### Data Format Standardization

The original dataset recorded behavioral events using Unix timestamps, which are not human-readable. These were converted to standard datetime format using the pandas.to\_datetime() function. This conversion allowed for accurate time-based calculations, particularly in computing the recency metric in the RFM model.

userbehavior['datetime'] = pd.to\_datetime(userbehavior['timestamp'], unit='s')

### Feature Engineering

To perform user segmentation, an RFM (Recency, Frequency, Monetary) model was constructed:

* **Recency**: Calculated as the number of days between the most recent purchase and the latest date in the dataset.
* **Frequency**: Measured as the total number of purchase actions per user.
* **Monetary**: Since actual spending amounts were not available, the purchase frequency was used as a proxy for user monetary value.

These metrics were derived only from records where behavior\_type == 'purchase'. After computing the RFM values, they were standardized using the StandardScaler function from scikit-learn to ensure that all three dimensions contributed equally to clustering.

# 3. Methodology and Technical Implementation

## 3.1 Technical Framework。

The project was implemented in **Python** using the following libraries:

* pandas, numpy: Data manipulation and statistics
* matplotlib, seaborn: Data visualization
* sklearn: Clustering (KMeans), scaling (StandardScaler), and evaluation (silhouette score)
* colorama: Colored CLI output for easier interpretation
* Jupyter Notebook: Development environment

## 3.2 Core Algorithms and Models

**Model Selection:**

* **RFM Analysis:** A widely used marketing model for customer segmentation based on their transaction history (Recency, Frequency, Monetary value).
* **K-Means Clustering:** An unsupervised machine learning algorithm used to partition data points into 'K' distinct, non-overlapping clusters based on feature similarity. In this project, it's applied to the RFM metrics.

### Implementation Steps

* **Data Loading and Initial Exploration**  
  The dataset UserBehavior.csv was loaded with predefined column names and data types to optimize memory usage. A 100,000-rows sample was selected from the full dataset for practical computation. Basic exploration functions such as head() and info() were used to validate the structure.
* **Data Cleaning**  
  The timestamp column was converted from Unix time to Python datetime format using pd.to\_datetime() for further time-based analysis. No missing values were found in the dataset, so no imputation or deletion was needed. Outliers in the frequency field were removed using the 99th percentile threshold to reduce skew.
* **RFM Metrics Calculation**  
  A filtered subset of data containing only 'purchase' actions was created. The following metrics were calculated for each user\_id:
  + **Recency**: Days since the most recent purchase.
  + **Frequency**: Count of purchases.
  + **Monetary**: Purchase frequency used as a proxy due to lack of transaction value data.  
    The resulting DataFrame was named rfm\_data.
* **Data Normalization**  
  The recency, frequency, and monetary features were scaled using StandardScaler from scikit-learn, resulting in rfm\_normalized.
* **Determining Optimal K with Silhouette Score**  
  A loop was implemented to calculate silhouette scores for k = 2 to k = 9, using:
* silhouette\_score(rfm\_normalized, model.fit\_predict(rfm\_normalized))

This helped assess clustering quality and guided the final cluster count.

* **KMeans Clustering**  
  Based on silhouette score trends, **K = 4** was selected. A KMeans model was initialized with n\_clusters=4 and fitted to the normalized data. Cluster labels were assigned to each user and merged back into the original rfm\_data DataFrame.
* **Cluster Profiling and Segment Naming**  
  Average values of Recency, Frequency, and Monetary were calculated for each cluster to interpret user types. A manual mapping was then applied:
* cluster\_labels = {
* 0: 'VIP',
* 1: 'Churn Risk',
* 2: 'Loyal',
* 3: 'At\_Risk'
* }

These labels helped clearly define the business meaning of each cluster.

* **Visualization**  
  Several visualizations were created using matplotlib and seaborn:
  + Boxplots of R, F, M by cluster
  + Bar charts of average RFM values
  + Pie charts for user distribution by cluster and by segment
  + 2D scatter plots for Recency vs Frequency and Frequency vs Monetary
  + Annotated bar plots showing the count of users per segment

