# Applications of Unsupervised Techniques on Instacart

**Devanshi Deswal and Karthik Chevuru**
Northeastern University

## Abstract

The project is aimed at uncovering information of Instacart's customer base, based on their interactions with the retail business, in terms of purchase conduct and patterns. Additionally, the project focuses on analyzing the purchasing behavior of customers to demonstrate better knowledge of their needs and wants.

Cluster analysis on Instacart's customer data was performed using two different algorithms. These were K-means clustering and Gaussian Mixture Model (GMM). The input data for the clustering algorithms was sparse and high dimensional. It contained statistics about customer's past purchases. For each clustering algorithm, a parameter search was carried out. The best clustering was measured based on the highest silhouette score compared among each of the clustering results.

Market Basket Analysis was performed using the Apriori algorithm for identifying possible rules. The rules were pruned by calculating the following three metrics – support, confidence and lift. To further explore and deploy cross selling a recommendation system was designed. The main approach used for the recommender systems was collaborative filtering, where we tried to categorize similar products together. The information derived from each grouping was used to make recommendations to the customers. The input to the collaborative filtering algorithm was the customer purchases upon which cosine similarity and matrix factorization methods were applied to predict recommended items.

The results for the clustering algorithm indicate that the K-means performed better than GMM. However, it can be stated that the algorithm showed the best general results on t-SNE with three distinguishable customer groups. And, for the collaborative filtering-based recommendation system, it was found that it performs better than the baseline model in terms of root mean squared error and time computation.

## Introduction

Instacart is an American company with 1-day delivery services for grocery and fresh food items. Customers select products through an online portal among the listed items which are then delivered to their doorstep. Instacart has a large assortment of products online with almost 50 thousand distinct products. Hence, it implies choosing between many products and can be mentally exhausting. With an analysis of data on past customers' purchases, a recommender system was designed to help customers make selections for the items following their choice.

The project majorly focuses on building marketing strategies for similar customer groups by analyzing their spending behavior based on their interactions with the business. Additionally, for increasing revenue of the firm the customers' needs are evaluated to promote the sale of more products by identifying interesting relations among the items.

To better understand the customer's behavior cluster analysis was performed on past purchases of the customers to find interesting patterns in the data. The project hence aims at answering the question below:

- Which of the following clustering algorithms among K-means (center-based clustering) and Gaussian Mixture Model(GMM) perform better on a sparse data set based on average silhouette value?

Also, of interest was to examine how the results are impacted through different preprocessing techniques in sequence followed with a different choice of cluster grouping using the elbow method. Analyzing the resulting clusters for interesting patterns to identify similar customer behavior, when grouped based on the similarity of previous purchases.

To understand the relationships between the products that people buy market basket analysis was carried out using the Apriori algorithm and picking out rules that are worth pursuing. By and large, the latter was done by implementing thresholds for support, confidence, and lift. The output summarized how compelling the relationship between the products is. The results are extracted from the transaction data to structure an understanding of the customer's choice. The information is then evaluated for cross-selling marketing activities.

As a more concrete approach to revenue generation, a recommender system was built to offer a list of suggestions for products to the user. The data was decomposed using Singular Value Decomposition (SVD) and using cosine similarity as a measure, the recommendations generated were returned to the user. The core of the recommendation algorithm is based on a collaborative filtering that examines each item on the target user's list and finds other items in the choice set that seems similar to the item. We refined and tuned the parameters for the algorithm by comparing our predicted items against the base-line prediction.

## Background

Clustering algorithms generates groups such that within clusters entities have some similar characteristics among them. Similarity is defined in terms of distance between objects in space.

### K-means

K-means algorithm is a popular centroid based algorithm where each cluster's center is represented by the mean value of the objects in the cluster. For a good K-means clustering the variance within each cluster should be as small as possible and the variation between clusters should be clear.

Input: k - the number of clusters, D - a data set containing n objects.

Output: A set of k clusters.

