Bank Customer Segmentation using Big Data

- Module 4 Final Project – ALY6110
- Professor Olesya Agafontseva
- Group 6 – Payal Sharma, Min Zeng, Hang Chen, Yuhong Tao
- June 11, 2025

Project Overview

- **Background**

In the era of digital banking, understanding customer behavior is critical to driving personalized services and improving financial decision-making. Banks collect massive transaction data daily, which presents a valuable opportunity for customer segmentation, targeting, and fraud detection.

- **Business Problem**

How can we better understand customer behavior and segment bank customers to improve product targeting, detect transactional anomalies, and enhance service delivery across regions?

- **Project Goal**

Leverage big data, machine learning, and graph analytics to uncover meaningful patterns in customer demographics, transaction behavior, and banking networks. These insights will support more effective marketing, fraud detection, risk management, and customer service strategies.

Dataset Overview

- Source: Kaggle – Bank Customer Segmentation (1M+ Transactions)
- Size: Over 1 million transaction records
- Key Fields: Age, Gender, Account Balance, Transaction Amount, Location
- Tools Used: Python (Pandas, Seaborn, Matplotlib, Scikit-learn)

Data Cleaning

Variable Description

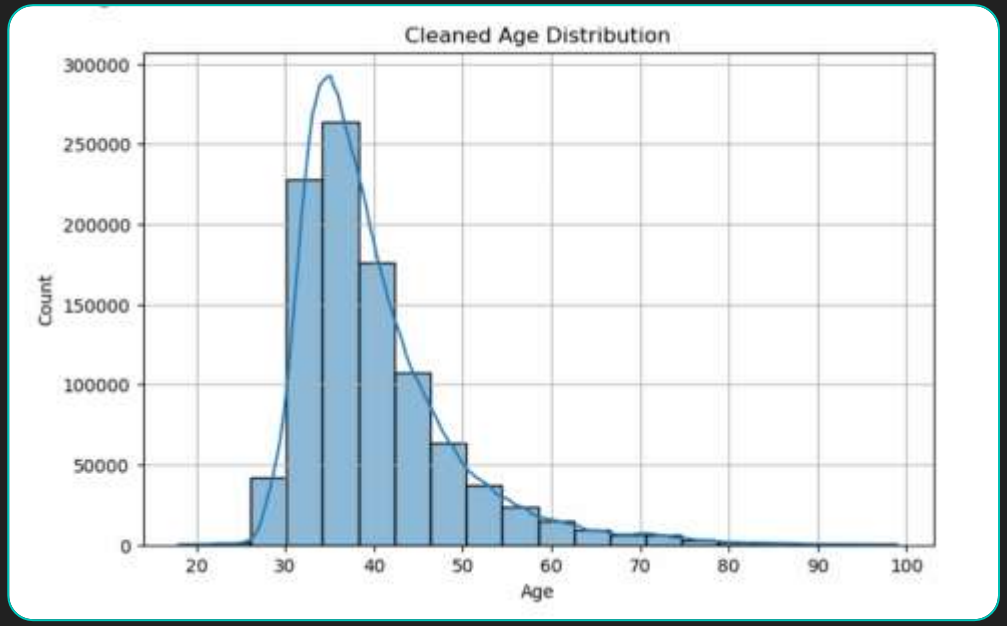
Column Name	Data Type	Description
`TransactionID`	object (str)	Unique ID assigned to each transaction.
`CustomerID`	object (str)	Unique ID assigned to each customer.
`CustomerDOB`	object (date)	Date of birth in `DD/MM/YY` format. Can be used to derive customer age.
`CustGender`	object (str)	Customer gender (`M` = Male, `F` = Female).
`CustLocation`	object (str)	Customer's city/location.
`CustAccountBalance`	float	Current balance of the customer's bank account (in INR).
`TransactionDate`	object (date)	Date of transaction in `DD/MM/YY` format.
`TransactionTime`	int	Time of transaction in HHMMSS format (e.g., 143207 = 14:32:07).
`TransactionAmount (INR)`	float	Amount of the transaction in Indian Rupees (INR).

Notes:

- You can derive `Age` from `CustomerDOB`.
- `TransactionTime` can be converted into `Hour`, `Minute`, or time ranges for time-series analysis.
- Consider encoding `CustGender` and `CustLocation` for modeling tasks.
- Recommended checks include missing values and outlier detection for `TransactionAmount` and `CustAccountBalance`.