# 4. Results Analysis and Visualization

## 4.1 Analytical Results

### Quantitative Results

* **RFM Value Distributions**:  
  Histograms and boxplots were used to visualize the distributions of recency, frequency, and monetary values. These revealed that many users had low purchase frequency and monetary scores, with only a small portion demonstrating high value in all three dimensions.
* **Optimal Number of Clusters (K)**:  
  The silhouette score was calculated for K values from 2 to 9. Based on the highest silhouette score, **K = 4** was chosen as the most suitable number of clusters:
* for k in range(2, 10):
* model = KMeans(n\_clusters=k, random\_state=42)
* labels = model.fit\_predict(rfm\_normalized)
* score = silhouette\_score(rfm\_normalized, labels)
* print(f"k = {k}, silhouette score = {score:.4f}")
* **KMeans Cluster Centroids/Profiles**:  
  After fitting the KMeans model with K=4, the mean recency, frequency, and monetary values for each cluster were computed using:
* cluster\_summary = rfm\_data.groupby('cluster').agg({
* 'recency': 'mean',
* 'frequency': 'mean',
* 'monetary': 'mean',
* 'user\_id': 'count'
* }).rename(columns = {'user\_id': 'num\_user'})
* **Segment Profiles Based on RFM Centroids**:
  + **Cluster 0 (VIP)**: Very low recency, high frequency, high monetary — top-spending and loyal users.
  + **Cluster 1 (Churn Risk)**: High recency, moderate frequency and monetary — haven’t purchased in a while but were active before.
  + **Cluster 2 (Loyal)**: Moderate recency, good frequency and monetary — engaged and consistent buyers.
  + **Cluster 3 (At\_Risk)**: High recency, low frequency and monetary — likely to churn, low activity.
* **Segment Sizes**:  
  A pie chart and bar graph were used to display the number of users in each segment, showing the proportion of users across all four clusters. These visualizations provided insight into how users are distributed across value tiers.

### Qualitative Conclusions

Based on KMeans clustering with K=4, users were segmented into four actionable groups:

* **VIP (High Value)**:
  + **Profile**: Very recent activity, frequent purchases, high monetary value
  + **Strategy**: Prioritize with premium loyalty programs, early access offers, and personalized messages.
* **Loyal (Medium-High Value)**:
  + **Profile**: Recent and consistent activity, above-average frequency and spend
  + **Strategy**: Send tailored product recommendations and maintain engagement with personalized incentives.
* **Churn Risk (Medium-Low Value)**:
  + **Profile**: Active in the past, but haven’t returned recently
  + **Strategy**: Run win-back email campaigns, offer time-limited discounts or reminders.
* **At\_Risk (Low Value)**:
  + **Profile**: Rare and outdated activity, low spend
  + **Strategy**: Cost-effective re-engagement campaigns, bundled offers, or retargeting ads.

These segments were visualized with pie charts and bar graphs, and each group was given a descriptive name for easier business interpretation. This segmentation provides the foundation for **personalized, data-driven marketing strategies** to increase retention and sales.

## 4.2 Visualization Outputs

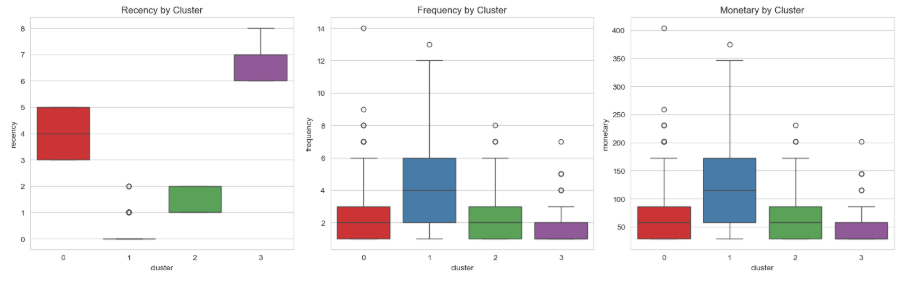
## Visualization Outputs

### Chart Types & Tools

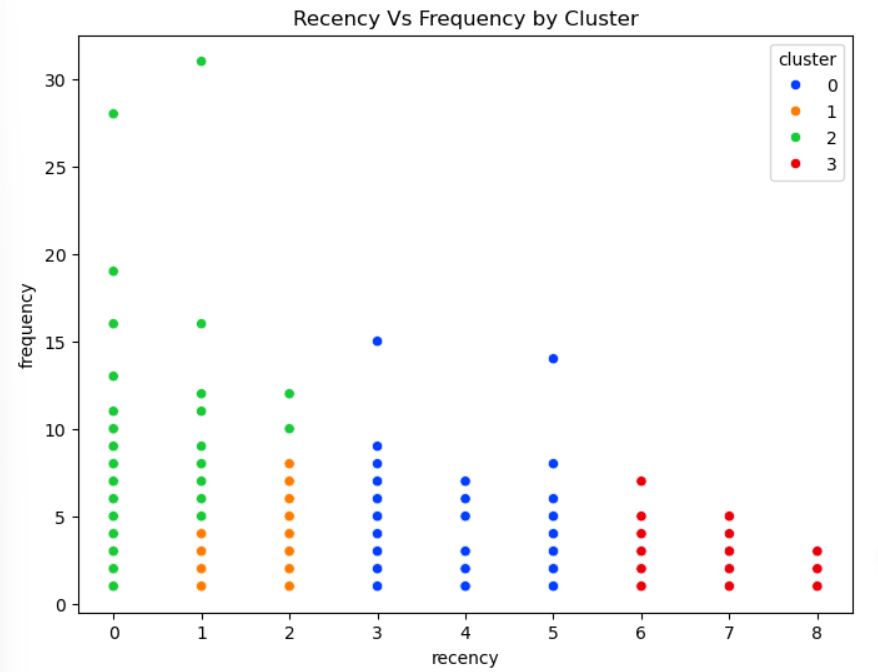
All visualizations were created using **Matplotlib** and **Seaborn** within a **Jupyter Notebook**. These tools provided clear and interpretable charts for analyzing RFM distributions, clustering results, and customer segments.

