What Are We Eating?

An Analysis of Online Grocery Purchasing Habits

Evan Steiner, Jason Stock, Keegan Millard
Introduction to Distributed Systems
Colorado State University

1. Introduction

Food is an integral part of our lives and as such should be explored deeply in all aspects. One of the primary questions that is on all our minds when thinking about food is: “What should we eat?” The answer is often ill-defined and subject to the needs of the individual, but can be generalized to fall between a healthy and unhealthy option. Making the right choice can be overwhelming with the mass amount of products available in grocery stores. The rise of e-commerce and an increase in data collection means that companies can use computer analysis help consumers find healthy foods and make the right choice of what to eat.

In 2012, the North American company, Instacart, was founded with the mission to offer same-day delivery of food products from local retailers. Essentially creating a 50,000 square foot store that can be viewed from a mobile phone application, or web-browser. Having an online presence made it simple for customers to browse and order items with ease. Instacart monitors not only the products and sales, but also the users. Tracking their purchases to find spending patterns and likely to find insights into the population as a whole. This opens new doors for marketing to specific individuals, gaining a better understanding of their habits and with those habits a more complete understanding of the food market. In fact, external datasets related to food nutrients and ingredients could be used to complement findings and conclude if the decisions made by these customers are those in their best health interest or not.

Developing a greater understanding of how customers behave is important for many domains. One of the most prominent interests coming from financial markets. Having a profile for various customers can strengthen an organization's business canvas and revenue models [1], thereby maximizing profits by studying recurring trends and purchasing habits. Furthermore, analyzing these habits could give insights into a population’s standard of living, as it has been shown that an individual's nutritional and dietary strategies have direct effects on their quality of living [2]. As to contribute towards this research, we seek to further analyze the 2017 Instacart dataset [3] in conjunction with data from the United States Department of Agriculture Food Composition Database (USDA) [4].

Using the two datasets, our team will be able to conduct experiments that address the aforementioned claims by addressing and attempting to solve a variety of questions with

differing analytical approaches. We will begin by analyzing the nutritional guidelines of food products purchased by users to measure how healthy the populous is. Specifically comparing the sugar composition of products purchased by users of Instacart and the overall sugar content from all food products available on their platform. Thereafter, we will determine what sort of ingredients and nutrients make up the foods that we eat and categorize food based on those ingredients and nutrients. Finally, we will recommend alternative products, to the user's of Instacart, by examining the product ratings of similar users. This highlights the research of customer habits to generalize assumptions of the population and produce insightful information.

2. Problem Characterization

The Instacart dataset exposes 3.5 million orders from 200,000 users over a two year period. The dataset is highly anonymized to protect Instacart's retailers and users. Only products that have been purchased by people from the various retailers are included. Making it impossible to determine the brand or size of most items because only a simplified product name and general isle and department categorization are given.

A user is identifiable by randomly generated id which is linked with a sequence of orders. The order in which products have been added to a user's cart, and whether an item was purchased previously is available. The exact date of orders are not revealed, but a number representing the day of week (the numbers corresponding to specific days day is not specified), time of day and days since last order are given. Further information describing the data available from Instacart is available here [5].

The second dataset we used is the Branded Food Product Database (BFPD) from the USDA. It contains data on over 260,000 food product from 20,000 manufacturers. The information provided is the general information required on product packaging; including, a universal product code, a list of ingredients and nutrition facts presented from BFPD public-private partnerships.

In order to analyze the nutritional choices of Instacart users we planned to link products in the Instacart dataset to similar products in the BFPD. Due to the Instacart dataset being sanitized, finding exact matches between products in the two is not possible, and it is necessary to use an algorithmic approach to make best approximations of similar items based on product names. Instacart included non-food items, which do not have a good match in the BFPD and were removed from consideration.

Once the two datasets are linked satisfactorily, it is possible to analyze the nutritional choices of Instacart users. Interesting inquiries we identified, include: whether there is an association between time of day and nutritional preference, whether certain nutritional compositions tend to be added in a user's cart differently, i.e, healthy food first, and poor nutritional value last, and what is the distribution of sugar consumption for Instacart users.

