Instacart Customer Purchase Predictions & Analysis

Meghana Bhimasani

Objective:

Our goal is to recommend personalized products to Instacart's customers. The idea behind this is to better appeal to Instacart's customer base as well as improve customer satisfaction and shopping experience. To do this, focus was first placed on analyzing Instacart's customer purchasing habits, before progressing to building recommendation engines. This analysis will help Instacart understand its users' needs and habits, as well as help identify important features to build the machine learning model.

Data Pre-Processing

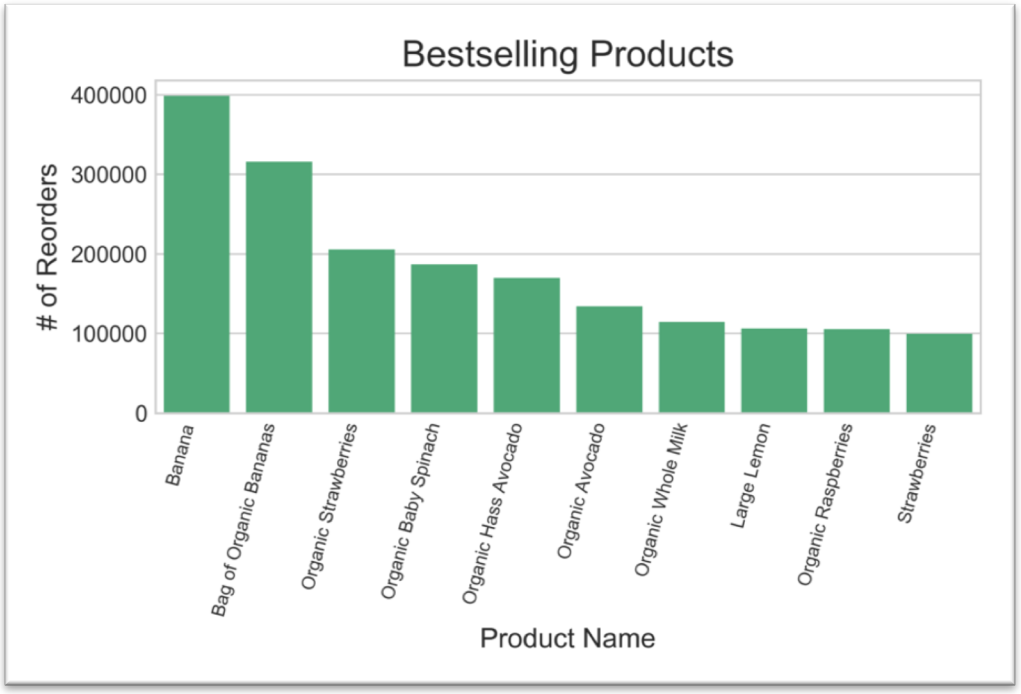
The goal in this process was to ensure that all important features were integers, and to understand the number of rows per dataset and if any values were missing. Fortunately, the datasets on orders, products, and purchase history provided have little errors. The features to note are all in the same data type(int64) and in most of the data sets, there were no missing values. The exception is the null values in the days since last order column in the orders file.[1] However, because these missing values are only for the first order of each user, and are only present in the orders file and not in the training and testing data, its unlikely they will cause errors during data modeling. Lastly, the reorder ratio was explored in the prior order and train datasets to gauge if there is a similar distribution between the two. 58.97% of all orders in the prior dataset contain reordered products, and 59.89% of all orders in the training set contain reordered products.

Data Visualization

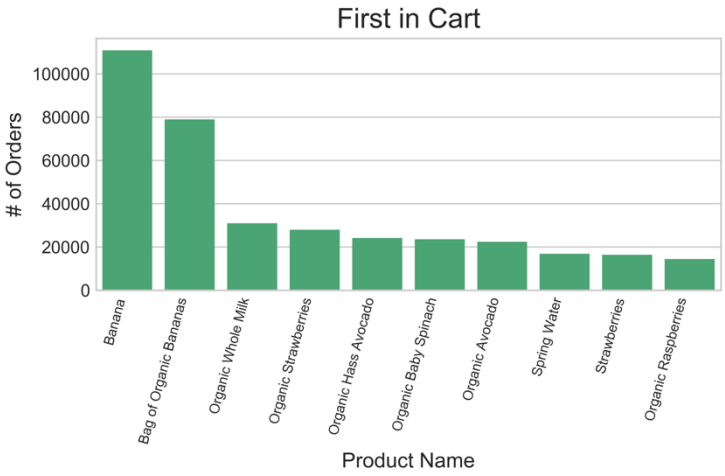
To gain a holistic perspective, the approach to this project was first hypothesizing what factors could possibly influence customer shopping habits. Subsequently, any trends in the types of products bought, times of day/week/month when orders are placed, proportion of reorders, and specific products that are most consistently bought were explored. Again, this is helps to explore what products are in demand as well as understand how customers use Instacart, their needs, and habits.