### Plots and Visual Insights:

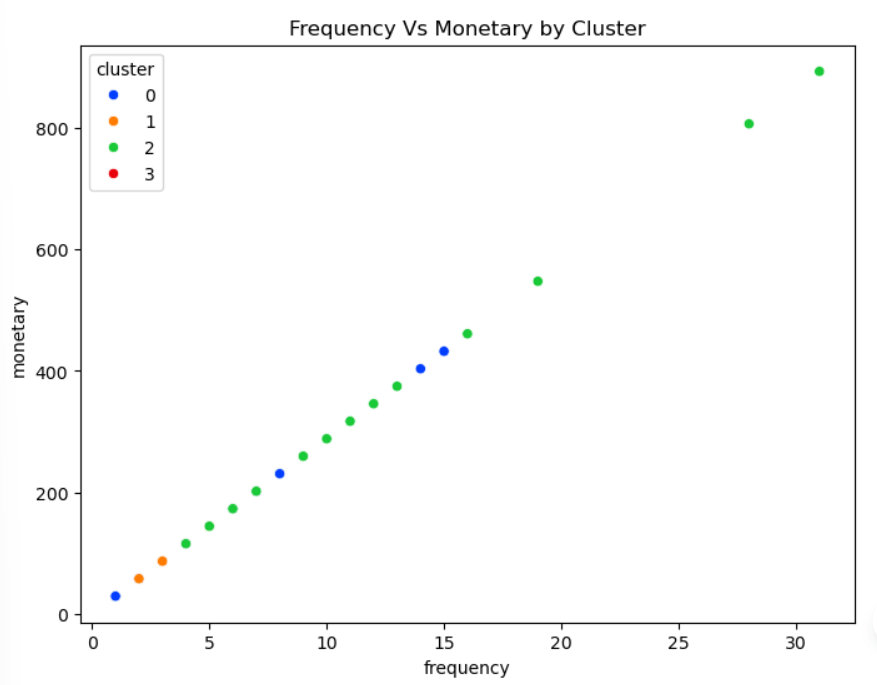
* **Boxplots**  
  Boxplots were used to compare the distributions of **Recency**, **Frequency**, and **Monetary** values across different KMeans clusters.
  + **Figure 1**: Boxplot of Recency by Cluster
  + **Figure 2**: Boxplot of Frequency by Cluster
  + **Figure 3**: Boxplot of Monetary by Cluster



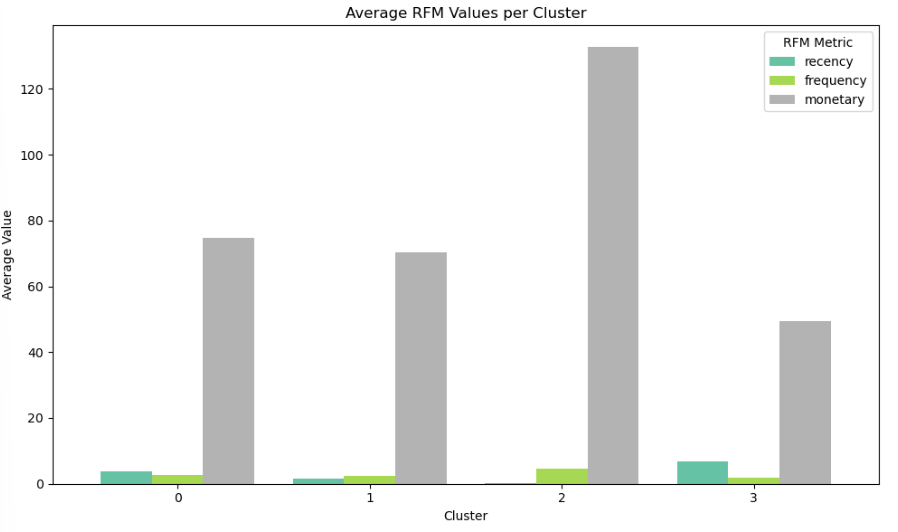
* **2D Scatter Plots**  
  To visually explore how customers are grouped by KMeans, colored scatter plots were created:
  + **Figure 4**: Recency vs. Frequency by Cluster (with hue='cluster')



