Grayson Fiveash, Sterling Gutterman, Laney Brown, Caroline Hagood
DS 3001: Foundations of Machine Learning
December 14, 2024

Instacart Project: Final Paper

Abstract:

This study presents an in-depth analysis of Instacart customer behavior. The central question that guides our analysis is, which key variables most strongly influence customer purchasing frequency and preferences? The data originates from the 2017 Kaggle competition "Instacart Market Basket Analysis" by Stanley et al. It consists of three million grocery store orders from 200,000 users. For each user, there are anywhere between four and 100 orders documented. For each order, the sequence of products, hour of purchase, and day of the week are also documented. Additionally, the time between orders for each customer is recorded (Stanley et al., 2017). Initial exploratory data analysis (EDA) observations reveal the frequency of purchases in the dairy and produce departments, common order days being on the weekend, and a tendency to shop during daylight hours. Analysis using single and multiple linear regression, PCA, Lasso, k-Means Clustering, and decision trees examined interactions between the add_to_cart_order, reordered_yes/no, total_order_count, and days_since_prior_order variables. Our results highlight a need to frequently reorder items that go bad quickly, a preference toward making larger orders on Sunday, a tendency to reorder repeat items on the weekends and early days of the week, and a pattern of adding items to the cart based on their relative importance. This study demonstrates how user data can be leveraged to optimize different levels of a consumer experience product. Future work could focus on improving efficiency and generating actionable recommendations.

Introduction:

Instacart is the leading online grocery delivery service, partnered with more than 1,500 retailers and located in 85,000 places. Created in 2012, the platform is now valued at $10 billion (Inklebarger, 2023). In serving over 7.7 million consumers monthly, we were interested in centering the data around behavior we can relate to. Thus, we chose customers as the main observation. The main question that will guide our inquiry is which key variables most strongly influence customer purchasing frequency and preferences? Our supporting ideas include investigating when and how often customers place orders, and from which departments. Additionally, we observed how the order of product selection communicates high-need items.

Our data was initially presented as 5 distinct dataframes, and after merging, we were left with over 32 million data points–representing products in orders. Due to processing constraints with the resources available, we significantly cut the data. In order to maintain meaning in the data, we narrowed in on the habits of frequent shoppers, defined by total_order_count. We will further explore the impacts of cutting the data after presenting our results.

Our findings demonstrated a variety of interesting relationships and patterns of activity in how shoppers place their Instacart orders. While there were over 30,000 unique products and 134 aisles, there were only 21 departments, so we chose to focus our qualitative investigation on that field. Histograms revealed that the dairy and eggs and produce departments were the most often ordered, the highest total number of orders was placed on Sundays, and shopping occurs during normal hours of the day (9-17 hour range), with a peak during mid-morning (10 am). We also explored a variety of cross-tabulations to see variable interactions. We observed a general decrease in items purchased as the time between orders increases, an increase in ordering repeated products on Saturdays and Sunday, and a steady decrease as the interval of time lengthened; however, a significant number of orders were placed on a monthly basis. Lastly, we

computed descriptive statistics on the relationships between variables. The mean days_since_prior_order was relatively consistent by department, with a small increase for the babies department. It was also consistent for order_day, with a higher mean for Saturday and Sunday. There was a larger range of values for mean add_to_cart_order based on department, with the highest being for babies. From these findings, we noticed departments with products that go bad quickly are purchased more frequently and weekends are common shopping days with Sunday being a day to buy products that have been previously purchased.

To further our analysis, we focus on the reordered, add_to_cart_order, total_order_count, and days_since_prior_order variables. In order to identify the relationships between variables in the dataset, we used a variety of methods: single and multiple linear regression, PCA, Lasso, *k*-Means Clustering, and decision trees.

Through linear regression models and kernel density plots demonstrating a distribution of days_since_prior_order organized by department types, we found that the majority of orders were placed in the 0 to 8 day range, with the most frequently ordered departments being dairy/eggs and produce, occurring at an even lesser interval of time between orders. Another single linear regression model that regressed days_since_prior_order on order_dow, which although produced an R^2 value that did not demonstrate a high correlation, produced the highest coefficient of 4.09 for Sunday.

In an effort to support which features of our data were the most impactful in producing patterns in our numeric variables of interest, particularly total_order_count and add_to_cart_order, we performed PCA and LASSO analyses. When looking at which features influence add_to_cart_order, the babies department proved to have the largest coefficient, 0.918, demonstrating that products related to babies were most often placed in customers' carts first. In

addition, in a regression of total_order_count, after using LASSO and one-hot encoding, we found that the produce and dairy/egg departments were the most influential towards total_order_count, with the highest slopes of 0.124 and 0.0275, respectively. This could demonstrate that because dairy, eggs, and produce have shorter shelf lives in the home, customers were likely placing a heightened number of orders in these departments at a higher rate in comparison to other departments.

