1.0 Name:

Runchen Tao, Xueli Yang, Xuan Shi

2.0 Project Title:

Personalized Product Purchase Prediction based on Purchase History and Customer Group

3.0 Source Data:

https://www.kaggle.com/c/instacart-market-basket-analysis/data

Instacart Market Basket Analysis is a dataset from a Kaggle competition launched by Instacart in 2017. Instacart is an American company that provides grocery delivery service. "The dataset contains more than 3 million grocery orders from over 200,000 Instacart users" (Instacart).

4.0 Project Description:

During the global pandemic in 2020, the demand for non-contact grocery delivery service is thriving. It is vital to Instacart business revenue whether they can provide an exclusive and personalized shopping experience to their customers because many other business competitors in the market are developing similar services. On the Instacart platform, numerous customers who are placing online orders every day. "Whether you shop from meticulously planned grocery lists or let whimsy guide your grazing, our unique food rituals define who we are" (Instacart). Therefore, the ability to discover insights from customer shopping behavior is necessary for producing meaningful personalized shopping experiences for every customer. However, if we collect every single transactional data from each customer, the size of the dataset will be

enormous. In order to provide personalized service to Instacart customers, we will apply multiple data mining techniques to the Instacart dataset to solve the following problems:

4.1 When an Instacart customer is shopping online, how to predict the next item that this particular customer will most likely purchase based on the customer's purchase history?

4.2 If an Instacart customer puts a new product which never appeared in the customer's purchase history into the shopping cart, how to predict the next item that this particular customer will most likely purchase?

5.0 Methodology:

In the following sections 5.1 and 5.2, we proposed two methods to solve problems 4.1 and 4.2.

5.1 Personalized Product Purchase Prediction based on Purchase History

If the current customer puts an item that has been purchased in the past into the shopping cart, we can use the customer's purchase history as a reference to predict the next item that is most likely to be purchased. The purchase history of a customer may imply a product purchase behavioral pattern. We can utilize the behavioral pattern to predict the next item *y* that a customer wants to purchase after a customer *i* puts an item *x* into the shopping cart. However, sometimes the next item *y* that a customer *i* wants to purchase may not be relevant to the purchase history. The next item *y* could be more relevant to the current items that already exist in the customer's shopping cart. In other words, the orders that customers put items into the shopping cart could also be used as a reference to predict the next item *y*. Therefore, there are two approaches:

1. Use the past purchase history of the current customer to predict the next item *y* to be purchased. Here are some algorithms/frameworks we could use:
    a. Decision Tree

   b. Logistic Regression
   c. XGBoost
   d. LightGBM

2. Use the current items in the customer's shopping cart as a reference to predict the next item *y* to be purchased. In this case, we need to consider the order of each item being added to the shopping cart.
   a. Since the items are added by different order and time, we can use time series forecasting methods to make the prediction.

5.2 Personalized Product Purchase Prediction based on Customer Group

If the customer added an item that never appeared in purchased history, we should use another reference to make a prediction. If we can separate Instacart customer into different groups, then the customer purchase behavior inside a group may act similarly. By analyzing the purchase items from all customers who belong to the same group as current customer *i*, we may find the most frequent items *y* that are usually purchased with item *x* from this customer group. Here are some approaches we may use:

1. Customer Group Segmentation
   a. K-nearest Neighbor (KNN)
   b. K-means Clustering
2. Market Basket Analysis (MBA)
   a. Apriori Algorithm to find the most frequent item set
   b. Association Rules Mining
3. Combine knowledge to predict the next item *y* if item *x* is purchased first.

5.3 Association Rule

Association rule is built by using the association between different items in the dataset. Association rules reflect the correlation between one thing and other things. If we know the correlation then we could predict the right items for customers. The purpose for the study is trying to save the customers' time and also let the merchant increase their porfits. For this project, we may use the Apriori Algorithm. Apriori algorithm is an association rule mining algorithm. Using a support-based pruning technique, the exponential growth of the candidate set is systematically controlled, also it is level-wise algorithm, from frequent one item set to the longest frequent item set, it traverses one layer of the itemset lattice each time. Apriori Algorithm uses a generation-and-test policy to find frequent item sets. After each iteration, the new candidate item set is generated from the frequent item set found in the previous iteration, and then the support of each candidate item set is calculated and compared with the minimum support threshold. The total number of iterations required by this algorithm is Kmax + 1, where the "Kmax" is the maximum length of the frequent item set.

Apriori-Gen function generates a candidate item set through two operations.

- Generation of candidate item sets.
- Pruning of candidate item sets.

If one subset is infrequent, X will be pruned immediately. This method can effectively reduce the number of candidate itemsets to be considered during the support count. After calculating the support count of the candidate set, delete all candidate sets whose support count is less than minsup. When there are no new frequent item sets generated, the algorithm ends.