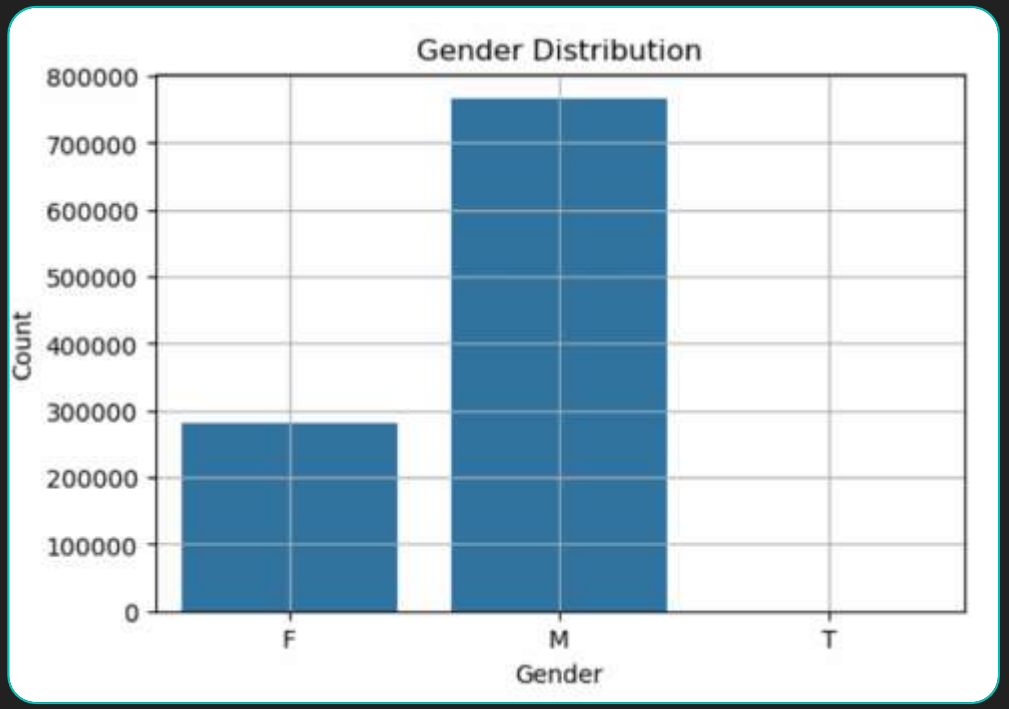
EDA - Age Distribution

- Most customers are aged 30–40.
- Ideal for targeting financial products.
- Older customers are less frequent.



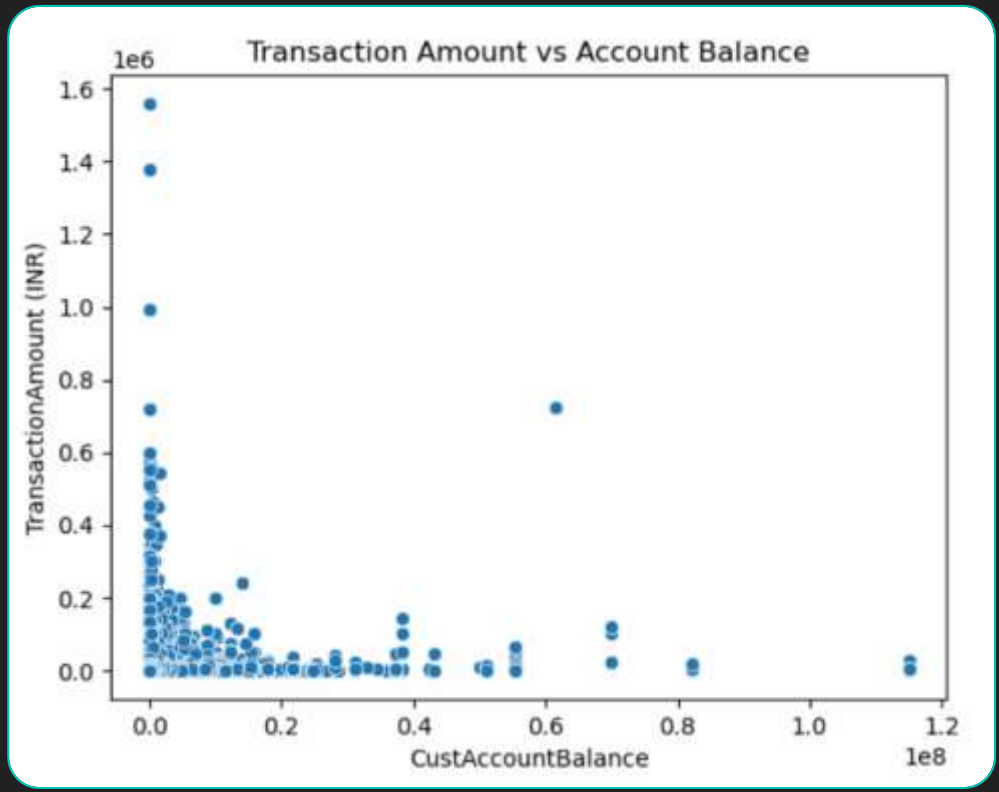
EDA - Gender Distribution

- Majority male (~750,000), females ~280,000.
- 'Other' values are minimal.
- Imbalance may affect modeling.