Additional areas to explore include product recommendations for Instacart users, and an analysis of ingredients and nutrients of the spread of products available in the BFPD. Product recommendation poses a significant challenge due to the large number of products and users in the Instacart dataset. It is possible that having too many ratings could lead to sparse recommendations. Therefore, we will have to find an appropriate method to represent the 10 billion ratings using a common method of collaborative filtering know as matrix factorization. The above analyses focus heavily on the users to better understand behaviors. However, the composition of the foods themselves could pose for interesting findings.

Analysis of ingredients in the BFPD is challenging as the ingredient list is taken straight from the human-readable ingredient list found on the back of products in stores. The names of ingredients used in each list are non-standard, and while ingredients are listed by amount in descending order, the exact amounts are not given. By comparison, the situation regarding nutrient information is more ideal since the breakdown of key nutrients are given for every item and are listed by weight and serving size.

3. Dominant Approaches

There are numerous ways to approach a recommendation system but most approaches can be put into one of two categories, content filtering and collaborative filtering. Content filtering relies on having descriptive metadata for the items within a dataset - without which results in very poor recommendations. This approach is often used for recommending documents, web pages and other items that have lots of auxiliary data available to use for recommendations. Content filtering is very effective even when your data does not contain any user preferences [5], making it ideal for situations where your users have lots of activity on a variety of items, but do not, or are unable, to rate them. While this technique can be very effective in these scenarios, it will not be effective in creating recommendations for our food data - which does not have highly descriptive metadata.

The other sector of recommendation algorithms, and the one that is experimented with here, is collaborative filtering. Collaborative filtering is an approach that is effective when your items do not have descriptive metadata. Rather we have access to user ratings for a sufficient number of individual items. In this approach we create a sparse user item matrix where each row corresponds to a user and their rating for each product in our set, although they likely will not have reviewed/rated everything. Users are then recommended items liked by similar users but not yet rated themselves. This approach is highly effective when there is little content to describe items [6], as well as being effective on large datasets. The primary issues with this approach lie in situations where we have access to very little data, both in users and in items. In these cases collaborative filtering algorithms are unable to recommend effectively. It also has issues when users have rated only a small percentage of the items in the dataset. In which cases, the algorithms have trouble correlating similar users. Despite the downfalls in these areas, it is still a

highly effective approach in many situations and as such, has been, and still is, used by a number of high profile companies such as Netflix and Amazon [6]. Therefore collaborative filtering is always an approach that should be considered if the data meets the prerequisites as our data does.

Other inspirations to our approach come from a 2017 Stanford poster [7], in which the students tested the accuracy of many different machine learning techniques for recommending products to users. They tested a variety of models including logistic regression, support vector networks, and neural networks. Accuracy for all networks tested lied in a range from 10 to 30 percent [7]. This did not seem great to us, so we wanted to try different approaches from those defined in their paper and take into account some of the issues they had. One of their primary issues was that the data is extremely sparse and that 99% of the products appear in less than 1% of the orders [7]. We were able to somewhat remedy this issue in our algorithm by only using the top N most frequent users and only including the products they purchased in their purchase sets.

4. Methodology

Utilizing data from Instacart and USDA introduced a challenging problem early on. It was important to combine the two together, but there was no described way to do so. Both had titles relevant to the products they were describing; that is, a label was assigned for each unique item, but was formatted according to the specific dataset. Perhaps this could be leveraged in joining the data by finding similar item labels. Each of these labels are strings that could be split to represent a set of words. Therefore, the question of finding the similarity of two labels can be described as finding the most similar sets. A common mathematical approach to this is expressed by the Jaccard index [8]. Measuring the overlapping space of two finite sets is defined as,

$$J(A_m, B_n) = \frac{|A_m B_n|}{|A_m B_n|}$$

where A_m is the m^{th} set from M items in Instacart, and B_n is the n^{th}set from N items in USDA. The cartesian product can be computed over all MN items to find the maximum Jaccard index for all products M products. Thus, representing the most similar joined products in the two datasets quantitatively. This was a computationally explosive task, but only had to be ran once to define foreign key to later join products.

Having the ability to join the datasets presented opportunities that allowed analysis of Instacart users to be tied into food specific contributions; such as, nutritional value, or ingredient consumption. This idea was explored to see how much sugar was being consumed to generalize the “healthiness” of Instacart users. This was accomplished by finding the average sugar purchased over all food products for each user. Each user then has an associated value of the average sugar they purchased. This could then be compared to the global average sugar over all food products in the Instacart dataset to analyze the habits of customers (figure 1).