K-Means Clustering (*k*-MC) was conducted to demonstrate interesting pattern-based relationships between variables of choice in our dataset. Through using the condensed version of the *k*-MC algorithm that sets iterations ahead of time using the order_hour_of_day and order_dow variables, a scree plot to determine the best cluster number which was 2, and an initial scatterplot for comparison, we were able to determine that the reordered_yes/no variable had demonstrable patterns in terms of when customers were placing orders. Based on the comparison and similarities between *k*-MC plot using 2 clusters and initial scatterplot, we found that customers were most often reordering specific products on Saturday through Tuesday at relatively all times, demonstrating that customers could be more likely to reorder repeat products on these days based on weekly routines and a regularly-occurring need for these items.

We used decision trees in an attempt to find the most influential variables in predicting days_since_prior_order, focusing on order_hour_of_day, order_dow, department_id, total_order_count, and aisle_id (decision tree's five features). From this analysis, we determined that at a rate of 84% of influence, total_order_count had the highest effect on how long customers waited between orders.

The remainder of the paper discusses key variables, pre-analysis thinking, results, challenges and implications. Despite finding a range of results, which are described in more

detail below, Instacart engineers may be interested in using these models to understand customers' grocery shopping habits and the factors that influence their behavior.

Data Overview and Key Variables

The key variables of the data were initially split into five distinct dataframes, which were categorized by variables related to customer order patterns–such as order frequency or days since a customer's previous order–purchased products, and product aisles and departments (Stanley et al., 2017). There were 17 total variables across the five dataframes. Order_id, user_id, product_id, aisle_id, and department_id represent unique order identifiers, which we ultimately used to merge the five dataframes. The numerical values are order_number, order_hour_of_day, days_since_prior_order, and add_to_cart_order, representing the position out of total orders per customer, the hour of day the customer placed an order, the number of days between customer orders, and the order in which a product was added to the cart, respectively. Order_dow and reordered are coded with numbers; for the day of week, 0 represents Saturday and 6 represents Friday, and for reordered–whether or not a product has been previously purchased–0 corresponds to no, and 1 corresponds to yes. Lastly, product_name, aisle, and department represent information about the products (Stanley et al., 2017). After cutting our data, described later, we added a variable called total_order_count, which indicated a customers' total number of Instacart orders.

The first step in cleaning the variables is to identify any missingness and replace these values with 'nans.' For example, through examining the first five rows of the dataset, the variable days_since_prior order contained a lot of "NaN" values. In attempting cleaning this variable, we coerced it to numeric and tried to replace any potential missing values with a nan

value. Coercion of this variable to numeric and replacement of missing values with nans did not change the total count of the variable, however, and when the total number of missing values for this variable was calculated, the result was 2,078,068. This is because missing values seem to already be classified by "NaN," so there are no other blank or null values for this variable that could have been replaced.

Another variable that we cleaned was the initial numeric order_dow variable. The days Saturday through Friday, chronologically, were originally classified as 0 through 6. We added a new column called order_day to the dataset that associated and replaced each number with its corresponding date. We also cleaned the initially numeric reordered variable, as an observation of 0 corresponded to the product not being reordered, while an observation of 1 corresponded to the product being reordered. We added a new column called reordered_yes/no that associated and replaced the 0s and 1s with their corresponding no and yes.

Plots and Descriptive Statistics Tables:

We conducted some initial Exploratory Data Analysis (EDA) after merging the overall products and orders datasets to gain an understanding of their respective patterns. To start, we examined the shape, types of variables, and column names to get an initial sense of the overall dataset. For product-related variables, we examined the frequency of department distributions specifically, first using .unique() and .value_counts() to observe the number of unique values as well as frequencies of department types, and visualized this using a histogram, from which we observed that "produce," "dairy eggs," and "beverages," were the three highest occurring product department categories. We then examined aisles using .unique() and .value_counts(), also visualizing this variable related to products with a histogram.

For orders related variables, we initially examined the shape, the types of variables, and the column names initially. We looked at the .value_counts() and .describe() results of the order_day new column to see the most popular days and also visualized this through a histogram; Saturday and Sunday were the most popular days. We also made a kernel density plot of the original order_dow variable, which produced the same results. In addition, we created a histogram plotting the order_hour_of_day variable, with the most popular hours of the day being in between the 9 to 17-hour range. We also made a boxplot of the add_to_cart_order variable, with the 50% quantile being seen around the values 6 to 7. Finally, we created a cross-tabulation between the reordered_yes/no variable and the order_day variable. From this cross-tabulation, we observed that Saturday and Sunday were the days that repeat orders were most likely to occur. After gaining a general understanding of the data, we considered how to organize it effectively and identified the key insights we aimed to uncover.