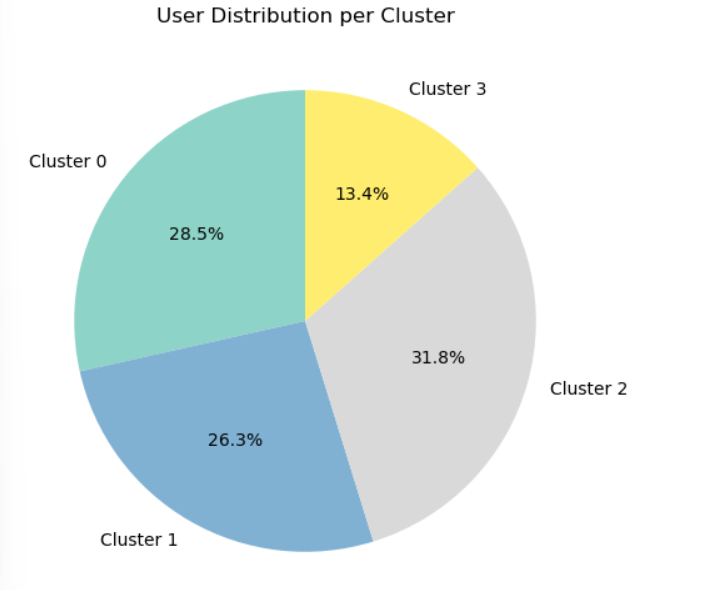
* + **Figure 5**: Frequency vs. Monetary by Cluster



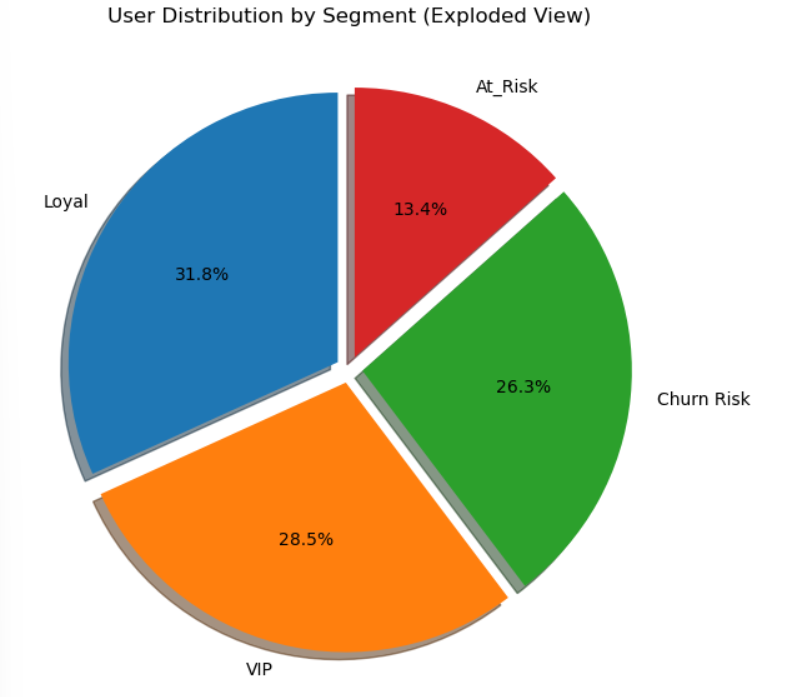
* **Cluster Summary Bar Chart**  
  A bar chart was generated to compare average R, F, and M values for each cluster, highlighting behavior differences:
  + **Figure 6**: Average RFM Values per Cluster



