

NORTHEASTERN UNIVERSITY

INTRODUCTION TO DATA MANAGEMENT AND PROCESSING

PROJECT REPORT

DS 5110

PREDICTING NEXT PURCHASE OF AN INSTACART CONSUMER

Authors

Rohit Thakur
Chetana Sharma
Sapna Sharma

Supervisor

Dr. Kylie Ariel Bemis

April 19, 2020

1. Summary

1.1 Background

Online supermarkets and mobile apps are the most preferred form of grocery shopping these days. That being said, it is important for retailers to effectively increase sales by analyzing trends in customer purchase histories, identifying relationships between two or more products to create bundles of items that might sell together. This kind of data analysis helps retailers in efficiently managing inventory, decide pricing on items, and create focused offers targeting specific customer segments.

1.2 Purpose

This project mainly focuses on implementing various methods to predict which products a user is most likely to buy in the next transaction session based on purchase history. We seek to demystify trends in consumer purchases through exploratory analysis. Further, we aim to generate bundles of items which are bought frequently by users also known as frequent itemsets. Also, providing recommendations on product and department level based on an existing order.

1.3 Description of Data

The data shown in fig. A.1 consists of metadata for over 3 million Instacart orders from more than 200,000 users distributed over five tables which are department, aisle, product, prior_orders, transactions. Each table can be identified by a unique id assigned to it. These tables have some useful information regarding orders such as day of the week, hour of the day. Also it provides useful insights regarding whether a product was reordered and sequence by which a product was placed into a shopping cart.

1.4 Overview

Using exploratory data analysis we aim to observe consumer buying patterns on department, aisles and products level; effect of day of week and hour of day on number of orders; best selling products throughout orders ; reordering proportions and patterns to know products getting reordered and interval of reorders; number of times each product getting reordered; average number of products getting added to shopping cart. In addition, we seek to identify item sets which are usually bought together and generate association rules based on different parameters such as support, threshold and lift. We build a recommender system to suggest what items a consumer will purchase next based on current ordering information by exploiting various collaborative filtering techniques.

2. Methods

2.1 Data Wrangling

The data given was mostly tidy. The main part was to join the different tables using the *dplyr* package to get the required columns in one order table. For example - the `order_products_prior` table was joined with the `products` table to get the `department_id` so as to be able to do the analysis of the orders based on the department and aisle as well. Certain columns like `order_dow`, which is the day of the week the order was placed, from the order table were converted from numeric to factor. One other such column was `order_hour_of_day`. The exploratory data analysis was then performed to get insights of the trends in purchase of products.

2.2 Modeling

We used two approaches for the recommendation. The first was to generate frequent item-set and the other was to recommend the next few products in the cart given some existing products in the cart.

2.2.1 Frequent itemset

Apriori Algorithm is the most used method to find out associations between items. It assumes that any subset of a frequent itemset must be frequent. It helps the retailers find the relationship between the products that users buy together. Association Rules discovered in the transactions on the basis of confidence, and lift are of great interest to the retailers. Association Rule-based algorithms are viewed as a two-step approach:

1. **Frequent Itemset Generation:** Find all frequent item-sets with support \geq predetermined `min_support` count
2. **Rule Generation:** List all Association Rules from frequent item-sets. Calculate Support and Confidence for all rules. Prune rules that fail `min_support` and `min_confidence` thresholds.

$$\begin{aligned}
 \text{Rule: } X \Rightarrow Y & \begin{cases} \nearrow \text{Support} = \frac{\text{freq}(X, Y)}{N} \\ \rightarrow \text{Confidence} = \frac{\text{freq}(X, Y)}{\text{freq}(X)} \\ \searrow \text{Lift} = \frac{\text{Support}}{\text{Supp}(X) \times \text{Supp}(Y)} \end{cases}
 \end{aligned}$$

Figure 2.1: Parameters in association rules

Support gives an idea of how frequent an itemset is in all the transactions.

Confidence is the likeliness of occurrence of consequent on the cart given that the cart already has the antecedents.

Lift is the rise in probability of having $\{Y\}$ on the cart with the knowledge of $\{X\}$ being present over the probability of having $\{Y\}$ on the cart without any knowledge about presence of $\{X\}$.

