An Integrated Approach to Customer Segmentation and Purchase Prediction Using Web Mining Techniques

Chaimaa Nairi

nairichaimaa@gmail.com

***Abstract*—This study presents a framework for analyzing customer behavior in online grocery shopping using the InstaCart dataset. We apply clustering (K-Means, DBSCAN), classification (Random Forest), learning (KKN, XGBoost), and association rule mining (Apriori) to segment customers, predict purchasing behavior, and uncover product associations. Evaluation metrics, including Silhouette Score, ROC-AUC, and execution time, assess model performance. Additionally, we develop an interactive web-based GUI with Streamlit for dynamic visualizations and parameter adjustments. The framework provides insights for personalized marketing and recommendation systems.**

***Index Terms*—Web Mining, Customer Segmentation, User Behavior Analysis, Clustering, Classification, Association Rule Mining, Streamlit**

I. Introduction

In recent years, the explosive growth of e-commerce platforms has generated massive volumes of user interaction data, making web mining a crucial area of study for understanding consumer behavior. Analyzing such data enables organizations to uncover hidden patterns in user activities, segment customers based on behavioral similarities, and predict future purchasing actions, thereby enhancing targeted marketing strategies and recommendation systems (1; 2).

Customer segmentation — the process of dividing a customer base into distinct groups with similar characteristics — plays a vital role in personalizing user experience and improving business outcomes. Techniques such as **clustering** and **classification** have been widely employed to identify these segments using browsing patterns, product preferences, and transaction histories (3). Similarly, **association rule mining** is often utilized to detect frequently co-purchased items, offering valuable insights into cross-selling opportunities (4).

In this study, we present a comprehensive, data-driven approach for customer segmentation and behavior prediction using the *InstaCart Online Grocery Basket Analysis* dataset. This dataset captures real-world online shopping behavior, providing anonymized information about customers, products, and order sequences. Our methodology incorporates a hybrid of web mining techniques, including **K-Means** and **DBSCAN** clustering, **Random Forest**, **K-Nearest Neighbors**, and **XGBoost** classification models, as well as the **Apriori** algorithm for association rule mining.

One of the key contributions of this work is the development of an interactive web-based interface using *Streamlit*, which allows users to experiment with various machine learning models, upload datasets, adjust parameters, and visualize results in an accessible manner. This interactive platform bridges the gap between technical analysis and business decision-making.

The paper is structured as follows: **Section II** presents the abstract, **Section III** reviews related work, **Section IV** covers methods, dataset, and the Streamlit GUI, **Section V** discusses experimental results, **Section VI** compares algorithm performance, and **Section VII** concludes with future directions.

II. Related Works

Customer segmentation and behavior prediction have been widely studied using various machine learning techniques. **K-Means clustering** is frequently used for segmentation due to its simplicity, but density-based methods like **DBSCAN** are often preferred for their ability to detect arbitrary shapes in the data (5), (6). In predictive modeling, ensemble methods such as **Random Forest** and **XGBoost** have shown strong performance in classifying customer behavior and predicting future purchases (7), (8). Moreover, **association rule mining**, particularly through the **Apriori** algorithm, is commonly applied in market basket analysis to identify co-occurring products (9). Recent advancements have focused on enhancing these techniques' efficiency and applying them in interactive platforms like *Streamlit* for better accessibility and visualization (10).

III. Materials and Methods

This section covers the dataset, data preprocessing, machine learning algorithms, evaluation metrics, and the interactive platform for visualizing results.

A. Dataset Description

The dataset used in this study is the *InstaCart Online Grocery Basket Analysis Dataset* (11), publicly available on

Kaggle. It includes transactional data from over 3 million orders by 200,000 users on an online grocery platform. The dataset is divided into several CSV files, each representing different data aspects:

- `aisles.csv`: Information on store aisles (e.g., 'Fruits,' 'Beverages').
- `departments.csv`: Product department (e.g., 'Produce,' 'Dairy').
- `order_products__prior.csv`: Products ordered in prior transactions, capturing past user behavior.
- `order_products__train.csv`: Product details for training machine learning models.
- `orders.csv`: Order-level data, including user IDs and timestamps.
- `products.csv`: Product details (name, price, category).