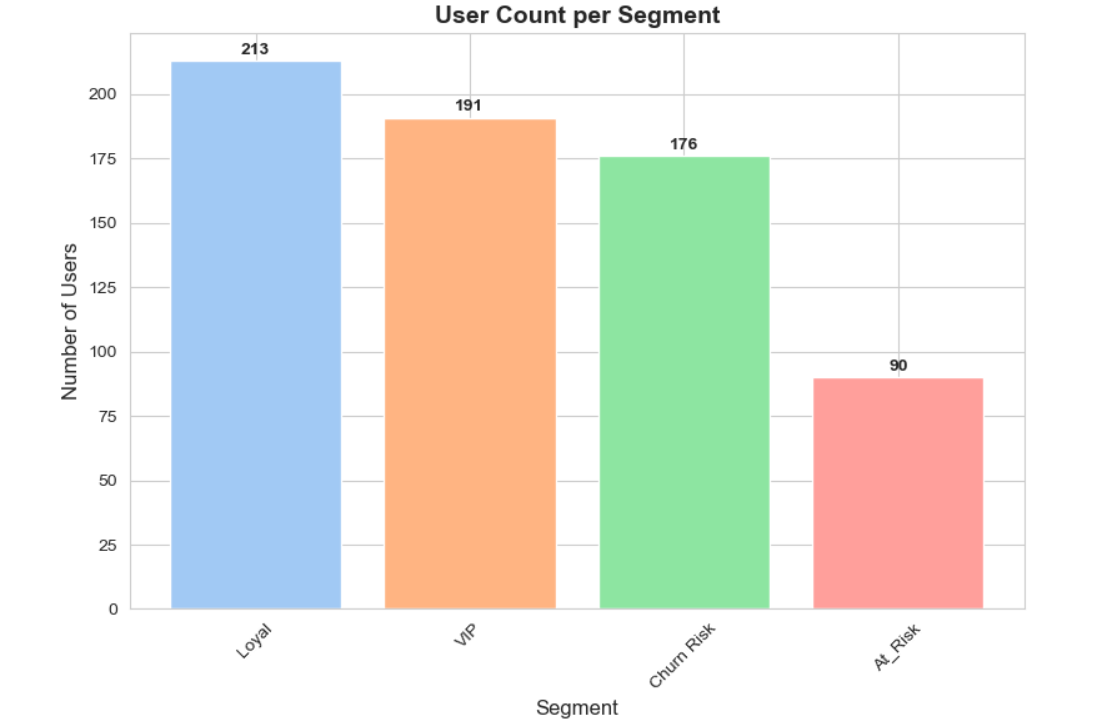
* **Pie Charts**  
  Pie charts showed customer distribution:
  + **Figure 7**: User Distribution per Cluster



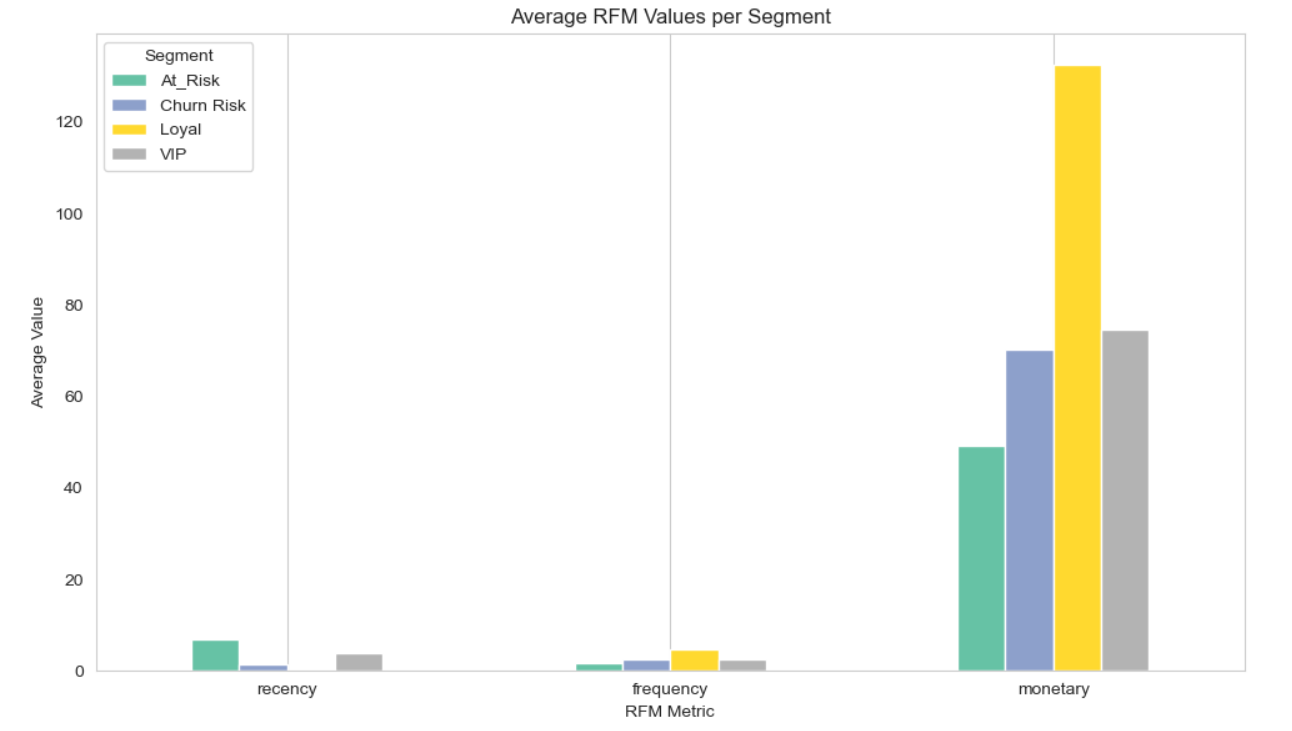
* + **Figure 8**: User Distribution by Segment (after assigning descriptive labels like VIP, Loyal, etc.)