Methods & Analysis Plan

We focused on supervised learning, which relies on identifiable datasets and predictor variables to produce an identifiable result and to optimize algorithm efficiency and correctness (Delua, 2021). First, we used regression, which produces a numeric prediction result, in our predictions around the days of the week and times unique customers are most likely to order, as an example. On the other hand, classification, which produces a categorical prediction result, was used in our categorical predictions, such as which products unique customers are likely to repeatedly order. We made predictions using decision trees, focusing on potentially identifying any elements of non-linearity. After finding R^2 and mean squared error results, from regression/classification and decision trees, we will make comparisons and determine which

findings are the most valuable. When we explore non-linear relationships, such as days_since_prior_order and reordered_yes/no, decision trees may capture the variance of predicted variables better than traditional regression or classification.

While we focused on supervised learning, we explored how PCA, an unsupervised learning algorithm, impacts a model's performance by reducing multicollinearity and avoiding overfitting. We will compare instances of linear regression and classification models, evaluating their performance both with and without applying PCA before analysis. We hope the models will perform differently, given how multicollinearity can cause overfitting and skew predictor results. Multicollinearity may arise between highly correlated variables, such as order_day_of_week and order_hour_of_day. To be addressed later, After facing difficulties in finding meaningful results from PCA, we pivoted to Lasso, which aims to similarly reduce multicollinearity and identify the most powerful predictors.

For our analysis, many categorical variables need to be one-hot encoded, or transformed into numeric variables, to perform linear regression. Some of these variables include product_name, aisle, and department. Once these variables are transformed, we will be able to run the models. For regression, the two numeric variables that can be predicted are days_since_prior and add_to_cart_order. For the former, order_day_of_week, order_hour_of_day, or order_number may all influence why a customer's purchase sequence is the way it is. We can also run a regression of days_since_prior on the department to see if two orders are close together due to different grocery store categories. When predicting add_to_cart_order, product, department, and aisle will be the most relevant variables. For classification, we can perform regression on categorical variables, such as reordered_yes/no, order_day_of_week, aisle, and department. As an example, features like order_day_of_week,

aisle, and days_since_prior_order might influence whether or not a customer reorders a product (reordered_yes/no). We will most likely encounter multicollinearity in both regression and classification and principal component analysis, or PCA, will be useful to tackle highly correlated variables. Additionally, we will use *K*-Means Clustering (*k*-MC) as our final unsupervised learning strategy, which applies well to an extensive dataset like ours. This can help us identify patterns in customer behavior, identifying commonalities among order_dow,, order_hour_of_the_day, or preferred department/aisle. We will select k, using the elbow point method, to be the centroids, apply maxmin normalization to the variables of choice, likely variables that we seek to identify relationships between such as user_id, order_day, days_since_prior_order, reordered_yes/no, and others, and conduct *k*-MC, performing multiple iterations.

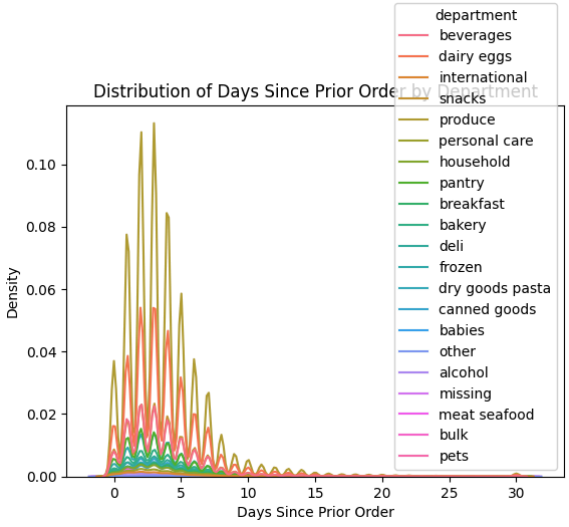
We will know if our approach works when we can produce an identifiable pattern in our predictions. Success for our project will mean being able to make conclusions about which customers are most likely to order which products based on unique product identifiers and reordering data and when they are most likely to place these orders based on days of the week and times of day. Success for regression and classification is determined by R^2. Success is most possible when we remove any outliers, ensure that the sum of squared error (SSE) is the smallest that it can be, and ensure that we are not underfitting or overfitting with our model. The closer the value is to 1, the better fit the model is. Lower and negative R^2 values indicate severe over-fitting; however, they are not always poor outputs because they can still communicate observations about the data. PCA is a method used to combat collinearity, where the number of variables is reduced but information is maintained (IBM, 2023). We will define success with PCA as obtaining the least sum of squared error values. Success with *K*-means clustering will be

observed in identifying relationships between variables that we did not previously hypothesize about and minimizing the SSE by choosing the best cluster value. In terms of presenting our results, we could produce a table of regression coefficients after performing linear regression. PCA is often displayed in a scatter plot to show the relationship between components.