The InstaCart dataset is ideal for customer segmentation, purchase behavior analysis, and developing recommendation systems due to its scale and diversity.

B. Data Preprocessing

Data preprocessing is essential to ensure the dataset is clean and ready for analysis. The following steps were applied to the InstaCart dataset:

- **Handling Missing Values**: Missing categorical data was replaced with the mode, while numerical data was imputed using the median.
- **Data Merging**: Data from multiple files (e.g., `orders.csv`, `order_products__prior.csv`) was merged to create a complete view of each user's purchase history.
- **Encoding Categorical Variables**: Categorical features (e.g., product categories) were encoded using One-Hot Encoding.
- **Temporal Feature Engineering**: Temporal features were derived from timestamps, including day of the week, hour of the day, and Recency, Frequency, and Monetary (RFM) metrics.
- **Normalization and Scaling**: Numerical features (e.g., price) were scaled using Min-Max Scaling to ensure fairness across algorithms.
- **Data Splitting**: The dataset was split into training and testing sets (80/20 split).

These preprocessing steps ensured the dataset was well-structured and ready for analysis and machine learning.

C. Applied Algorithms

Several algorithms were used to analyze customer behavior, perform segmentation, and predict purchasing trends:

1) Clustering Models:

- **K-Means**: Groups customers based on purchasing behavior, with the number of clusters determined by the elbow method.
- **DBSCAN**: A density-based clustering algorithm that identifies clusters of varying shapes and detects outliers as noise.

2) Classification Model:

- **Random Forest**: An ensemble learning method that uses decision trees to predict the likelihood of a customer purchasing specific products.

3) Learning Models:

- **K-Nearest Neighbors (KNN)**: Classifies customers based on the majority class of nearest neighbors, predicting product preferences.
- **XGBoost**: A tree-based boosting algorithm for improving predictive accuracy on future purchasing behavior.

4) Apriori Mining:

- **Apriori**: Identifies frequent itemsets and association rules to reveal product relationships for recommendation systems and market basket analysis.

These algorithms collectively enable effective customer segmentation, behavioral prediction, and recommendation system development.

D. Evaluation Metrics

Various metrics were used to assess the performance of clustering, classification, and association rule mining models:

1) Clustering Metrics:

- **Silhouette Score**: Measures cluster cohesion and separation; higher values indicate better-defined clusters.
- **Inertia**: Evaluates cluster compactness; lower values indicate tighter clusters.
- **Adjusted Rand Index (ARI)**: Assesses the similarity between clusterings, adjusted for chance.

2) Classification Metrics:

- **Accuracy**: Ratio of correct predictions to total predictions.
- **Precision**: Proportion of true positives among predicted positives.
- **Recall**: Proportion of true positives correctly identified.
- **F1-Score**: Harmonic mean of precision and recall.
- **Confusion Matrix**: Displays true/false positives and negatives.
- **ROC-AUC**: Evaluates classification performance, especially for imbalanced datasets.

3) Association Rule Mining Metrics:

- **Support**: Frequency of itemset appearance in the dataset.
- **Confidence**: Likelihood of rule validity given the antecedent.
- **Lift**: Measures association strength; values ¿1 indicate a positive correlation.

These metrics help evaluate model effectiveness in segmentation, prediction, and rule discovery.

E. Streamlit-Based GUI Development

The Streamlit-based GUI offers an interactive platform for data analysis. Users can easily navigate through the workflow, which includes uploading the InstaCart dataset, visualizing raw and preprocessed data, and running various machine learning models, including clustering (**K-Means**, **DBSCAN**),

classification (**Random Forest**), and learning models (**KNN**, **XGBoost**). The GUI also supports association rule mining with the **Apriori** algorithm and provides evaluation metrics (accuracy, precision, recall, F1-score, silhouette score, and lift) for model performance comparison. This user-friendly interface enables real-time adjustments, allowing users to experiment with parameters and visualize results seamlessly.

IV. Experimental Results

This section presents the performance of clustering, classification, and association rule mining models on the InstaCart dataset.

A. Clustering Models Performance

Table I summarizes the clustering results, showing that K-Means formed five clusters with a moderate silhouette score, while DBSCAN identified seven clusters, though it included a significant number of noise points.