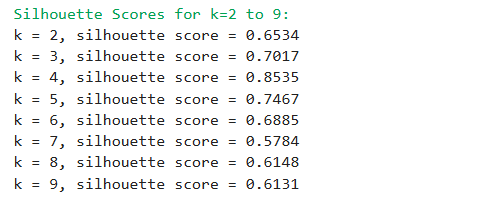
* **Segment Count Bar Plot**  
  A pastel-colored bar chart showed the number of users per segment (VIP, Loyal, Churn Risk, At\_Risk). Each bar was annotated with the exact user count.
  + **Figure 9**: User Count per Segment



* **Segment Summary RFM Plot**  
  A grouped bar plot was used to compare **average Recency, Frequency, and Monetary values** per segment.
  + **Figure 10**: Average RFM Values by Segment

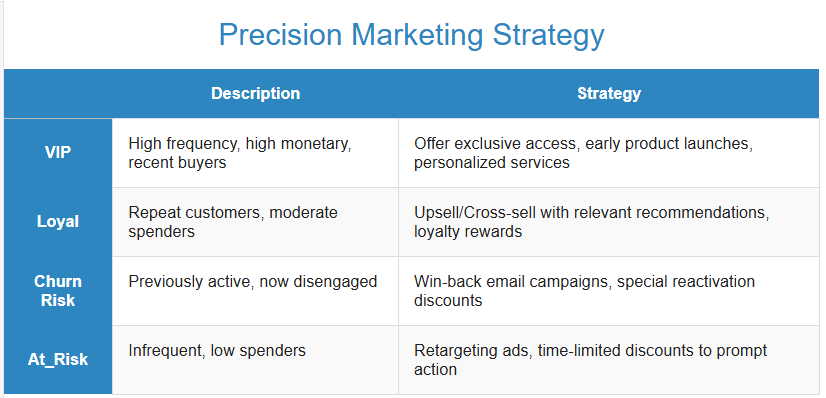


* **Silhouette Score Evaluation**  
  Although not plotted visually, a loop was run to compute and display silhouette scores for k = 2 to 9, helping to determine the optimal number of clusters. The best score was found at k=4.



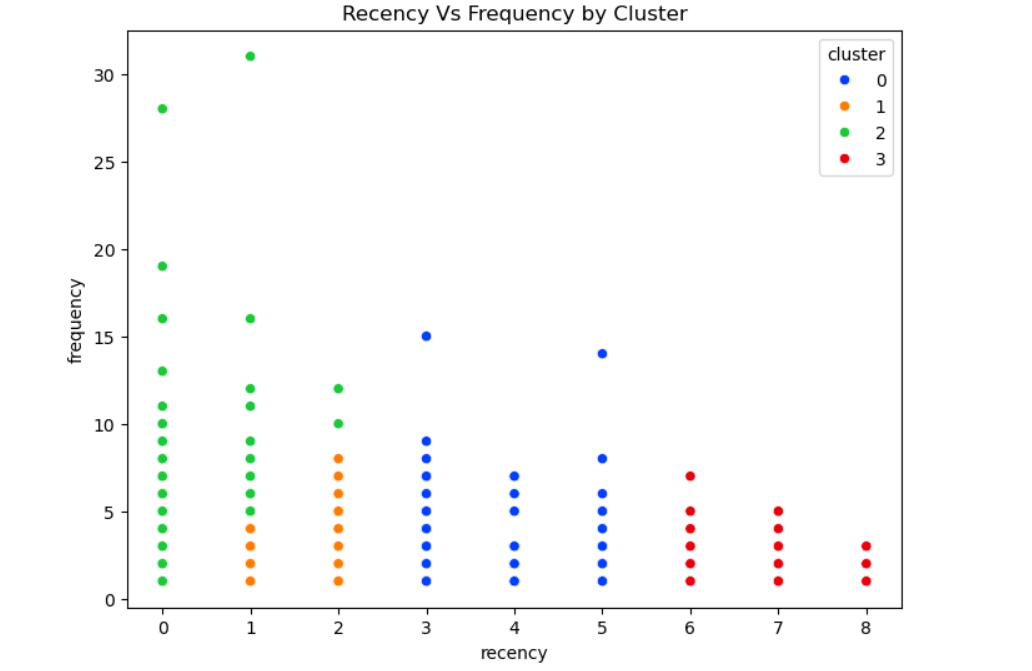
### Summary Table (Conceptual)

* Cluster labels were mapped to descriptive names (VIP, Loyal, Churn Risk, At\_Risk).
* Each segment was tied to actionable marketing strategies.
* **Table 1**: Segment Descriptions, RFM Characteristics, and Strategic Recommendations



### Example Annotation

**Figure 4: Recency vs. Frequency by Cluster**  
This plot shows how users naturally group into four distinct segments. For example, **Cluster 0 (VIPs)** typically have **low Recency and high Frequency**, indicating frequent, recent purchasers.