2.2.2 Recommendation of next products

We used the *recommenderlab* package for this objective. The first step was to select a subset of the data from the huge dataset available. To avoid the problem of sparsity, the orders of the top 200 most frequent users were used for training and evaluation of algorithms. The *recommenderlab* training method required training data in the form of binary matrices. Therefore, a user vs product matrix and a user vs department matrix was created. The cells corresponding to products purchased by the users were marked as 1 else 0.

Next, these matrices generated were used to train and evaluate a few algorithms using 5-fold cross-validation with 80% of the data used for training and the remaining for evaluation. The precision-recall curve was used to choose the best algorithm for doing the final recommendation. The user-based collaborative filtering performed best for both the user-department binary matrix as well the user-product rating matrix as seen in fig. 2.2. The numbers in the box indicate the ‘number of items recommended’ while evaluating the algorithm. The algorithm was able to recommend the next 10 relevant products with a better accuracy than other numbers.

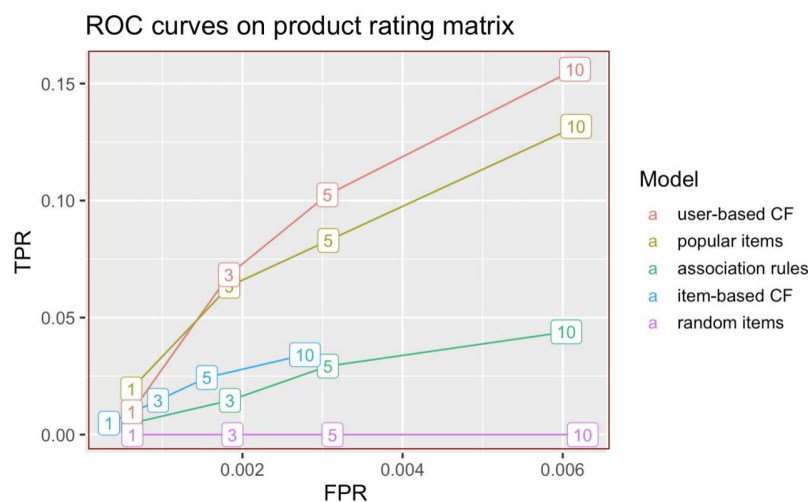


Figure 2.2: ROC curves for different algorithms on user-product matrix

3. Results

3.1 Exploratory Data Analysis

3.1.1 Customer buying pattern: Popularity of Department and aisle

From the treemap we see that, produce department is the most popular department as per the sales of the products and fresh fruits and fresh vegetables are the aisles from where most of the sales occur, followed by the packaged vegetables and fruits aisle.

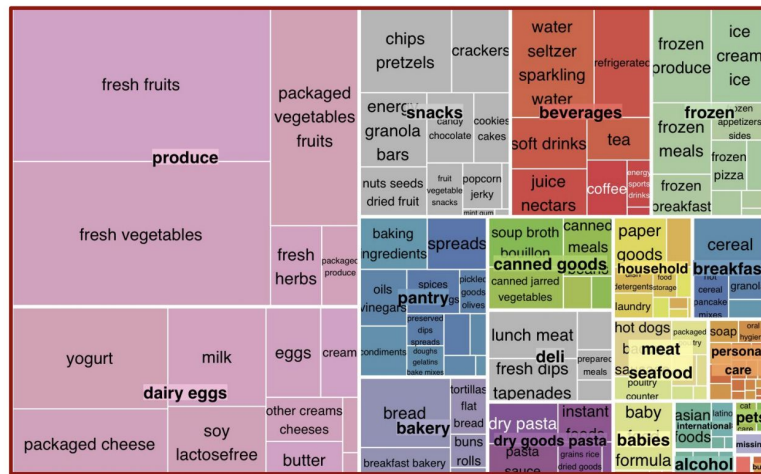


Figure 3.1: Treemap showing departments and aisles by sales

3.1.2 Customer buying pattern : Most popular products

It is interesting to note that the most popular items ordered from Instacart are bananas and bags of organic bananas, followed by organic strawberries and organic baby spinach. Top twenty most popular items can be seen in the bar plot below:

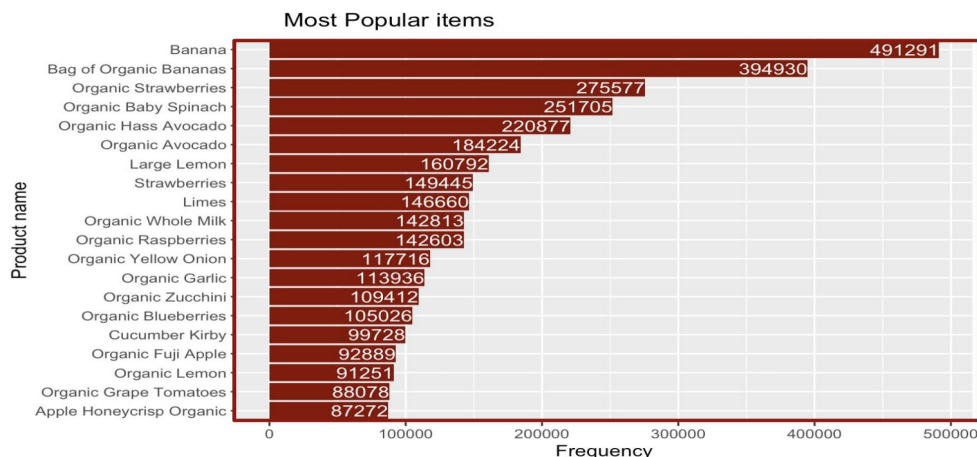


Figure 3.2 Bar chart of the twenty most popular products

3.1.3 Customer buying pattern : By the day of the week and hour of the day

We see that most customers book an order between 9 a.m. to 3 p.m. We see that a higher number of orders are placed on day0 and day1, which indicate that these maybe weekends.

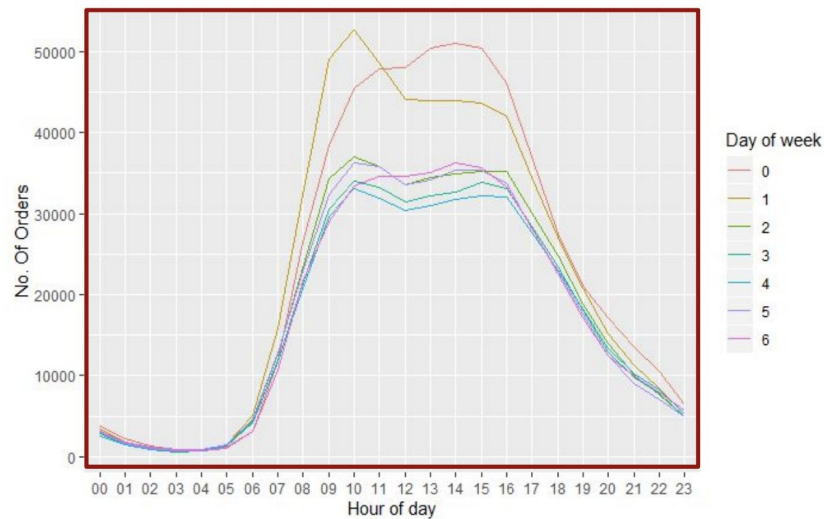


Figure 3.3 Plot showing the number of orders by the day and hour

3.1.4 Customer reordering pattern :

We see that most reordered products are again bananas and a bag of organic bananas. From the plot of reorders by the day of the month we see that most customers reorder on the seventh day or at the end of the month. Local peaks at day 14 and day 21 also indicate weekly buying patterns.

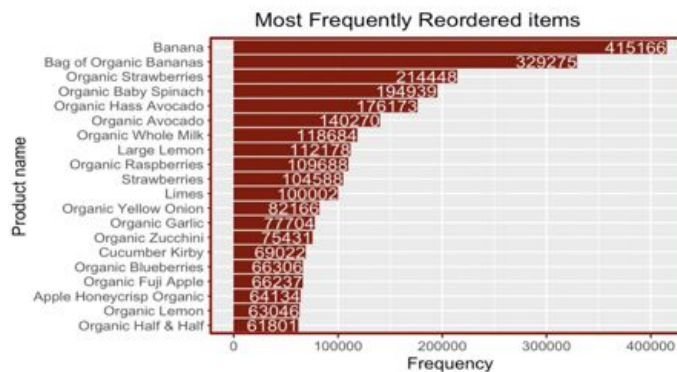


Figure 3.4 Bar chart of most frequently re-ordered products

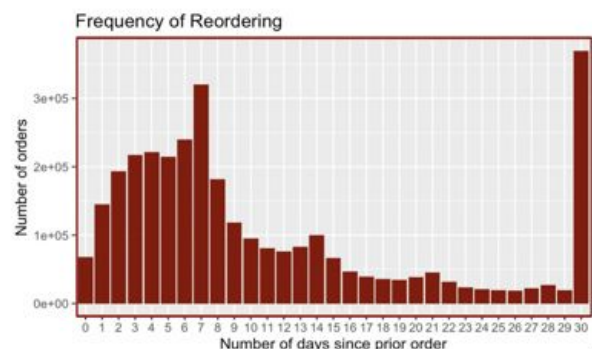


Figure 3.5 Frequency of reordering

3.2 Model output

3.2.1 Frequent itemset generation

Twenty-seven rules were generated for a support of 0.01 and confidence of 0.1. From the table of rules, we can say that if a customer buys organic strawberries, it is likely that the customer will buy organic raspberries in their next purchase.