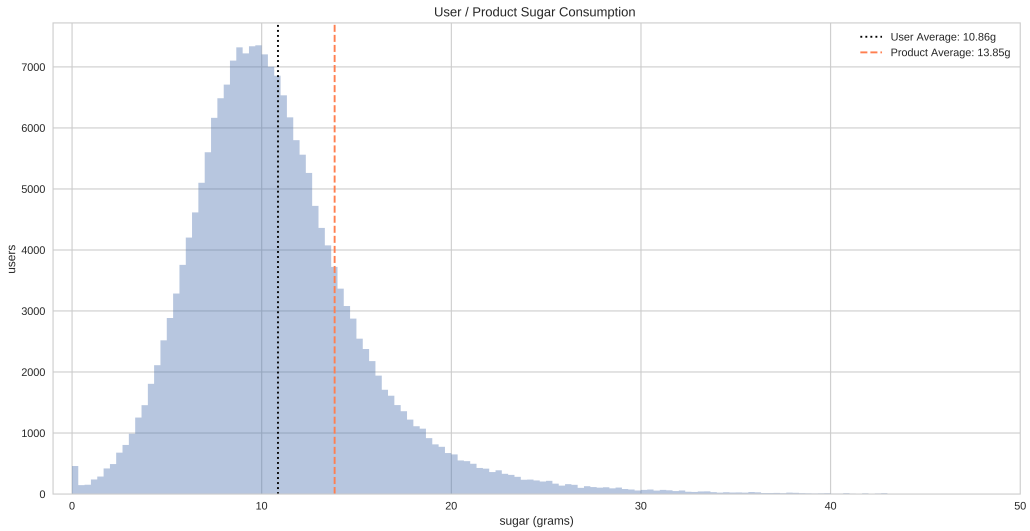

Figure 1 - Histogram of the average sugar purchased for all users in Instacart, measured per 100 grams of food, compared to the average sugar consumption of all products.

Looking further at the products purchased by customers, it became apparent that various items were often grouped to a unique set of users. This meant there was overlap in user purchases, but that there were also many items that remained unseen by many users. Therefore, we could introduce recommendations to customers following a collaborative filtering approach. Collaborative filtering is built to deal with large information sets, which is perfect for our use case. Collaborative filtering is based on the assumption that if user *A* and user *B* both like item *n*, then user *A* and user *B* are more likely to share ratings/opinions on other items. Essentially, we should recommend to user *B*, items that are liked by the most similar users to *B*.

In order to begin with this approach we first had to extract user ratings from our data. Since the dataset does not actually contain any user ratings we felt the best way to approximate a user rating was based off their purchase history. A product frequently purchased by a user, will have a higher rating than a product purchased infrequently, and those that were not purchased had no associated ratings. It could be argued that this might lessen the ratings of good products that do not need to be purchased frequently - such as tissue paper. However, this is a desired side effect since products that are not purchased frequently are not products that we want to recommend frequently.

In our case we will use the matrix factorization implementation of collaborative filtering available in the Apache Spark machine learning library. This is a technique that was first used [9] during the Netflix Prize competition which was held in 2006. The technique (described in detail here [10]) involves creating a matrix $\underset{u \times p}{\mathrm{R}}$ representing the original user product ratings from *u*

users with p products. The goal is to then discover k latent features by finding two matrices U and P, where the product approximates R with,

$$\underset{u \times p}{\tilde{\mathrm{R}}} \approx \underset{u \times k}{\mathrm{U}} \; \underset{k \times p}{\mathrm{P}^{\mathrm{T}}}$$

The matrix $\tilde{\mathrm{R}}$ reveals rating for each product in the set for each user. In order to calculate U and P we initialize these two matrices with k random values and calculate the distance of their product from the true values. This can be accomplished by using gradient descent to minimize the root-mean-square-error (RMSE) between the known values in R and the approximations in $\tilde{\mathrm{R}}$. This can be computed for all values by,

$$RMSE = \sqrt{\frac{1}{pu} \sum_{i=1}^{p} \sum_{j=1}^{u} \left(R_{i,j} - \tilde{R}_{i,j} \right)^2}$$

This approach is ideal for use on our dataset because of its efficiency with large dimensionalities. Another important factor for our choice of this method was its performance on similar datasets. For example the Netflix Prize dataset is quite similar to ours, in terms of size, and this algorithm proved to perform quite well with their data [9].