Method: (1) arbitrarily choose k objects from D as the initial cluster centers; (2) repeat (3) (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster; (4) update the cluster means, that is, calculate the mean value of the objects for each cluster; (5) until no change.

### Gaussian Mixture Model(GMM)

Gaussian Mixture Models (GMM) assumes that there are a certain number of Gaussian distributions and each of these distributions represents a cluster. Hence, a GMM tends to group the data points belonging to a single distribution together.

Method: (1)expectation step or E step, consists of calculating the expectation of the component assignments for each data point given the model parameters; (2) The second step is known as the maximization step or M step, which consists of maximizing the expectations calculated in the E step with respect to the model parameters. This step consists of updating the values; (3) The entire iterative process repeats until the algorithm converges, giving maximum liklihood estimate.

### Cosine Similarity

Cosine similarity is a metric used to measure how similar the two items or products are. It measures the cosine of an angle between two items projected in multi-dimensional space.

Cosine similarity allows us to measure the similarity of a document of any type. Due to a multi-dimensional array, any number of variables (which are treated as dimensions) can be used, which in turn supports large sized documents.

Mathematically, the cosine of the angle of between two vectors is derived from the dot product of the two vectors divided by the product of the two vectors' magnitude.

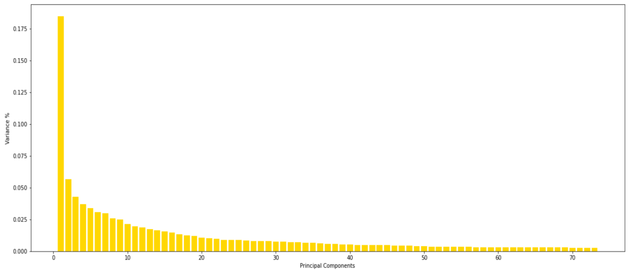
$$\text{Similarity}(p,q) = \cos\theta = \frac{p \cdot q}{\|p\|\|q\|} = \frac{\sum_{i=1}^{n} p_i q_i}{\sqrt{\sum_{i=1}^{n} p_i^{2}}\sqrt{\sum_{i=1}^{n} q_i^{2}}}$$

The output of cosine similarity ranges from 1 to -1. A value of -1 signifies that the items are dissimilar while 1 indicates complete similarity of the products.

## Parameter Search and Clustering

Purchases by customers were used to build a sparse utility matrix, with user-id as rows, products as columns, and entries as purchase frequencies. Since the customer segmentation data was high dimensional Principal Component Analysis(PCA) was applied. PCA was used to preserve the underlying structure of the data while reducing the dimensionality. Since the proportion of variance explained by individual features was only marginal for the last n ordered components, hence for each percentage of increase in variance the variables were significantly increasing. As a result, a choice of preservation of 90% of the variance in the data was made which significantly reduced the number of dimensions from 403 to 73. This led to a reduction of data by approximately 80%. This contrasts with 44% components required for explaining 95% of the variance in the data

Individual components explaining 90% variance



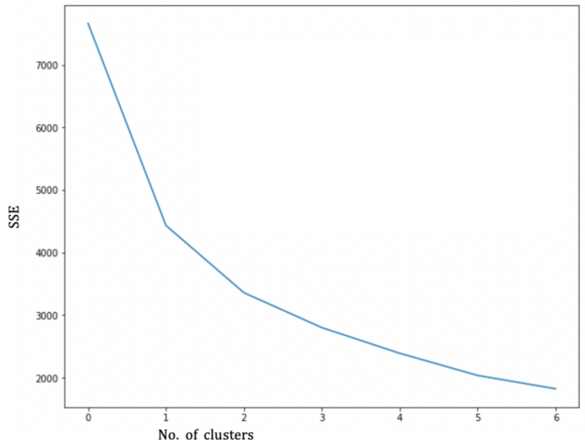
The transformed data formed a new coordinate system and the values were recomputed using the top n principal components.