Regression and Classification (Pre-PCA and Lasso)

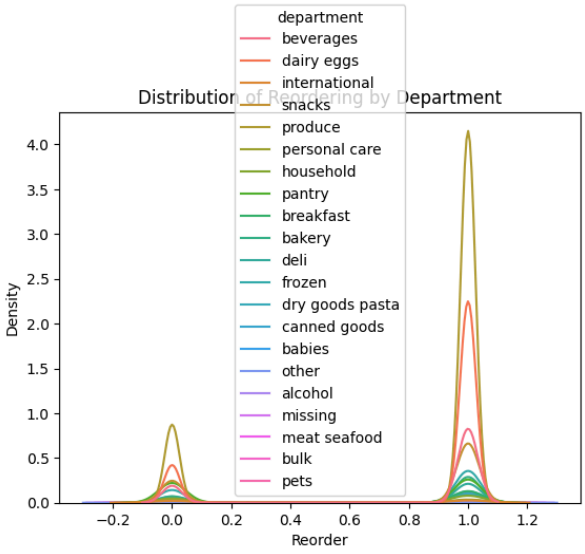
To begin our exploration of the data, we used linear regression models. We chose this model as it seeks to explain which variables are related to each other. Since our data has the potential for many unique explanations it was important to try out a variety of combinations. Our first step was doing more EDA. We ran various cross-tabulations, group-by statements, and visualizations to understand the relationships between different variables. These early results gave us ideas about what we wanted to further explore. Then we created kernel density plots. This type of plot is organized by peaks, where tall peaks communicate concentrations of the dependent variable and wide peaks communicate variability in the dependent variable with regard to the independent one.

The plot to the right displays how days_since_prior order is distributed across different departments. Since most of the peaks are centered over 0 to 8 days_since_prior order, there is low variability in the range of possible ordering times. In other words, customers shop on a bi-weekly to weekly basis. However, the height of the produce and dairy/eggs peaks indicate that a smaller ordering interval is concentrated in those



departments. These departments may be especially high because they house products that go bad more quickly. Next, we ran a single linear regression model regressing days_since_prior order on department. The R^2 value was 0.001 indicating 1% of the variance can be explained by the model–essentially there is no correlation between the two variables. The coefficients of the model ranged from 3.98 for babies to 3.23 for alcohol. In general, department-specific behavior around reordering is not significantly varied. However, it is interesting to note that alcohol is associated with a lower value, as it is a pastime for many. These results were expected because it makes sense that more than just the department of a product would impact how often a customer makes a purchase.

Next, we looked at the distribution of departments in relation to whether or not products were reordered. Again, the height of the produce and dairy/eggs departments indicates more concentrated reordering. The width is irrelevant as the variable is binary. For the model regressing reorders on departments, the R^2 value was 0.045, indicating 4.5% of the model can be explained. While this value is slightly higher, it still reveals a lack of correlation between the two variables. The coefficients ranged from 0.83 for produce and 0.49 for personal care products, showing produce is concentrated is more likely to be reordered and there is less of a pattern in consumer behavior for personal care products. The last single linear regression model we ran was regressing days since prior order on order day of the week. The R^2 was 0.007 indicating no correlation. However, it



R-squared: 0.007362401986344 08

	variable	coefficient
0	friday	3.888866
1	monday	3.802638
2	saturday	3.995510
3	sunday	4.092941
4	thursday	3.466639
5	tuesday	3.453142
6	wednesday	3.409978

was interesting to see that the coefficients ranged from 3.41 to 4.09, with the highest being for Sunday. This shows that customers who placed Instacart orders on Sundays had the longest periods of time in between orders, possibly indicating that Sunday orders are more likely to be planned or larger orders, compared to more frequent orders placed throughout the week.