# 5. Summary and Recommendations

## 5.1 Project Summary

Summarize achievements and challenges (e.g., data scraping difficulties, model tuning).

This project applied RFM analysis and K-Means clustering to perform customer segmentation for an e-commerce platform, using behavioral data from Taobao. The main achievements of the project include:

* Completion of data preprocessing, including timestamp conversion, filtering for purchase behavior, and calculation of Recency, Frequency, and Monetary metrics for each user.
* Implementation of a clustering approach using K-Means on standardized RFM features to segment users based on their purchasing behavior.
* Evaluation of clustering quality using silhouette scores for different values of k, with four clusters selected as optimal based on the score performance.
* Generation of a summary of cluster profiles by calculating the average RFM values within each cluster and interpreting behavioral patterns across segments.
* Assignment of descriptive labels to each cluster, namely: VIP, Loyal, Churn Risk, and At\_Risk, for practical business interpretation.
* Development of specific marketing strategies tailored to each segment’s behavioral characteristics.
* Use of Python tools and libraries, including pandas, scikit-learn, matplotlib, and seaborn, for data analysis, modeling, and visualization within a Jupyter Notebook environment.

**Challenges encountered during the project included:**

* Managing the large volume of data and maintaining efficient performance during processing and analysis (original file was over 3.4GB).
* Interpreting silhouette score trends to identify the most appropriate number of clusters in the absence of a clearly defined elbow point.
* No real monetary values in the dataset, requiring estimation.

## 5.2 Suggestions For Improvement

Based on the clustering results from the RFM analysis and K-Means model, users were segmented into four distinct behavioral groups: **VIP**, **Loyal**, **Churn Risk**, and **At\_Risk**. The following actionable strategies are proposed to help guide marketing, retention, and engagement efforts:

### 1. VIP (High-Value Users)

**Profile**: Recently active users with high purchase frequency and high monetary value.  
**Recommended Actions**:

* Send **personalized coupons** and **thank-you messages** to reinforce loyalty.
* Offer **early access to exclusive products**, **priority support**, and **VIP-only events**.
* Avoid unnecessary discounts; focus instead on **recognition and exclusive privileges**.

### 2. Loyal (Medium-to-High Value Users)

**Profile**: Consistent purchasing behavior, good frequency and monetary value, recent activity.  
**Recommended Actions**:

* Provide **product recommendations** based on purchase history.
* Encourage **membership enrollment** and offer **loyalty points or rewards** for continued engagement.
* Promote **referral incentives** to leverage their satisfaction and advocacy.

### 3. Churn Risk (Medium-to-Low Value Users)

**Profile**: Users who previously engaged but have not made recent purchases.  
**Recommended Actions**:

* Trigger **automated re-engagement emails** with **limited-time discounts** or **reminders** of previous purchases.
* Offer **“we miss you” promotions** and track response rates to adjust strategies.
* Collect feedback to understand why they stopped purchasing.

### 4. At\_Risk (Low-Value Users)

**Profile**: Inactive users with low frequency and monetary value.  
**Recommended Actions**:

* Send **welcome-back offers** or **seasonal discounts** to reactivate them.
* Use **cost-effective advertising** (e.g., retargeting ads on social media).
* Focus efforts on scalable campaigns rather than personalized outreach due to lower ROI.

### Overall Strategic Recommendations:

* **Segment-Based Personalization**: Integrate user segments into CRM and marketing platforms to automate targeted communication.
* **Resource Allocation**: Prioritize spending and personalization efforts on VIP and Loyal segments where the return on investment is highest.
* **Continuous Monitoring**: Update RFM segmentation regularly to reflect evolving user behavior.
* **Campaign Testing**: Perform A/B testing within each segment to optimize message types, timing, and offers.

These recommendations are directly informed by the quantitative and qualitative findings from the RFM clustering analysis and aim to support precise, data-driven decision-making in marketing strategy.

# 6. References

Cite relevant papers, technical documentation, or tools (e.g., Prophet documentation, library).