## k-Means

k-Means, a popular algorithm for customer segmentation was selected for clustering. Since the algorithm depends on a user-specified value of k clusters or subgroups, we determined an optimal value for k. Parameter search was performed as follows:

1. 1. K-Means algorithm was run for k = (1, 2, 3,…..7). Other relevant arguments such as maximum iterations were set to 100. The sum of square error was computed for each k and an elbow plot was plotted to compare among the clusters.
2. Similarly, the average Silhouette value was computed for each cluster to validate the selection of best k.
3. The best k was selected to be 3, on the basis of the elbow in the graph and the highest average silhouette value.

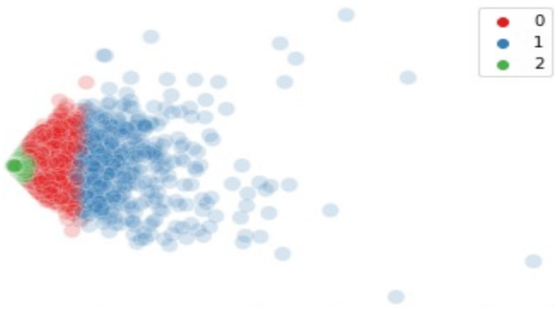
Elbow method to find optimal k



The k-Means algorithm was run with three centroids initialized to random data points from the dataset. With each iteration performed the algorithm minimized the intercluster squared error until convergence.

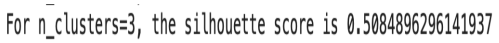
We then plot the k-Means derived cluster for visualizations. However, the clusters are not clearly separable in the 2D space.

K-Means on reduced data



The data was again reduced using t-SNE as an alternative and the results were compared to PCA k-Means. The reduction was performed down to three t-SNE components using the elbow plot.'n_iter' parameter was used to reduce the computation time required for t-SNE. Post reducing the data into three t-SNE components, k-Means algorithm was performed once again.

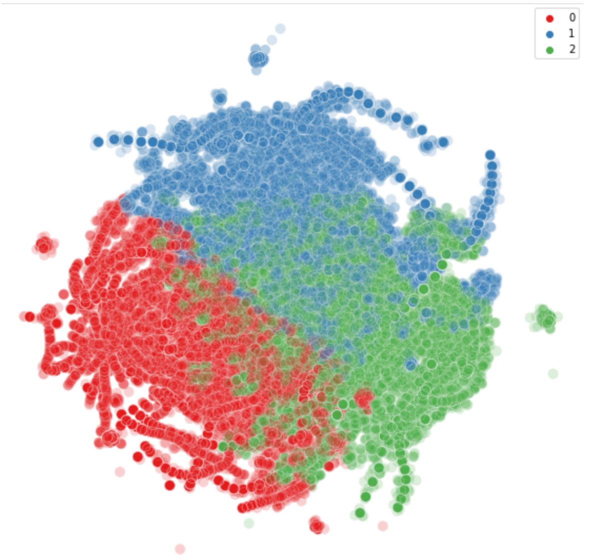
With K-means applied on the three derived components of t-SNE we observed a marginal improvement in the Silhouette score as compared to the k-Means performed on PCA. The difference in the silhouette score was approximately .03 with t-SNE having a silhouette score of .508 against .481 in the case of k-Means on PCA.



```
For n_clusters=3, the silhouette score is 0.5084896296141937
```

Clusters derived from t-SNE were displayed as broad multiple clusters. However, the relative advantage of t-SNE relates to the location and distance in space of each k-Means cluster. Moreover, using t-SNE, an explicit degree of separation was observed between the k-Means clusters.