Even though this approach is ideal for large datasets we still ran into many issues with memory and time. Due to the size of our datasets and lack of many high end computer we frequently ran out of memory and had long running computations spanning over six hours. In order to solve this issue we were forced to scale back from 200,000 users and 50,000 products to the top 2,000 most active users and the products in their purchase sets. The measure of most active users were determined by the number of products purchased.

Food items in the Branded Food Products Database provided by USDA contain the nutritional information found on their packaging under the Nutrition Facts product label. To categorize and find foods which are similar in composition we used a dimensionality reduction algorithm to visualize the distribution of foods in two dimensional space. Seven components including protein, total fat, total carbohydrates, total sugar, sodium, water, and ash (inorganic material), were selected for their high degree of commonality in the dataset and the large percentage of total food composition they represent. Nutrient data was available as mass per serving size and each item's nutrient values were scaled by serving size to represent a nutritional ratio. The nutritional vector for each food had to be normalized to account for foods where the selected components did not account for 100% of the mass.

The method of dimensionality reduction selected was Principal Component Analysis; chosen for its proven effectiveness in linear dimension reduction and its inclusion in the Apache Spark library. The results of the algorithm ran on the products in the database were plotted in Python using matplotlib. To evaluate the effectiveness of the dimensionality reduction and the

meaningfulness of the resultant plot, it was decided to break the output space into clusters and determine if the clusters contained similar foods. Clustering was done using *k*-means and a value of 10 for *k*. The results were plotted using color to represent cluster value and the most frequent word for each cluster was calculated to provide context to groupings.

5. Experimental Benchmarks

The primary benchmark used to analyze the performance and accuracy of our recommendation system was to measure the RMSE of various trained models. By manipulating hyper-parameters; including, training iterations, number of latent features, the learning rate, etc, we could find those that best represent the data. To do this, many different models were trained and their corresponding RMSE values were captured and compared to find the model with minimal error that would produce the best recommendations. Our implementation allowed for these values to be predefined, and tested over all possible combinations. Thus, hundreds of models could be trained, and the one with minimal error can be saved.

We explored 52 different models which took a total of 86 minutes 28 seconds to analyze, thereby averaging 1.66 minutes to train each model. Plotting these errors for all different combinations of hyper-parameters gave insight to how different models behave (figure 2). Moreover, it can be seen that the model that minimized the error between actual and predicted rating matrices had a RMSE value of 8.6147 which, given items with ratings ranging from 1 to over 100, is not terrible. It is highly likely that some of the higher ratings had larger errors, and