	lhs <fctr>		rhs <fctr>	support <dbl>	confidence <dbl>	lift <dbl>
[1]	{Organic Strawberries}	=>	{Organic Raspberries}	0.01061928	0.1289404	3.025499
[2]	{Organic Raspberries}	=>	{Organic Strawberries}	0.01061928	0.2491743	3.025499
[3]	{Organic Fuji Apple}	=>	{Banana}	0.01050572	0.3784409	2.577484
[4]	{Organic Raspberries}	=>	{Bag of Organic Bananas}	0.01263657	0.2965085	2.512197
[5]	{Bag of Organic Bananas}	=>	{Organic Raspberries}	0.01263657	0.1070645	2.512197

Figure 3.6 Rule generation output

3.2.2 Recommendation of next products

Using user-based collaborative filtering, we predicted the next few products of a test order.

```
prod_test_order <- c("Ancient Grain Blueberry Hemp Granola",
  "Tuna Salad",
  "Organic Raspberries",
  "Ground Turkey Breast",
  "Organic Sour Cream")
```

Recommender of type 'UBCF' for 'binaryRatingMatrix'
learned using 160 users.

\$`1`

```
[1] "Banana"           "Organic Strawberries" "Strawberries"
[4] "Organic Fuji Apple" "Honey Nut Cheerios"  "Bag of Organic Bananas"
[7] "Extra Virgin Olive Oil" "Small Hass Avocado"  "Organic Blackberries"
[10] "Limes"
```

Figure 3.7 User-based collaborative filtering results

Three of the ten recommended products mentioned in fig 3.7 were present in the set of products most frequently ordered along with the test products.

4. Discussion

4.1 Outcome

From the results, it can be observed that most of the consumers follow a healthy eating habit by ordering fruits and vegetables more frequently. The data as well as the intuition aligns on the fact that most of the orders are placed in the middle of the day on weekends. Also, we discovered that the customers place the next order at a weekly or monthly interval. The User-based Collaborative filtering recommended relevant items even after training on a small subset.

4.2 Future Scope

- The frequent itemset could be generated for more number of items in a set.
- The sampling method to get training data from the given huge dataset could be improved to include an unbiased set of users irrespective of the frequency of ordering.
- The high execution time and less accurate results due to sparsity of the data could be improved by using a larger training data set and using matrix factorisation like Singular Value Decomposition.
- Future work, could include a text analysis on the name of the products and to determine the healthy vs unhealthy purchase of items based on keywords present in the name of the products like organic, burger, etc.

5. Statement of Contributions

The entire team participated in the ideation of the project, EDA and research around the different types of recommender systems available. Chetana Sharma and Rohit Thakur were involved in the data cleaning, wrangling and modeling using collaborative filtering techniques and evaluation of the results. Sapna Sharma was involved in the data tidying, EDA and generation of frequent itemset using association rules.

Task	Team Members		
	Chetana Sharma	Rohit Thakur	Sapna Sharma
Data Understanding			
Data Cleaning			
Data Pre-processing and Wrangling			
EDA and Visualizations			
Frequent Itemset generation and Association Rules			
Collaborative Filtering Techniques			
Evaluation			