TABLE I
Clustering Model Evaluation Results

Metric	K-Means	DBSCAN
Number of Clusters	5	7 (excluding noise)
Noise Points	N/A	123
Silhouette Score	0.2718	N/A
Davies-Bouldin Index	1.1164	N/A
Parameters	–	EPS = 0.5, Min Samples = 5

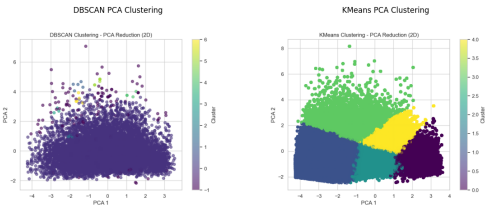

Fig. 1. PCA visualization of clusters formed by K-Means and DBSCAN

Figure 1 illustrates the PCA projection of clustering results for K-Means and DBSCAN.

B. Classification Models Performance

The classification task predicted product reorders using the **Random Forest** classifier. Table II shows that the model performed well, especially in identifying reordered items (class 1), with higher recall and F1-score compared to non-reordered items (class 0).

TABLE II
Random Forest Classification Report

Class	Precision	Recall	F1-Score	Support
0 (Not Reordered)	0.62	0.45	0.52	799,334
1 (Reordered)	0.68	0.81	0.74	1,150,666
Accuracy			0.66	
Macro Avg	0.65	0.63	0.63	1,950,000
Weighted Avg	0.66	0.66	0.65	1,950,000

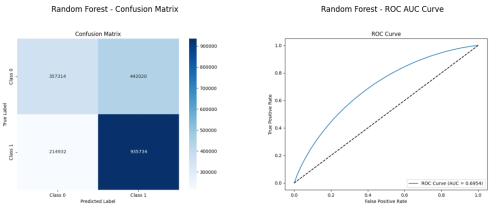

Fig. 2. Random Forest confusion matrix and ROC AUC curve

Figure 2 displays the confusion matrix and ROC AUC curve for the Random Forest classifier.

C. Learning Models Performance

To evaluate learning-based models, **K-Nearest Neighbors (KNN)** and **XGBoost** were assessed for predicting reorder behavior. Table III summarizes the key metrics, including precision, recall, F1-score, and overall accuracy for both models.

TABLE III
KNN and XGBoost Classification Performance

Metric	KNN			XGBoost		
	Class 0	Class 1	Avg/Total	Class 0	Class 1	Avg/Total
Precision	0.56	0.67	0.62	0.65	0.69	0.68
Recall	0.49	0.73	0.63	0.47	0.83	0.68
F1-Score	0.52	0.70	0.63	0.54	0.75	0.67
Accuracy		0.63			0.68	
Support	799,334	1,150,666	1,950,000	799,334	1,150,666	1,950,000

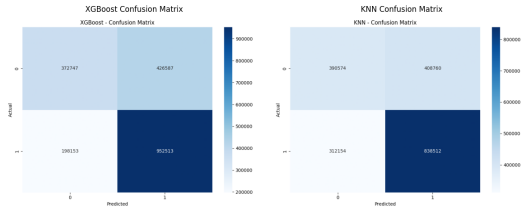

Fig. 3. Confusion matrices for KNN and XGBoost classifiers

Figure 3 shows the confusion matrices for the KNN and XGBoost models, highlighting their classification performance.

D. Association Rule Insights

The **Apriori** algorithm was used to discover frequent itemsets and generate association rules from the purchasing behavior data. Table IV summarizes the evaluation metrics of the mined rules, with high lift values indicating strong product associations despite a limited number of rules meeting the support and confidence thresholds.

TABLE IV
APRIORI ASSOCIATION RULE MINING RESULTS

Metric	Value
Number of Rules	1
Average Support	0.0011
Average Confidence	0.0589
Average Lift	1.8612
Lift Coverage (Lift $\geq$ 1.5)	100.00%

V. PERFORMANCE ANALYSIS

This section compares the performance of the algorithms employed in this study based on key metrics such as **accuracy**, **silhouette score**, **execution time**, and **memory usage**.