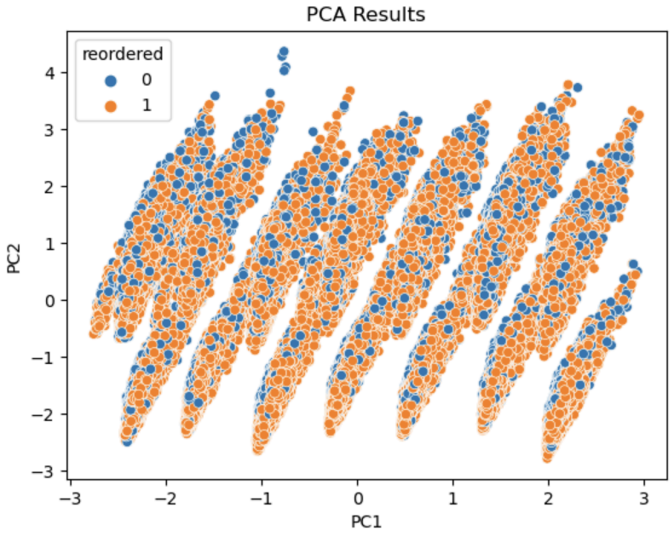
It is clear that these models are under-fitted because one variable can not capture complex relationships. Thus, it was time to explore multiple linear regression. We shifted to looking at classification and chose reorder as the dependent variable because we are most interested in understanding reordering behavior in customers. After regressing reorder on department (one hot encoded), order day (one hot encoded), days_since_prior order, and add to cart order, the R^2 value was 0.095. Since the R^2 value increased each time we added a new variable in comparison to the isolated single linear regressions, we were able to determine multicollinearity is present in the data. Multicollinearity is a common phenomenon in data sets that occurs when more than one variable is correlated with another. When these variables are regressed on each other, they cancel out and can not accurately predict the correlation between them. PCA (principal component analysis) and LASSO (Least Absolute Shrinkage and Selection Operator) are two methods for reducing multicollinearity. The former transforms features into a separate set of uncorrelated features, while the latter reduces the coefficients of insignificant features.

PCA

Based on our results from the multi-linear regression, we chose to perform PCA on the reordered variable. The first step is to run the multiple linear regression on select features to provide a baseline R^2. While PCA accounts for all possible features when determining which independent variables may influence the dependent variable, we chose the most relevant

features–excluding variables like product_id or departmet_id, which would not affect our results–due to runtime errors. We also chose to split the data into train and test sections to evaluate how future data might perform. We calculated an R^2 of -3.33 for the train set, indicating severe overfitting of the model and multicollinearity between variables. Before we could apply the PCA model, we had to pick the number of components to perform the decomposition on. The number of components is related to the number of independent features and visualizing a scree plot shows how much variation each component explains. The elbow point indicates where a larger number of components stops explaining variation in the data. Our elbow was sharply at 2, so we created a PCA model using 2 principal components.

After determining the number of components, we ran the PCA transformation on the train and test sets. To our surprise, the R^2 value decreased to -3.37. While we expected this result to increase, we ultimately concluded that there is more to the story. The graph pictured to the right illustrates the first two components of reorders in a scatter plot, with 1 (orange) indicating a customer reordered a product and 0 (blue) indicating a customer did not reorder a product. The x-axis is the first component and the y-axis is the second component. Unfortunately, this graph illustrates no relationship between the components and consumer reordering behavior, as the goal is to separate the two components visually and determine the magnitude of the two components. The orange and blue dots are quite interspersed, suggesting neither component has strong predictive power in determining whether a



customer reorders. While it is interesting that the data follows a continuous zigzag pattern, more successful PCA results would show clear clusters of instances. Clearly, neither component captures the variability or predictive patterns in reordered behavior.

LASSO

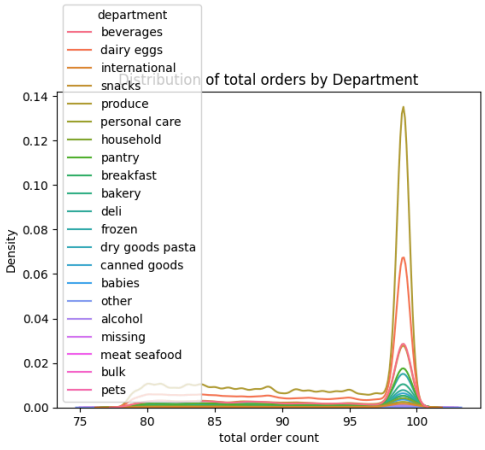
Since we did not identify meaningful conclusions from PCA, we ultimately decided to try LASSO for feature selection. We hoped that LASSO would pinpoint the features with the most influence on key numerical variables. LASSO performs better on continuous variables so we pivoted from reorder to focus on other key numeric variables. In our analysis, we focused on add_to_cart_order, which represents the order in which a customer added a product to their order, and total_order_count, which is the total number of orders a customer placed in the 30 days observed.