* Fader, P. S., Hardie, B. G., & Lee, K. L. (2005). "RFM and CLV: Using iso-value curves for customer base analysis." *Journal of marketing research*, 42(4), 415-430.
* Scikit-learn official documentation for KMeans: <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>
* Pandas official documentation: <https://pandas.pydata.org/pandas-docs/stable/>
* Seaborn official documentation: <https://seaborn.pydata.org/>

# 7. Appendix

## 7.1 Code Structure Documentation

Directory tree of code files

**Project\_Root/**

├── E-commerce\_User\_Behavior\_Analysis\_Dashboard.py # Interactive Dashboard with all python code.

└── E-commerceUserBehaviorAnalysis.ipynb # Jupyter Notebook with all Python code.

└── E-commerceUserBehaviorAnalysis\_Report.docx # Project Report.

└── Taobao\_Executive\_Summary.pptx # Summary of finding.

└── **UserBehavior.zip/**

└──UserBehavior.csv # Dataset file. #But Not included

└── README

Key function descriptions

### 1. ****Initial Setup & Library Imports****

This block imports essential Python libraries used for data analysis, clustering, and visualization:

* pandas, numpy for data manipulation
* matplotlib.pyplot, seaborn for data visualization
* sklearn.preprocessing.StandardScaler and sklearn.cluster.KMeans for scaling and clustering
* sklearn.metrics.silhouette\_score for evaluating clustering performance
* colorama for enhanced terminal output formatting

### 2. ****Data Loading Block****

Loads a 100,000-row sample from the UserBehavior.csv dataset using appropriate data types and column names for efficiency. The dataset includes user behavior logs (click, cart, collect, purchase) with Unix timestamps.

### 3. ****Data Cleaning & Preprocessing Block****

* Converts timestamp to datetime format using pd.to\_datetime()
* Filters only 'purchase' records for RFM analysis
* Groups data by user\_id to compute purchase-related behavior
* Drops outliers from the frequency column using the 99th percentile threshold

### 4. ****RFM Calculation Block****

* Calculates:
  + **Recency**: Days since last purchase relative to the most recent date in the dataset
  + **Frequency**: Count of purchases by each user
  + **Monetary**: Estimated via purchase count (as actual purchase amount is unavailable)
* Constructs the rfm\_data DataFrame containing all three metrics

### 5. ****Feature Scaling Block****

* Applies StandardScaler to the recency, frequency, and monetary columns
* Produces the normalized dataset rfm\_normalized for clustering

### 6. ****Silhouette Evaluation Block****

* Evaluates multiple values of k (2 to 9) for KMeans clustering
* Computes and prints silhouette scores to determine the optimal number of clusters

for k in range(2, 10):

model = KMeans(n\_clusters=k)

labels = model.fit\_predict(rfm\_normalized)

score = silhouette\_score(rfm\_normalized, labels)

print(f"k = {k}, silhouette score = {score}")

### 7. ****K-Means Clustering Block****

* Initializes and trains the KMeans model with n\_clusters=4
* Assigns cluster labels to each user
* Appends cluster results back to the original rfm\_data DataFrame

### 8. ****Cluster Profiling & Segment Labeling Block****

* Aggregates mean RFM values for each cluster using groupby()
* Interprets cluster characteristics and manually maps them to business-friendly labels:
  + Cluster 0: VIP
  + Cluster 1: Churn Risk
  + Cluster 2: Loyal
  + Cluster 3: At\_Risk
* Adds a segment column to label each user accordingly

### 9. ****Visualization Blocks****

Generates visual insights using matplotlib and seaborn:

* Boxplots of Recency, Frequency, and Monetary by cluster
* 2D scatter plots:
  + Recency vs Frequency
  + Frequency vs Monetary
* Pie charts:
  + User count per cluster
  + User distribution by segment
* Bar plots of average RFM values by cluster and by segment
* Annotated bar chart showing user count per segment

### 10. ****Segment Summary Table Block****

* Groups RFM metrics by segment and transposes the result
* Displays the data in a bar plot comparing average RFM values for each user segment

## 7.2 Dataset Description

### Dataset Field Definitions

The dataset used in this project is the publicly available **Taobao User Behavior Dataset**, which records user interaction logs on the Taobao e-commerce platform. It consists of five key fields:

* **user\_id**: A unique identifier for each user. The IDs have been anonymized.
* **item\_id**: A unique identifier for each item (product) involved in user interactions.
* **category\_id**: A unique identifier representing the product category.
* **behavior\_type**: The type of user behavior. It includes one of four actions: "pv" (page view), "cart" (add to cart), "fav" (add to favorites), and "buy" (purchase).
* **timestamp**: A Unix timestamp representing the time the behavior occurred.

### Data Scale

The full dataset contains over **100 million** rows and **5 columns**, representing a large-scale behavioral log of user interactions on Taobao over a period of **four months** (from **November 2017 to March 2018**).

For the purpose of this project, a **sample of 100,000 rows** was extracted for efficient processing and analysis, while still maintaining a representative distribution of user behaviors.