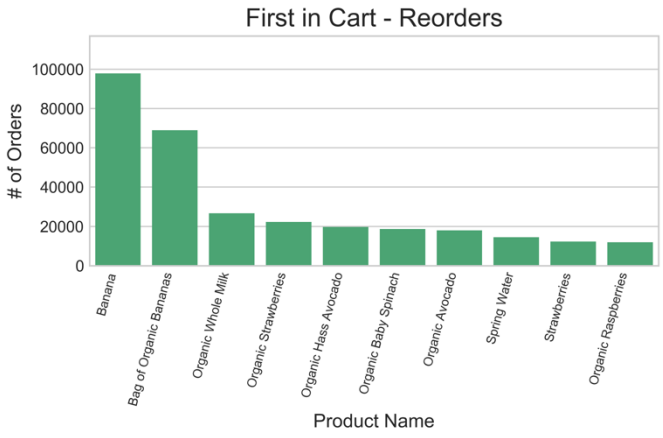
By merging the products and prior orders datasets together, and calculating the total reorders for each product, this report shows that the 10 most reordered items composed of organic foods, fruits and vegetables. Furthermore, bananas and organic bananas are the most popular items purchased. (Figure on next page)

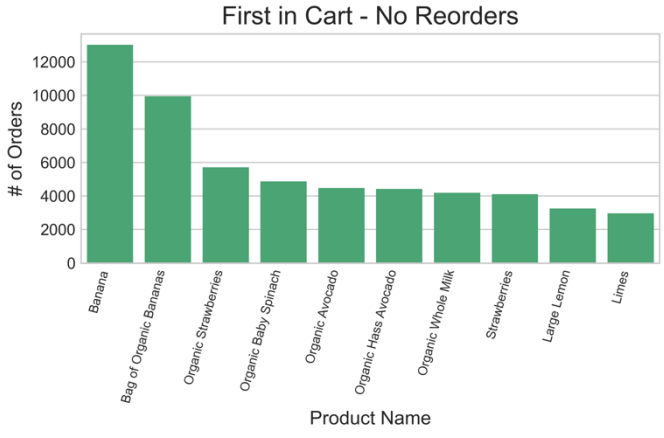
[1] From the orders csv file, I noticed that there were only 3,214,874 values for days since the last order in contrast to 3,421,083 values for the rest of the features, but this is due to Null values present for the first order of each user. I contemplated dropping these null values, but as this was still early in the exploratory process, I decided to leave them for now.



Just to go more in depth, this report also explores which products were frequently added into the cart first, and of these, how many were reorders. Most of the products that were added to customers' carts first were also the most popular products bought overall, and make up roughly a quarter of total sales for that product. Moreover, a majority of products added to carts first were reordered items, that is items that have been ordered by customers at least once before. For example, based on purchase history, roughly 400,000 thousand bananas were purchased, and roughly 110,000 bananas were added to carts first. Of the 110,000 bananas added to the cart first, roughly 979,000 were reorders. This gives a clear conclusion that items that customers add to their cart first are repurchased consistently.







To further understand customers' shopping habits, it's also worthwhile to look at when customers most often placed orders. By merging the orders and prior orders datasets, I explored trends in the amount of orders placed per hour across the week and trends in purchase activity over the month. This report shows that the busiest times were on Sunday afternoon (1-3pm) and Monday morning (9-11am), with a significant decrease during the week. Additionally, customers are placing orders frequently at the end of the week and at the end of the month. I felt that these time features could potentially be useful when building the model.

Data Modeling

Based on this exploratory analysis, important features to take into consideration when training this model would be order ID, days of the week that orders were placed, hours of the day that orders were placed, a products position in the cart (when products were added to the cart), and whether a product was reordered in the past.

I hypothesized that Random Forest Regressor could be an appropriate model to solve this problem as it is able to provide a good balance between precision and over-fitting.

Model training originally started with fitting to the orders training set with the features order ID, position in cart, and reorder status, and test on the order test set. However, it became apparent that this would be not be possible as the number of features in the orders train set (n=3) is different from the number of features in the orders test set (n=1).

Training then shifted instead to fitting the model to the prior orders dataset, since this dataset is composed of historical data and could provide for an accurate model. The prior orders dataset was split into test and train batches to train and test the model. For validation, the model can then go on to predict product IDs in the orders training dataset. After fitting the model on the training batch of the prior orders dataset, the r squared score was determined to be about -0.276, which indicates that this model is not very accurate at predicting product IDs.

Conclusions:

Random Forest was the hypothesized model due to its ability to prevent overfitting and maintain precision. However, Random Forest Regressor proved to not be a good model to help suggest products to customers. With an r squared value of -0.276, the model is not accurate and is not able to find a good correlation between the features to predict products. I believe that more tuning of the model is needed, and one possible way to improve the r squared score for the model is to increase the number of trees. I could also reassess important features for the model's inputs. A few other analyses to help reassess the model would be to explore the relationships between product ID and its position in the cart and between user ID, product ID, and position in the cart. If these steps are do not sufficiently improve the model, then a different model would be necessary. Regardless, to implement this model, it would be necessary to continue tracking customer purchase activity and monitor times (day of the week and hour of the day) that customers shop.

It is still possible to translate insights from the exploratory process into suggestions. Organic foods are frequently purchased, and most customers shop during Sunday afternoon and Monday morning, and at the end of the week and end of the month. As a starting point, organic food would be a good product suggestion to give to all customers.

Resources:

I used a few resources and references while working on this, namely the Titanic Kaggle competition to get started on the data science process, and documentation on seaborn and sk learn libraries.

Data:

7 datasets were explored in order to build a recommendation engine that suggests appropriate products to customers:

1. **orders** (information on all orders with order IDs, user IDs, order numbers, day of the week orders were placed, time of day orders were placed, and time between consecutive orders)
2. **products** (information on products with product IDs, product names, and IDs of aisles and departments the products were stocked in)
3. **aisles** (aisle IDs and aisle names)
4. **departments** (department IDs and department names)
5. **order products prior** (purchase history of customers with order ID, product ID, what position products were added into the cart, and which products in an order are reorders/purchased in the previous orders)
6. **order products train** (training set with same features as “order products prior” comprised on the most recent orders of a subset of users, new orders)
7. **order products test** (testing set with order ID and product ID, new orders)