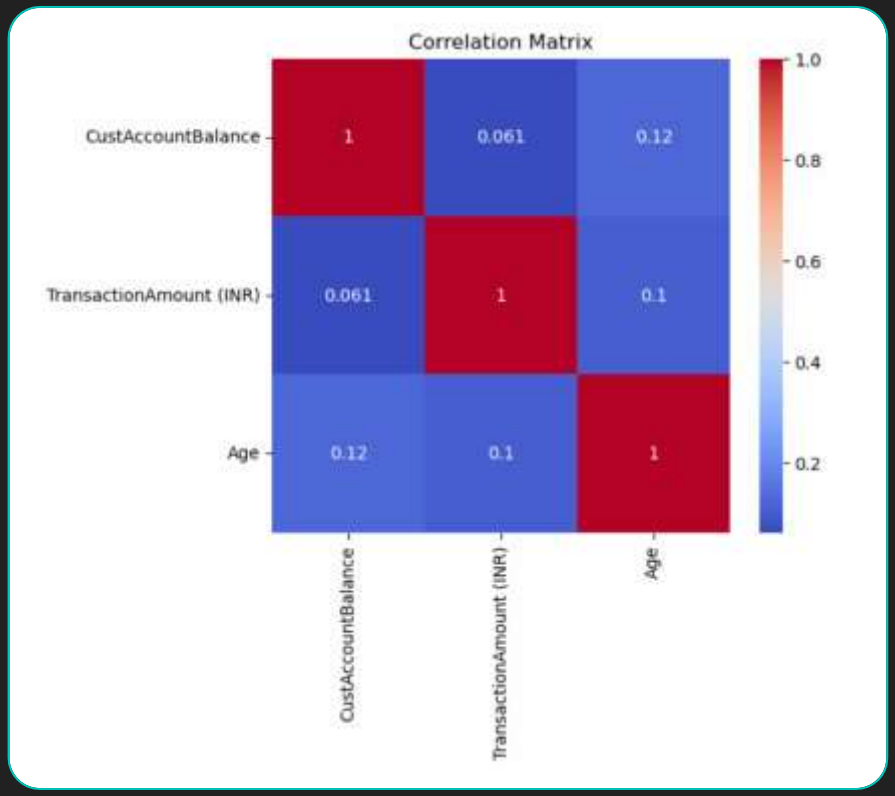
EDA - Transaction vs. Balance

- No strong linear relationship.
- High-value transactions seen from low-balance accounts.
- Potential anomalies or business behavior.