Legend
Participation


Figure 5.1 Contribution chart

References

- [1] Kaggle data: <https://www.kaggle.com/c/instacart-market-basket-analysis>
- [2] Jekaterina Novikova, PhD: Building a Movie Recommendation System.
URL: <https://rpubs.com/jeknov/movieRec>
- [3] Diego Usai. Recommenderlab:
URL: <https://towardsdatascience.com/market-basket-analysis-with-recommenderlab-5e8bdc0de236>
- [4] Michael Hahsler. recommenderlab: A Framework for Developing and Testing Recommendation Algorithms
URL: <https://cran.r-project.org/web/packages/recommenderlab/vignettes/recommenderlab.pdf>
- [5] Association rule learning:
URL: https://en.wikipedia.org/wiki/Association_rule_learning
- [6] Nagesh singh chauhan. Market basket analysis:
URL: <https://towardsdatascience.com/market-basket-analysis-978ac064d8c6>
- [7] Susan Li. A Gentle Introduction on Market Basket Analysis — Association Rules
URL: http://www.saedsayad.com/association_rules.htm
- [8] R for Data Science. Wickham and Grolemund. 1st ed. 2017
- [9] The R Graph Gallery:
URL: <https://www.r-graph-gallery.com/>

Appendix

A.1 Data Description

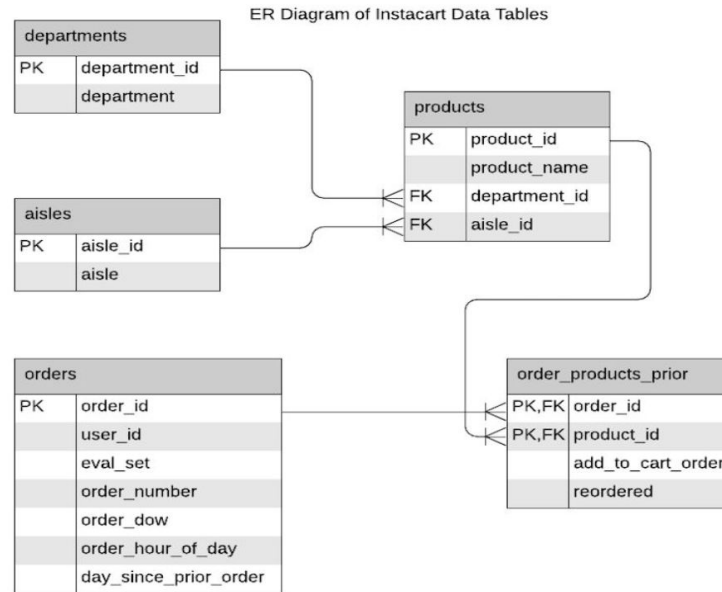


Figure A.1: ER Diagram of the data tables

A.2 EDA figures

A.2.1 Most popular Aisles

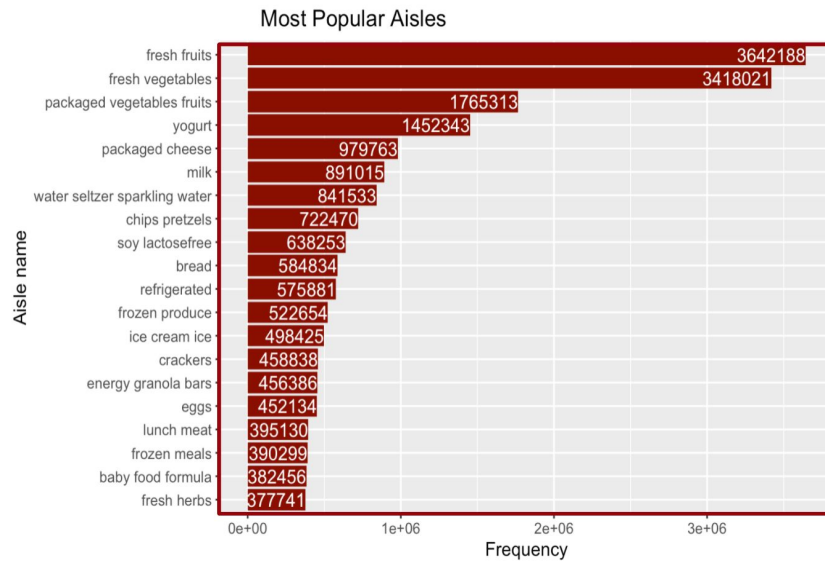


Figure A.2 Most popular aisles

A.2.2 Top products added to cart first

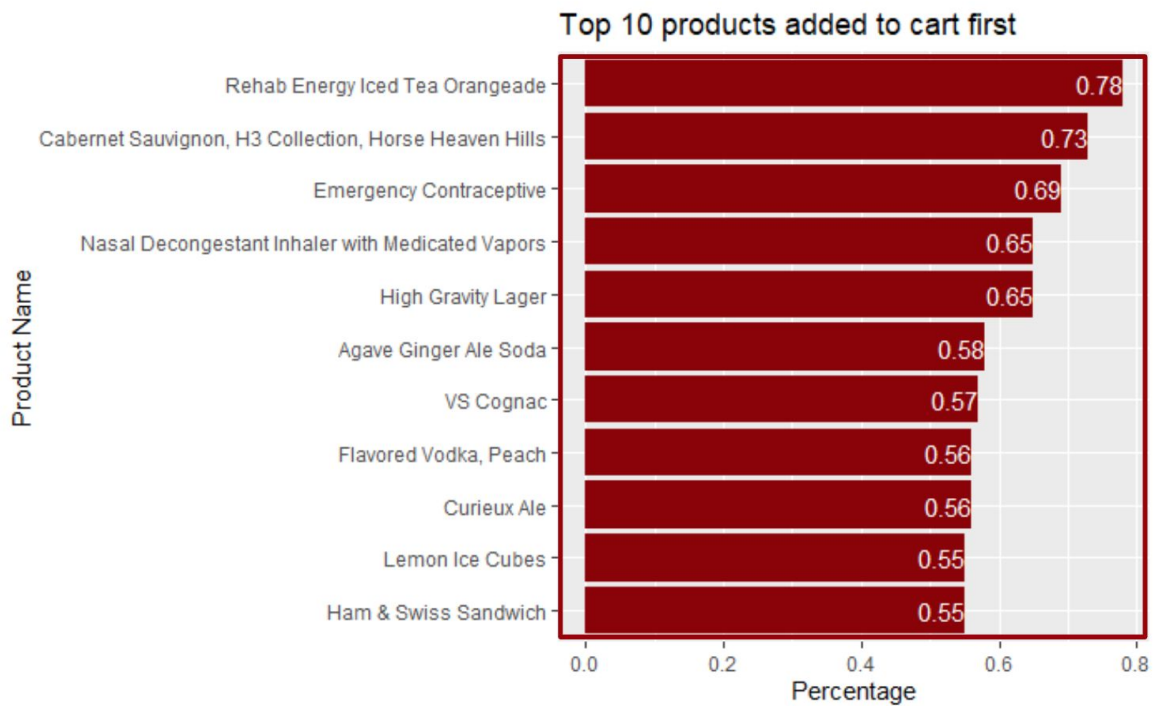


Figure A.3 Top 10 products added to cart first along with proportion

A.2.3 Association between number of orders and proportion of reorder

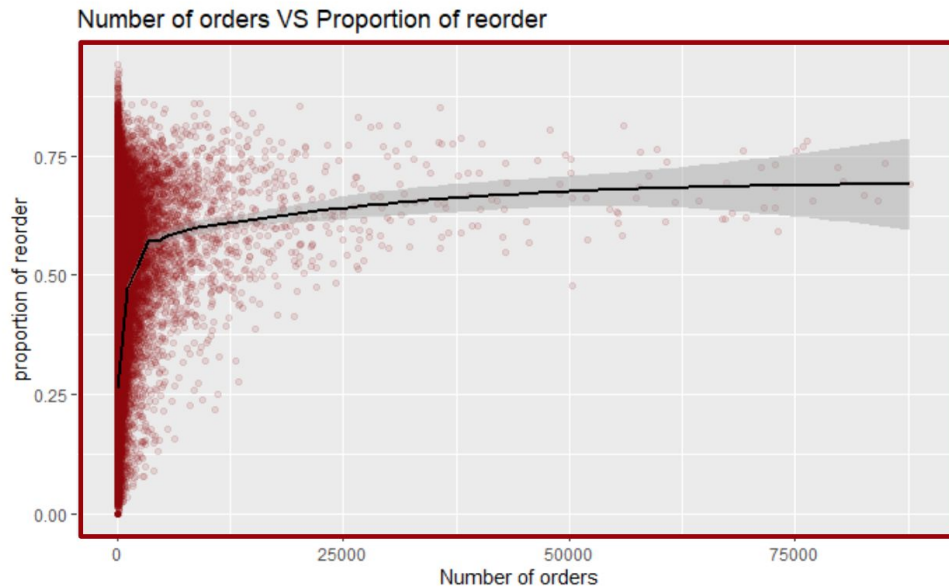