First, we created a model displaying add_to_cart_order regressed on select features. While regressing on all variables in the data set would provide the best results, we similarly chose the same relevant features as our PCA model due to runtime errors. In the case of Instacart, we hypothesized that customers may place their most prioritized or top-of-mind products into their cart first. However, since product_name has 31,357 unique entries, our model did not have the capacity to process such a large number of products. So, we excluded product_name from our list of features and instead, since departments essentially capture the same information, we used department. By shrinking non-relevant features to zero, this model identified the most important features. The pictured table shows the feature with the highest slope is babies, with a slope/coefficient

	variable	slope
0	order_number	0.004207
1	days_since_prior_order	0.235986
2	order_hour_of_day	-0.021183
3	order_dow	-0.054817
4	reordered	-1.211459
5	total_order_count	-0.001457
6	friday	0.000000
7	monday	-0.402037
8	saturday	0.000000
9	sunday	0.060852
10	thursday	0.068638
11	tuesday	-0.385121
12	wednesday	-0.188231
13	alcohol	-0.463989
14	babies	0.917915
15	bakery	-0.000000
16	beverages	-0.804052
17	breakfast	0.393952
18	bulk	-0.000000
19	canned goods	0.261164
20	dairy eggs	-0.744989
21	deli	0.000000
22	dry goods pasta	0.332854
23	frozen	0.000000
24	household	-0.052272
25	international	0.000000
26	meat seafood	-0.000000
27	missing	0.000000
28	other	-0.000000
29	pantry	0.030761
30	personal care	-0.000000
31	pets	-0.000000
32	produce	-0.409442
33	snacks	0.652613

of 0.918, showing products related to babies have the strongest influence on our add_to_cart_order prediction. Since add_to_cart_order represents the order in which customers put items in their cart, the LASSO results illustrate how customers frequently put baby products in their cart first.

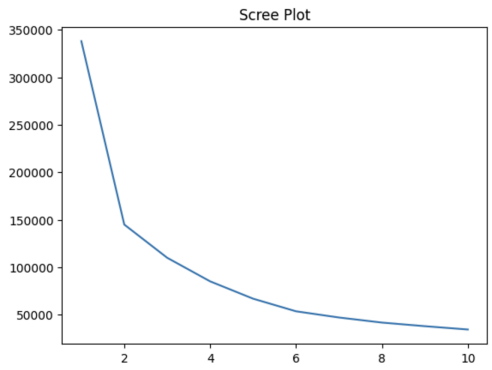
Secondly, we created a model that displays total_order_count regressed on key features. As mentioned above, we had to remove non-relevant features, such as product_id, due to runtime errors. After one hot encoding and running the LASSO model, our results illustrated that produce and dairy eggs–two departments–had two of the highest slopes. Produce has a slope of 0.124 and dairy eggs has a slope of 0.0275. Although neither coefficient is particularly large, their values being above zero indicate that both features contribute to predicting the total_order_count. So, in this model, as customers order more items from the produce or dairy eggs departments, their total order counts tend to increase as well. This may reflect how, since produce and dairy/eggs spoil faster, customers are ordering from these departments more often, and therefore, have a higher total order count. When performing EDA in earlier stages of our project, we noticed that produce (in yellow) and dairy eggs (in orange) have the highest density of total orders, as you can see in the kernel density plot to the right. This works to support our LASSO results that customers who have a larger total order count tend to purchase from the product and dairy eggs department the most.

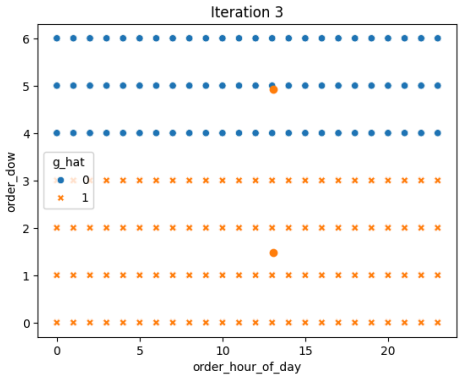


_K_-Means Clustering

Through *k*-Means Clustering (*k*-MC), an unsupervised algorithm method that focuses on patterns rather than a singular predictive answer, we sought to identify relevant patterns indicative of relationships between key variables in our dataset. The initial attempts at the standard *k*-MC algorithm with order_hour_of_day and order_dow were unsuccessful due to long runtimes. Because of this runtime error, the condensed version of the *k*-MC algorithm, with pre-fixed iteration numbers, was used instead, which was successful. We originally hoped to find a pattern between order_dow, order_hour_of_day, and departments. While the initial scatterplot did not illustrate any clear relationships between the variables of interest, the *k*-MC plot had some clustering. Thus, we decided to make a scree plot that would minimize the sum of squared error for the model.