k-Means on t-SNE

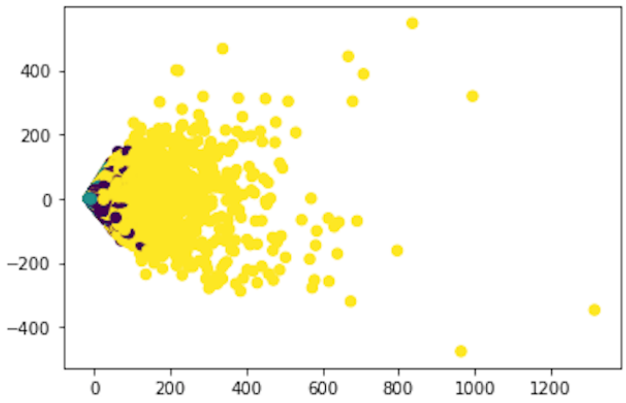


## Gaussian Mixture Model (GMM)

A limitation to k-Means is that it assumes all data points are only associated with one cluster, hence there is no probabilistic measure of uncertainty describing the measure of association of the points with the cluster. As an alternative to this hard clustering approach, a soft clustering method was applied using the Gaussian Mixture Model to group customers.

To run the GMM algorithm, we first initialed the parameters for the expectation algorithm. Similar values to k-Means algorithm were used for efficient computation and comparison. The algorithm which follows an iterative process executes the expectation step followed by the maximization steps. The likelihood continuously made progress and finally converged after 18 epochs.

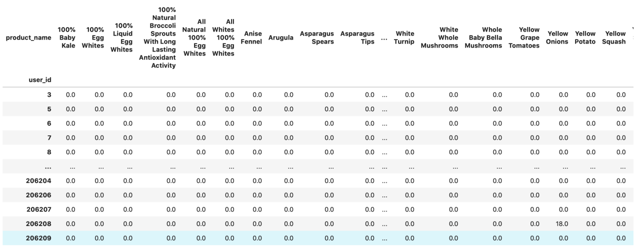
Clustering using GMM



## Recommender System

We have an nXm matrix with details about purchases of n users and m products. Each cell value in the matrix signifies a purchase. We want to recommend products to the user based on their behavior of past purchases. Item based collaborative filtering method was selected to recommend new products to the users. For every similar item i, we identified their previous purchases and multiplied cosine similarity of item p with item i. We trained the model on the training data randomly subsetted from the original data using 4:1 train: test ratio. Then the weighted averages were summed. Finally, the items were sorted by their average weights. The average weight served as an estimate for recommending products to the user. The products with the highest similarity was recommended to the user. Since the model gave a large number of predictions, we used a helper function to get only the top 3 recommended items. Since the products and the users were represented by id numbers, we used a second helper function for easy interpretable results.

Results on Apriori Algorithm



| product_name | 100% Baby Kale | 100% Egg Whites | 100% Liquid Egg Whites | 100% Natural Broccoli Sprouts With Long Lasting Antioxidant Activity | All Natural 100% Egg Whites | All Whites 100% Egg Whites | Anise Fennel | Arugula | Asparagus Spears | Asparagus Tips | ... | White Turnip | White Whole Mushrooms | Whole Baby Bella Mushrooms | Yellow Grape Tomatoes | Yellow Onions | Yellow Potato | Yellow Squash |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| user_id | | | | | | | | | | | | | | | | | | |
| 3 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 5 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 6 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 7 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 8 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 206204 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 206206 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 206207 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 206208 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 18.0 | 0.0 | 0.0 |
| 206209 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ... | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

However, for high dimensional data, collaborative recommender systems do not perform as well. Since our utility matrix is more than 99.8% sparse, we explored matrix factorization using SVD to uncover latent features that might help generate better recommendations.

We used SVD to generate a low rank approximation of the utility matrix with 50 and 100 latent factors. Below, A is the preference Matrix for (product,user) U (product factors) and V (user factors) are the latent singular vectors and D is the latent singular values.

$$A = UDV^T$$

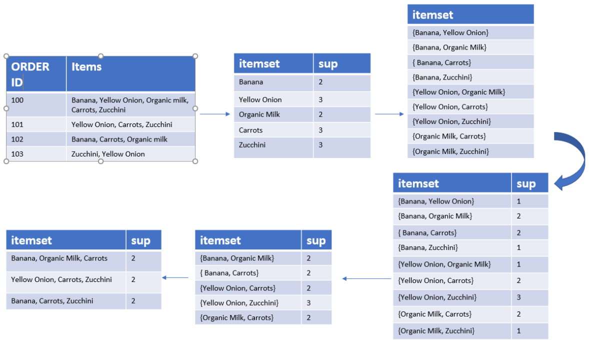
Product preferences for a user i can be found out by below formula. Using $A^T$ [i] we recommend K most preferred products which were not previously bought by that user.

$$A^T [i] = UDV [i]$$

## Market Basket Analysis

We have an nXm transaction matrix which contains information about items being brought together. However, the information of interest for us lies in the fact on how often they are brought together.