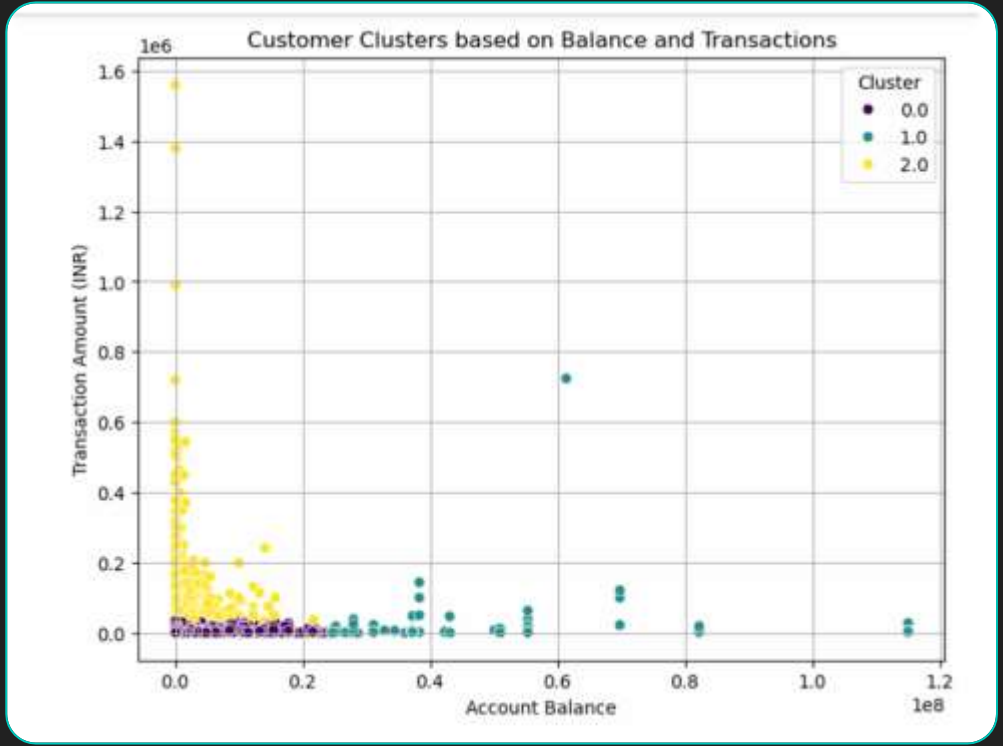
Correlation Analysis

- Weak correlations:
- • Balance & Transaction Amt: 0.061
- • Age & Balance: 0.12
- • Age & Transaction: 0.10



Clustering Analysis

- Three clusters found:
- • Cluster 0: Moderate balance, low transactions
- • Cluster 1: High balance, low activity
- • Cluster 2: Low balance, high transactions



Cluster Interpretation

- Cluster 0: Young, basic banking users
- Cluster 1: Seniors with wealth
- Cluster 2: Middle-aged, active users

Key Insights

- Age is a strong segmentation factor
- High-value, low-balance transactions indicate risk
- Most active customers are in Mumbai/Navi Mumbai

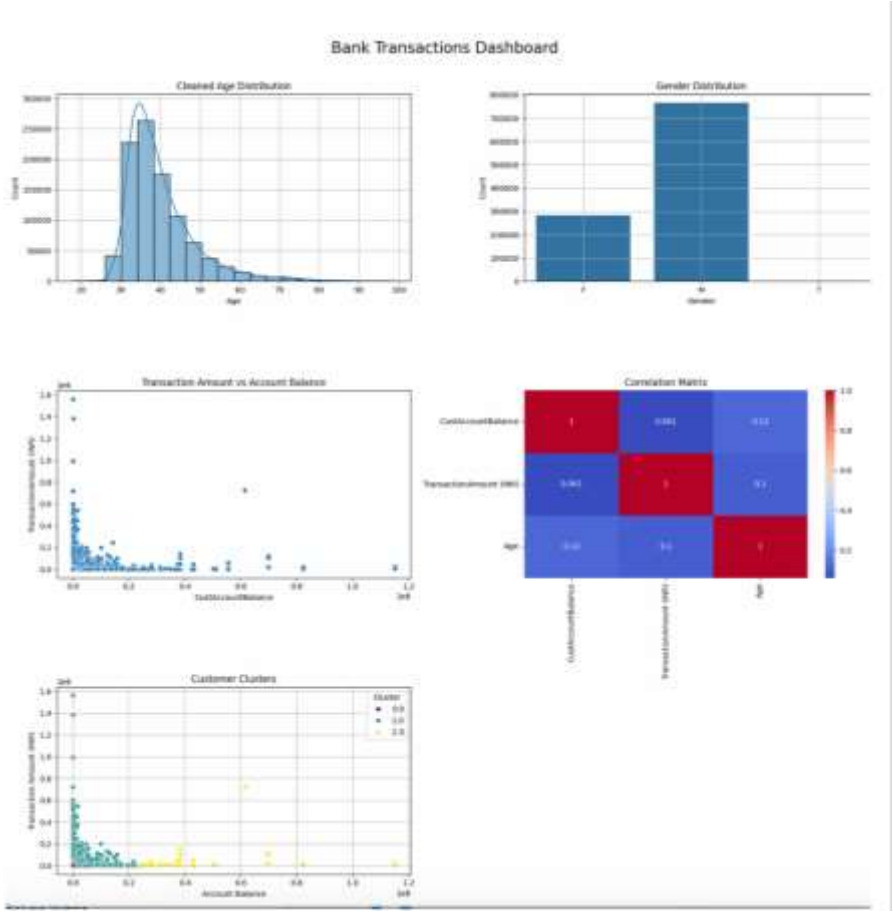
Dashboard Design

- The dashboard includes key visualizations to aid stakeholders:

- • Age Distribution Histogram – Identifies dominant age groups for targeting
- • Gender Breakdown Bar Chart – Highlights the male-female imbalance
- • Transaction vs. Balance Scatter Plot – Reveals anomalies or overdraft behavior
- • Correlation Heatmap – Shows relationships among variables
- • Customer Segments – Visual representation of clusters

- These elements provide an interactive overview of customer behavior and guide segmentation strategies.