Again, a scree plot helps identify the most efficient *k* value for the number of cluster choices that minimize the sum of squared error of the *k*-means clustering model, which can be found at the elbow point. This was observed to occur around 2, as seen in the left plot below. Since the initial *k*-MC was produced with a cluster value of 4, and we did not find a meaningful result, and we wanted to ensure that the sum of squared error was being minimized, we conducted another round of *k*-MC, this time with a cluster value of two.

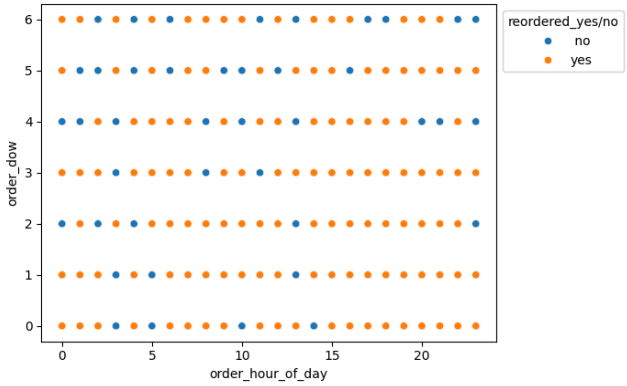




When the *k*-MCalgorithm was applied with a cluster value of 2, the plot to the right above was produced. In an attempt to conduct hierarchical clustering to further determine

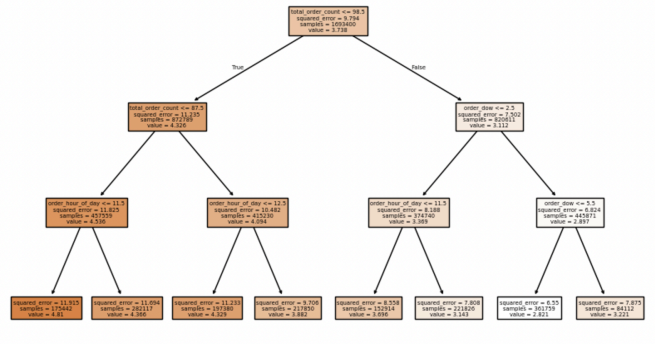
patterns, the runtime unfortunately crashed. Because of the difficulty in determining an observable pattern between the original scatterplot and the 4-cluster *k*-MC for the relationship between department, order_hour_of_day, and order_dow, we created another standard scatterplot to see if there was potentially a relationship between the reordered_yes/no variable, order_hour_of_day, order_dow. When this standard scatterplot was created, as seen below, it closely resembled the clustering seen in the *k*-MC plot when the number of clusters was reduced to 2.

Based on the comparison between this scatterplot and the *k*-MC plot, we can determine that there is a relevant pattern between order_dow, order_hour_of_day, and reordered_yes/no. It seemed that reordering of products most often occurred on the weekend and earlier days of the week, those being Saturday, Sunday, Monday, and Tuesday, at relatively all hours of the day, while customers were more likely to not be reordering products later in the week, Wednesday through Friday. This could be because customers who reordered products early in the week did not need them again later on in the week. Later in the week, customers might be more likely to order different, or even less routinely purchased, products that they did not previously reorder earlier in the week. This could be indicative of broader patterns in customer ordering, as customers might be more likely to place orders on the weekend and early days of the week, reordering repeat items from previous orders in a routine manner, as they are likely by that time running out of the repeat products they ordered the previous weekend.



Decision Trees

For our decision tree analysis, we aimed to identify the features that most strongly predict days_since_prior_order to understand patterns in time between orders placed by customers. The five variables used as features for the decision tree were order_hour_of_day, order_dow, total_order_count, aisle_id, and department_id. Our mean squared error, which explains the difference between predicted and action values, was 9.3. This means that, on average, the model was 9 days off from the actual number of days_since_prior_order. Our mean absolute error was 2.02, showing that our model's average error in predicting reorder intervals is relatively low. Since our mean absolute error was significantly lower than our mean square error, we can infer that there are potential outliers in the data that inflate the squared error. The total_order_count was the dominant feature for prediction with 84%, which means that customers who have placed more total orders tend to reorder more frequently. Order_dow and order_hour_of_day also had some impact on the output, explaining 7% and 8% respectively, but they had significantly less influence than total_order_count.



This analysis highlights that the frequency of customer orders, as measured by days_since_prior_order, is more closely tied to the cumulative number of orders over time rather than specific days or times of the week. Unfortunately, the analysis found that departments and aisle-level variables had little meaningful impact on whether an item was reordered compared to the other features.