Association rule mining is a machine learning technique which was used to uncover association among products. Apriori algorithm was the method used for the Instacart to generate candidate frequent item sets against the FP-growth algorithm which develops a tree by 'divide and conquer' strategy. For our dataset we first created a user-item matrix and with minimum support of 2 we iterated over and over until we reach a point where, we get no new candidate sets. Finally, we calculate the confidence and lift lift (confidence / unconditional probability)to perform pruning over the generated rules to subset out interesting associations**.**



| ORDER ID | Items |
|---|---|
| 100 | Banana, Yellow Onion, Organic milk, Carrots, Zucchini |
| 101 | Yellow Onion, Carrots, Zucchini |
| 102 | Banana, Carrots, Organic milk |
| 103 | Zucchini, Yellow Onion |

| itemset | sup |
|---|---|
| Banana | 2 |
| Yellow Onion | 3 |
| Organic Milk | 2 |
| Carrots | 3 |
| Zucchini | 3 |

| itemset |
|---|
| {Banana, Yellow Onion} |
| {Banana, Organic Milk} |
| { Banana, Carrots} |
| {Banana, Zucchini} |
| {Yellow Onion, Organic Milk} |
| {Yellow Onion, Carrots} |
| {Yellow Onion, Zucchini} |
| {Organic Milk, Carrots} |
| {Organic Milk, Zucchini} |

| itemset | sup |
|---|---|
| {Banana, Yellow Onion} | 1 |
| {Banana, Organic Milk} | 2 |
| { Banana, Carrots} | 2 |
| {Banana, Zucchini} | 1 |
| {Yellow Onion, Organic Milk} | 1 |
| {Yellow Onion, Carrots} | 2 |
| {Yellow Onion, Zucchini} | 3 |
| {Organic Milk, Carrots} | 2 |
| {Organic Milk, Zucchini} | 1 |

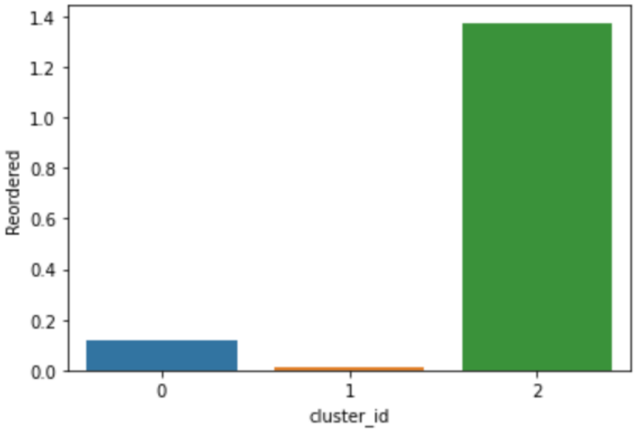
| itemset | sup |
|---|---|
| {Banana, Organic Milk} | 2 |
| { Banana, Carrots} | 2 |
| {Yellow Onion, Carrots} | 2 |
| {Yellow Onion, Zucchini} | 3 |
| {Organic Milk, Carrots} | 2 |

| itemset | sup |
|---|---|
| Banana, Organic Milk, Carrots | 2 |
| Yellow Onion, Carrots, Zucchini | 2 |
| Banana, Carrots, Zucchini | 2 |