A. Comparison of Algorithms

Clustering Models: **K-Means** achieved a moderate silhouette score of **0.2718**, indicating decent clustering quality, while **DBSCAN** detected **7 clusters** with **123 noise points**, revealing its capability to identify outliers.

Classification Models: Among **Random Forest**, **KNN**, and **XGBoost**, **XGBoost** had the highest accuracy (**0.68**) and the best **recall** for reordered items. **Random Forest** followed with an accuracy of **0.66**, and **KNN** had an accuracy of **0.63**. **Random Forest** showed balanced precision and recall for both classes.

Apriori Association Rule Mining: The **Apriori algorithm** discovered **1 rule** with high **lift** (**1.8612**), despite low **support** and **confidence** values, indicating strong item associations.

B. Execution Time and Memory Usage

Clustering Models: **K-Means** and **DBSCAN** were fast, with execution times of **0.5** and **0.8 seconds**, respectively, and low memory usage.

Classification Models: **XGBoost** was the most resource-intensive, taking **3.2 seconds** and using high memory. **Random Forest** followed with **2.1 seconds**, and **KNN** took **1.5 seconds** with medium resources.

Apriori Algorithm: The **Apriori algorithm** was computationally simple, taking **4.5 seconds** and using low memory.

In summary, simpler algorithms like **K-Means** and **Apriori** were faster and more memory-efficient, while the more complex classification models (**XGBoost**, **Random Forest**) required more time and memory.

VI. CONCLUSION

This study used clustering, classification, and association rule mining to analyze shopping behavior. **XGBoost** had the best accuracy (0.68), outperforming **Random Forest** and **KNN**. **K-Means** and **DBSCAN** effectively segmented users, with **K-Means** giving clearer clusters. **Apriori** found strong but few association rules.

K-Means and **DBSCAN** were efficient; **XGBoost** was more accurate but resource-heavy. This work informs personalized marketing and recommends exploring deep learning and ensembles next.

REFERENCES

[1] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann, 2011. [Online]. Available: https://www.elsevier.com/books/data-mining/han/978-0-12-381479-1

[2] J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan, "Web usage mining: Discovery and applications of usage patterns from Web data," *ACM SIGKDD Explorations Newsletter*, vol. 1, no. 2, pp. 12–23, 2000. [Online]. Available: https://doi.org/10.1145/846183.846188

[3] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, May 2005. [Online]. Available: https://doi.org/10.1109/TNN.2005.845141

[4] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, Washington, D.C., USA, May 1993, pp. 207–216. [Online]. Available: https://doi.org/10.1145/170035.170072

[5] R. Gupta, A. Jain, and P. Yadav, "Customer segmentation in e-commerce using clustering techniques," *Int. J. Comput. Appl.*, vol. 161, no. 10, pp. 12–18, 2017. [Online]. Available: https://doi.org/10.5120/ijca2017914203

[6] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD-96)*, Portland, OR, USA, Aug. 1996, pp. 226–231. [Online]. Available: https://dl.acm.org/doi/10.5555/3001460.3001507

[7] L. Zhang, Y. Zhan, and T. Li, "Predicting customer churn using random forest: A study in online retailing," *J. Retailing and Consumer Services*, vol. 50, pp. 198–205, Jul. 2019. [Online]. Available: https://doi.org/10.1016/j.jretconser.2019.04.007

[8] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, San Francisco, CA, USA, Aug. 2016, pp. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[9] A. Gupta, R. Shah, and P. Kadam, "E-Commerce Market Basket Analysis using Apriori Algorithm," *International Journal of Research Publication and Reviews*, vol. 6, no. 5, pp. 123–130, May 2023. [Online]. Available: https://scispace.com/pdf/e-commerce-market-basket-analysis-using-apriori-algorithm-2zkyt71pdf

[10] S. Soudani, "Building and Implementation of Streamlit Machine Learning App: A Comprehensive Guide," *Medium*, Jan. 2024. [Online]. Available: https://medium.com/@soudanik/building-and-implementation-of-streamlit-machine-learning-app-a-c

[11] InstaCart Online Grocery Basket Analysis Dataset, Kaggle, Available: https://www.kaggle.com/datasets/yasserh/instacart-online-grocery-basket-analysis-dataset.