Conclusion:

In conclusion, our analysis of the EDA and predictive model results uncovered interesting patterns within the dataset–centered around the customer. From the onset, we hoped to identify which key variables most strongly impact customer purchasing frequency and preferences and investigate trends in purchasing behavior. Our EDA results revealed key information on the times and days of week customers typically make Instacart purchases, as well as from which departments. As mentioned, Saturday and Sunday saw the highest order volumes, and the produce and dairy/eggs departments were the most popular of the 21 departments. After performing EDA, we shifted our focus to the reordered, add_to_cart_order, total_order_count, and days_since_prior_order variables. In performing regression, we identified that customers purchase from the produce and dairy/eggs departments with shorter intervals between orders, and customers placing orders on Sunday had the longest time interval before placing an order again. Next, we utilized Lasso to reveal that the babies department has the strongest influence on the order in which customers add items to their cart, as well as a strong relationship between the produce and dairy/eggs departments and total order count. In the hopes of pinpointing when customers place orders most frequently, *K*-means clustering uncovered how customers reordered products most often on the weekend and earlier in the week–such as Saturday, Sunday, Monday, and Tuesday. Our decision tree analysis demonstrated that a high frequency in customers' placing of orders was the most influential in determining the length of time between orders. Despite these meaningful findings, our analysis of the Instacart dataset faced certain challenges.

The first limitation we encountered was the mere size of our dataset. The original merged dataset contained roughly 32 million data entries, each representing a product in an order. In total, that constituted 3 million orders made by 200,000 users. In order to work around runtime

and capacity constraints in Colab and Github, we ultimately decided the best solution was to significantly, but strategically, cut our dataset. Since our observation and focus of analysis is the customer, we decided to cut the data by customers who placed the highest number of orders, essentially ordering the dataset by order_number, which represents an order's position out of the total orders per customer. In satisfying the GitHub capacity requirement, we were left with 2.4 million data points on 2,900 users whose total number of orders ranged from 79-94. This data is captured in a new variable, total_order_count, as mentioned above. We initially hoped that despite significantly reducing the amount of data in our dataset, focusing on the upper range would reveal interesting insights into frequent customer habits. While our results revealed interesting findings, the constraints on our data set did affect the power of the results, as it eliminated information that could provide insight into more typical purchasing behavior and help our models identify the most powerful predictors for select target variables. For example, in cutting our data, we noticed a difference in how the variables were skewed between the two sets, which could have led to our dataset not being as representative as its full potential. However, we do not think that we could have cut the data in a different way to mitigate this, and we were able to perform successful analyses and use techniques that demonstrated interesting patterns in how customers make choices during the Instacart experience.

Furthermore, we also faced challenges while performing PCA on our dataset. We considered PCA successful if our R^2 improved after running PCA on our linear regression models. However, after working through multiple regression models, we decided to proceed with PCA on the variable that resulted in the highest R^2: reordered. However, PCA did not improve the R^2 when compared to the initial regression. The goal of visualizing our PCA results was to separate the two components and determine the magnitude of each, highlighting how the

components are correlated to the data, but neither component had strong predictive power in determining whether a customer reorders. Again, due to how the data was cut, our understanding of the full picture was limited. As a result, PCA did not have all the information it could have had to determine which features had the biggest impact. This was a meaningful experience in learning about how manipulating the data can impact the results. If possible, it would be interesting to run all our models on the full data set to see if stronger results could be produced.

Additionally, in future data analyses of this type of consumer behavior, we think it would be interesting to gather more demographic information on the customers. For example, identifying diet, age, household size, and gender differences could reveal why certain product groups, such as baby products, were more prevalent. We would also like to further explore how certain products and their characteristics correlate to and influence certain customer behaviors, as we saw multiple results indicating this with dairy and produce.

Works Cited

Delua, J. (2021). *Supervised versus unsupervised learning: what's the difference?* IBM. https://www.ibm.com/think/topics/supervised-vs-unsupervised-learning.

IBM. (2023). *What is principal component analysis (PCA)?* IBM. https://www.ibm.com/topics/principal-component-analysis.

Inklebarger, Timothy. (2023). *A brief history of Instacart, from startup to planned IPO*. Supermarket News. https://www.supermarketnews.com/grocery-technology/a-brief-history-of-instacart-from-Startup-to-planned-ipo.

Stanley, J., Risdal, M., Sharathrao, & Cukierski, W. (2017). *Instacart market basket analysis*. Kaggle. https://kaggle.com/competitions/instacart-market-basket-analysis.