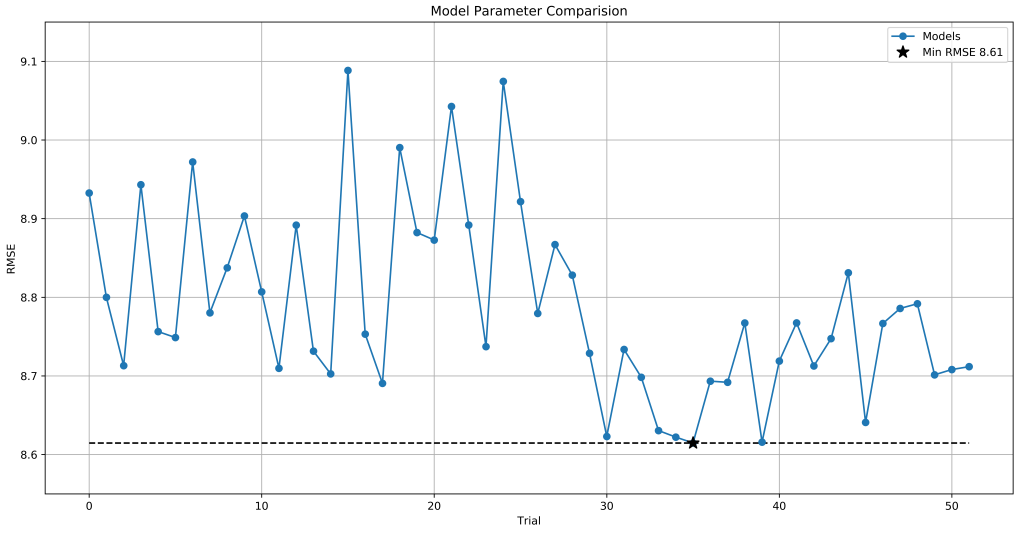

Figure 2 - Comparing 52 models with different hyper-parameters and comparing their RMSE values. The best model was seen with 2000 products, 6 training iterations, a regularization value of 0.50, rank (*k*) of 25, and alpha of 30.

that some of the lower ratings had smaller ones. Furthermore, we could likely have brought this RMSE down further given faster machines with more memory. Even though it only took roughly two minute to train a model, it was difficult to increase the complexity of the latent space due to memory limitations of our Spark cluster. As it stands we were unable to try iterations larger than six or rank (number of latent features) larger than 200 without receiving a memory overhead error.

Further analysis and benchmarking of our recommendations was completed via insights from external parties, i.e., introducing human judgement to measure the accuracy of our model. This was accomplished by using multiple test subjects to look over and correlate with the shopping habits of some user in Instacart. Thereafter, this subject was presented the corresponding recommendations and asked how satisfied they would be with said results. After running the test on five different subjects, it was noted the subjects had an average of 85.15% satisfaction - determined by how many of the recommendations they would happily purchase.

Benchmarking the effectiveness of our joining technique using the was done iteratively to find a threshold that would make the most logical sense. We ensured the accurateness of our algorithm by computing the Jaccard index for a few samples by hand and comparing them to our results to ensure equality. Second we analyzed the effectiveness of the approach by examining the Jaccard index for a variety of pairs to decide whether the results were ideal. The vast majority of the time the score gave a good approximation on the similarity of the two product names. Analyzing the results also allowed us to vary the threshold until a value of 0.5 was found to best represent the two datasets.

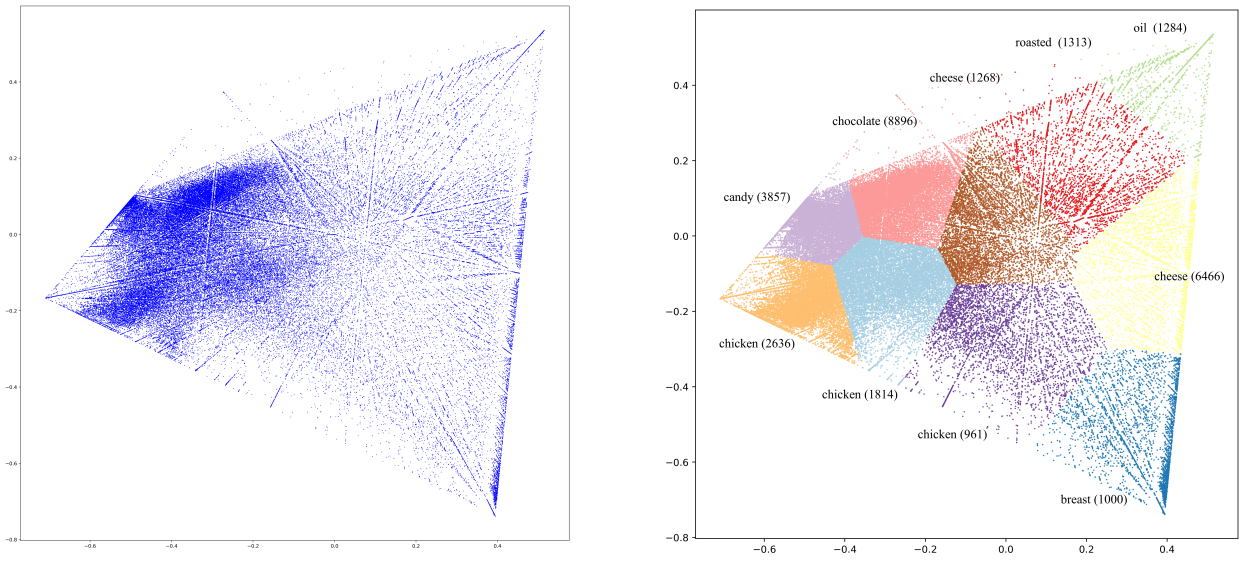

Figure 3 & 4 - Visualization of seven different nutrient values reduced to two-dimensions and clustered using *k*-means.