Figure A.4 Proportion of reorders with number of times product ordered

A.2.4 Proportion of products reordered

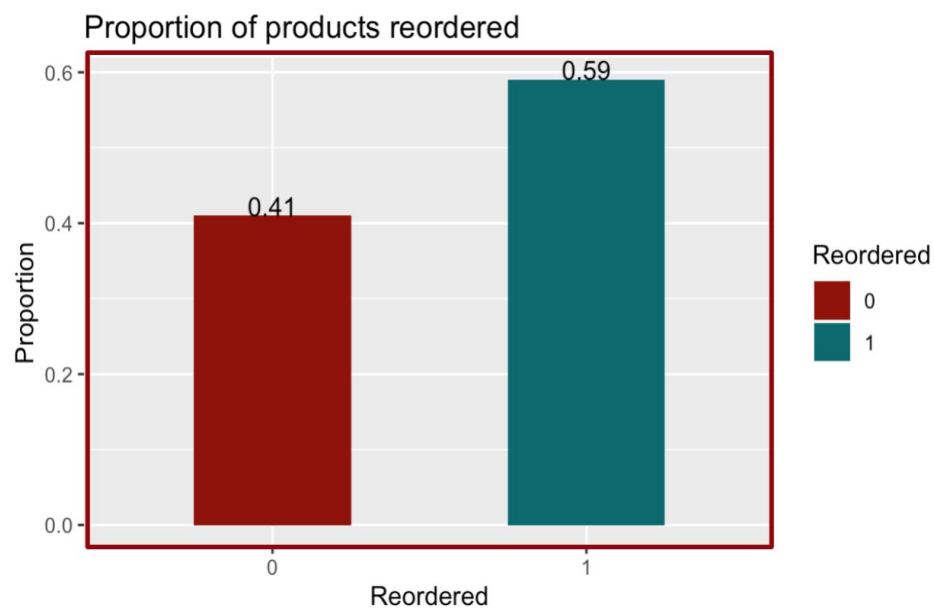


Figure A.5 Proportion of reordering

A.3 Modeling

A.3.1 Binary training matrices displaying more sparsity in user-product matrix

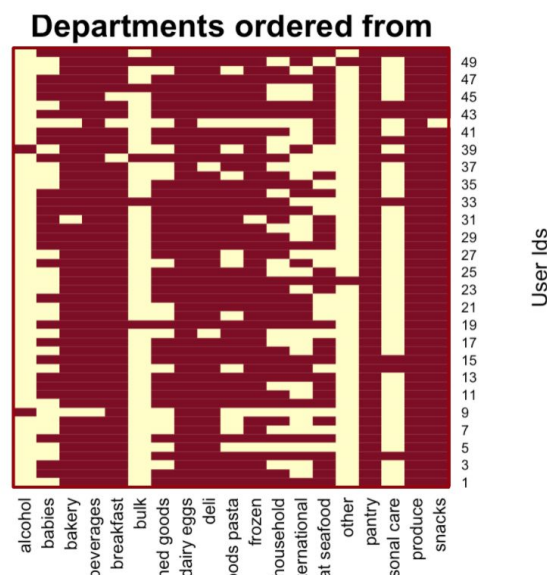


Figure A.6 User-Department matrix

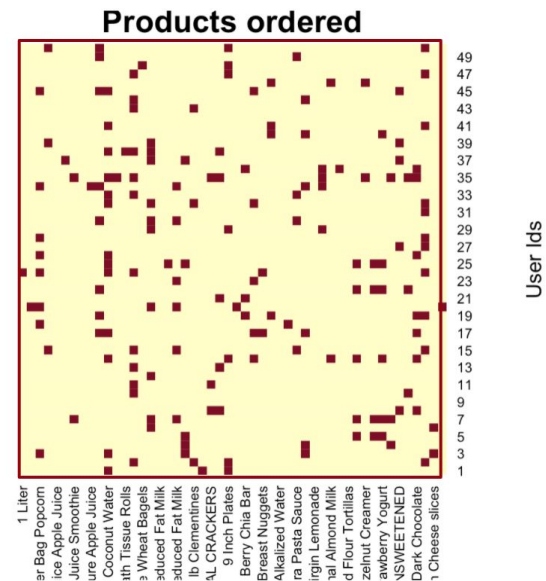


Figure A.7 User-Product matrix

A.3.2 Test results of User-based collaborative filtering on User-Department Binary Matrix

All of the recommended departments were actually ordered from in the test data.



Figure A.8 User-based collaborative filtering results on User-department matrix

A.4 Github Code Repository link

<https://github.com/thakur-ro/Data-Management-and-Processing>