Dashboard


Bank Transactions Dashboard
Cleaned Age Distribution
Gender Distribution
Transaction Amount vs Account Balance
Correlation Matrix
Customer Clusters

Graph Analysis of Banking Customer Networks

- City Connections and Network Structure

- This section explores the geographic and transactional structure of the bank's customer network using graph analytics.

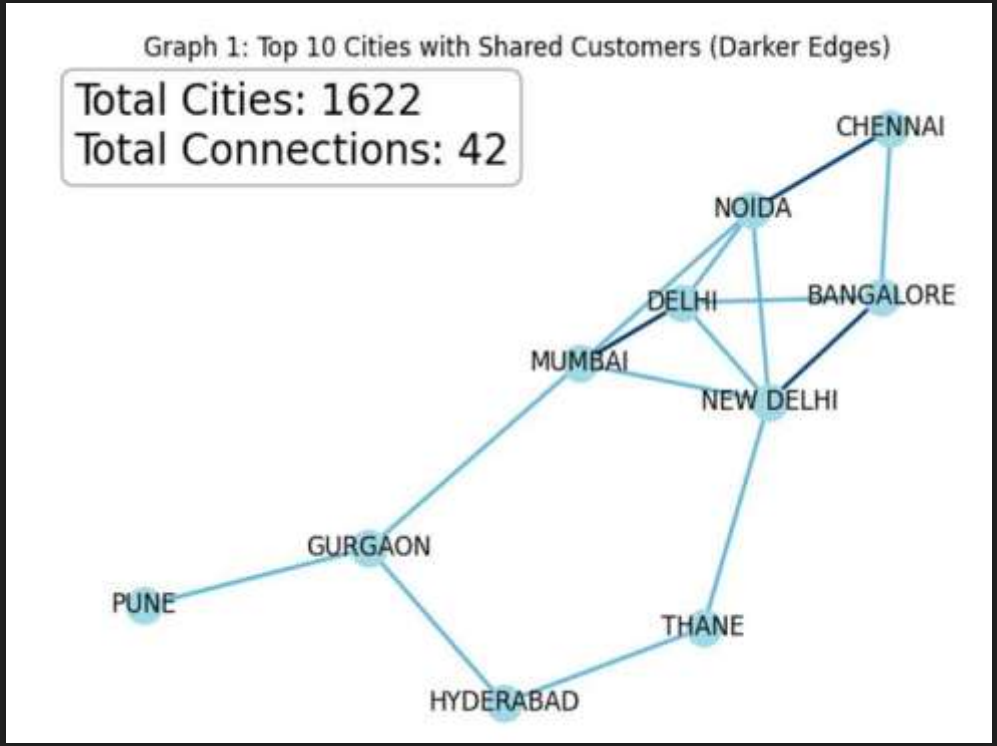
Geographic Network Analysis

Top 10 Cities with Shared Customers:

- 1,622 cities in the network

- Only 42 meaningful inter-city connections

- Mumbai, Delhi, and Bangalore form a strong financial triangle



Full Network Structure Analysis

Metric	Value	Significance
Nodes (Customers)	16,687	Core active customer base
Edges (Transactions)	5,171,660	Extremely dense network
Avg. Degree	620	Each customer connects to 620 others
Network Density	0.037	Higher than typical banking networks

- Key Observations:
- - 'Small world' network: average path length = 2.1
- - Power-law distribution: few nodes dominate transaction volume
- - Dense network: 5M+ edges among 16K nodes

- Insights:
- - Many high-volume business accounts
- - Potential fraud patterns (1:1 in/out flows)
- - High digital payment adoption in India

Network statistics output

```
>>> print("Number of nodes:", G.number_of_nodes())
Number of nodes: 16687
>>> print("Number of edges:", G.number_of_edges())
Number of edges: 5171660
```

Combined Insights & Recommendations

- Strategic Opportunities:
- - Metro-Centric Services: Focus on Mumbai, Delhi, Bangalore
- - Network-Based Risk Management: Monitor high-connection accounts
- - Geographic Expansion: Target Pune, Hyderabad, and underserved cities
- Technical Recommendations:
- - Real-time graph database (e.g., Neo4j)
- - Graph-based fraud detection features
- - Allocate infrastructure based on city connectivity

Key Achievements:
- Precision segmentation with clustering
- Metro-level geographic insights
- Graph-based fraud detection
- Small-world transaction structure

Recommendations:
- Product development for city clusters
- Real-time network monitoring for fraud
- Invest in metro infrastructure
- Use community detection for marketing

Conclusion