Benchmarking of our nutrient analysis was completed via human judgement. We plotted our results using matplotlib in order to visualize the problem (figure 3 & 4). In figure 3 above we display our initial results which seem to show clustering in a portion of the graph, indicating foods with similar nutritional values lie in those areas. We clustered the data shown in figure 3 as defined in the methodology to attempt to find groupings (figure 4). From this we were able to draw findings that add effective insights to why certain products correlate well with one another, strengthening previous results.

6. Insights

Early into our analyzation of nutrition attributes we began to understand how important good data is. Although our dataset was well formatted with minimal missing values, both valuable attributes in a dataset, we soon learned that it had some less than ideal characteristics. Specifically relating to the ingredients list for each product present in the USDA dataset. Marketing jargon had an influence on the naming convention for many entries in the dataset, resulting in poor naming conventions for names for data processing. These lists often had ingredients that, for all intents and purposes, were the same product but were identified with a different name. For example, various companies that sold water and other foods that contained water, listed the ingredient as “water”, “organic water”, “reverse osmosis water”, etc. This observation remained consistent for numerous products and their associated ingredients, making our analysis on the matter challenging and helped spur our switch to analyze the nutrient characteristics of food rather than ingredients.

We encountered various impediments while implementing our recommendation algorithm that inadvertently resulted in more satisfactory results. One of the major challenges discovered in the initial model was found trying to recommend all 50,000 products to all 200,000 users. Experimental results proved for this task to be computationally challenging. Even with Apache Spark’s optimized framework, having a recommendation matrix of 10 billion entries, lead to slow training and extremely high memory utilization. The memory use was remarkably exhaustive and caused serious issues until the number of products used for recommendations were reduced. This decision was twofold; foremost addressing the issue explained above, but to also provide more prevalent results by only considering the most frequently purchased items. Subsequent findings were more relatable and found much more efficiently.

Another factor we did not consider much in initial discussing of the project was how we would actually test and verify our results. This is an area that should likely have been at the forefront of our discussion and will be in the future. Many of our verification techniques involved us, and others, to look at our results and reason for accuracy and satisfaction. This was necessary because recommendations could algorithmically be determined, but hold a sense of undisclosed bias. Although this is always good practice, it seems that it would have been more useful and reliable to have a more deterministic way to verify our results. For example it would be nice to

be able to verify with more confidence that our user recommendations or nutrient decomposition graph are accurate. However, cross validating quantitative accuracies, such as RMSE, and human judgement made for satisfactory results.

7. Future Work

Future research in this problem space will revolve around better fitting the needs of evolving demographics. It is evident from the success of Instacart that the convenience of ordering groceries without leaving the confines of everyday busy schedules is growing in popularity. If people could continue to spend less time commuting to the store, or trying to decide what to eat, more time could be dedicated to tasks that are enjoyed. We do not believe the social and cultural practices of dining should change, but rather we can and should improve the way in which food is purchased.

With the advancements in data collection and machine learning, recommendation systems have the ability to be further explored to simplify the shopping process. Integrating future development of smart home, and smart devices could introduce autonomous agents to monitor the quantity of food items within a home. Over time these devices would learn the behaviors of the owners, and purchase food automatically. Feedback would be a valuable asset for this system allowing users to make requests such as eating healthier or planning for dinner parties. These changes would require better and more complex machine learning algorithms to find specific ingredients that would promote a healthy and enjoyable meal. These agents will be able to learn more about individual behaviors to simplify the way in which food is purchased.

Other methods such as analyzing the makeup of foods could be incorporated as well. These techniques could be used to help determine the nutritional values of food and rate them on a healthiness scale. Further nutrient decomposition could possibly even aid recommendations through some method such as *k*-means Clustering to find items with similar nutritional attributes or ingredients. Data from this research could further be used to create dietary plans for individuals in order to improve the healthiness of their diet. These plans would be drawn from their previous dietary history so that the plans could make improvements to all factors of an individual's health.

These sorts of methods, whereby a user or personal autonomous assistants, make decisions for the user’s diet based off of a set of algorithms, would be able to significantly reduce major health issues facing us today. One of the primary factors contributing to food related health issues is compulsiveness, a factor which could be dramatically reduced by not giving the option of unhealthy foods. While this would only work for users who are interested in trying to improve their health, that is all we can ever hope for.

8. Conclusion

Approximately 200,000 individuals were sampled from North America as recorded users of the Instacart grocery delivery platform. Tracking these users allowed for deeper analysis of their behaviors and food spending habits. Initial exploration looked at how nutritional and dietary structures correlate with frequently purchased products. According to the National Cancer Institution, adult men and women consume an average of 84.2 grams of sugar daily [11]. This is more than two times the recommended amount. Quantitatively measuring the consumption of sugar for individuals directly was not possible, but the patterns of purchased products gave insight to one's lifestyle.

It was seen that the majority of Instacart user's purchased items that, on average, contained 2 grams less sugar than the 13.85 gram average of available food products in Instacart (figure 1). Therefore indicating that these users are decisively purchasing items that support a healthier diet. This is a marketable observation as retailers could leverage the items with less sugar for profit; that is, promote organic options that have reduced sugars for greater margins. In fact, if these items are being bought more regularly by the majority of users, then a collaborative recommendation system could be implemented to targets items to similar users. Subsequently, generating more revenue.

Matrix factorization is an effective way to accurately recommend these products to users given a large and somewhat sparse user product rating set. Our algorithm was able to create satisfactory recommendations for each user using the approach outlined in the methodology section. Our algorithm had an acceptable RMSE, although given more high end computers it would have been possible to result in potentially better recommendations at an even lower RMSE. That being said, it is unlikely that matrix factorization on its own would be able to create much better recommendations and would likely need to be combined with other machine learning approaches and possibly more data in order to obtain the best possible results.

More data would be similarly useful in helping to improve our food grouping algorithm As it turns out, and as can be seen in the figure 4, nutritional values do not hold enough information to correctly characterize foods nor their groups. Although nutritional values do seem to create some groupings, the groupings created are not descriptive nor unique enough to be able to define any food groups. It is likely that further information about foods would be needed in order to correctly classify foods into groups, the most likely candidate being their ingredients.

Bibliography

[1] Neck, H. M., Neck, C. P., & Murray, E. L. (2018). Entrepreneurship: The practice and mindset. Los Angeles: SAGE.

[2] Gezer, C. (2018). Impact of Dietary Pattern on Human Life Quality and Life Expectancy: A Mini-Review. Research & Investigations in Sports Medicine,2(5).

[3] 3 Million Instacart Orders, Open Sourced. (n.d.). Retrieved from https://www.instacart.com/datasets/grocery-shopping-2017

[4] USDA Food Composition Databases. (n.d.). Retrieved from http://www.ars.usda.gov/nutrientdata

[5] The Instacart Online Grocery Shopping Dataset 2017 Data Descriptions. (n.d.). Retrieved from https://gist.github.com/jeremystan/c3b39d947d9b88b3ccff3147dbcf6c6b

[6] Isinkaye, F., Folajimi, Y., & Ojokoh, B. (2015). Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal,16(3), 261-273.

[7] Flores-Lopez, A., Perry, S., & Bhargava, P. (2017). What's for Dinner?: Online Grocery Recommendations[Scholarly project]. In CS229: Machine Learning. Retrieved from http://cs229.stanford.edu/proj2017/final-posters/5136630.pdf

[8] Kosub, S. (2019). A note on the triangle inequality for the Jaccard distance. Pattern Recognition Letters,120, 36-38.

[9] Funk, S. (2006, December 11). Netflix Update: Try This at Home. Retrieved from https://sifter.org/~simon/journal/20061211.html

[10] Yeung, A. A. (2010, September 16). Matrix Factorization: A Simple Tutorial and Implementation in Python. Retrieved from http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/

[11] Usual Daily Intake of Added sugars. (2010). Retrieved from https://epi.grants.cancer.gov/diet/usualintakes/pop/2007-10/table_a40.html