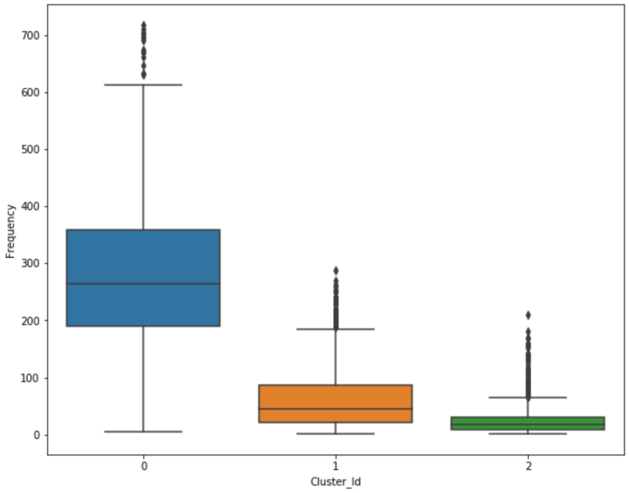
## Results

To summarize the results of the clustering algorithm, which was aimed at grouping similar customers together, we plotted graphs summarizing characteristics of each of the three customer groups.

Reorders comparison for customer clusters



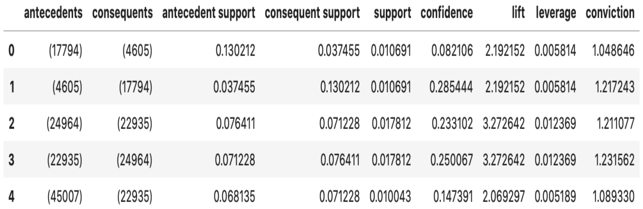
Frequency vs Customer cluster



We observe that users with cluster id 2 are the ones with least frequent purchases and comparatively very high number of reorders hence requiring attention by the business. To encourage regular purchases from such customers the business can offer membership deals with the benefit of earning credits on every dollar spent. Additionally, they have highest reorders implying confusion in product selection. Hence the recommendation system can be helpful for such group more specifically.

Users with cluster id 0 are frequent visitors with very low reorder proportions hence most profitable for the business to retain. While users with cluster id 1 are the ones with less frequent purchases and also minimal reorders, hence not much important to the business strategy team. Hence by studying the distinguishable characteristics among each customer group and right marketing strategy for the right group of customers we would be able to invest marketing budget more optimally. As a result, it would lead to large revenue generation.

For business to understand the relationship between the products using customers purpose pattern Apriori approach was used. By setting a support of 0.01, a confidence of 0.5, and a lift threshold of 2 to the Apriori algorithm we obtained 9000 results. The results obtained would increase if we increase either of the threshold value. The threshold values were chosen such that the rules returned were interesting and not just the commonly evident ones.

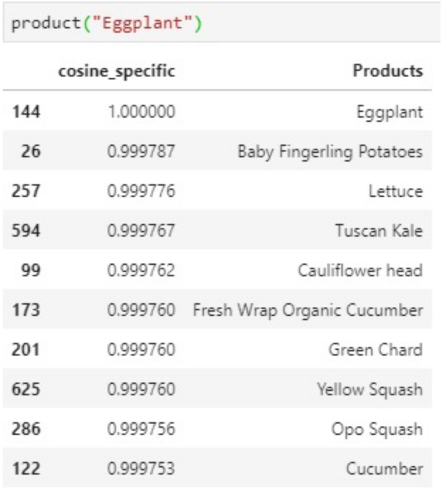


| | antecedents | consequents | antecedent support | consequent support | support | confidence | lift | leverage | conviction |
|---|---|---|---|---|---|---|---|---|---|
| 0 | (17794) | (4605) | 0.130212 | 0.037455 | 0.010691 | 0.082106 | 2.192152 | 0.005814 | 1.048646 |
| 1 | (4605) | (17794) | 0.037455 | 0.130212 | 0.010691 | 0.285444 | 2.192152 | 0.005814 | 1.217243 |
| 2 | (24964) | (22935) | 0.076411 | 0.071228 | 0.017812 | 0.233102 | 3.272642 | 0.012369 | 1.211077 |
| 3 | (22935) | (24964) | 0.071228 | 0.076411 | 0.017812 | 0.250067 | 3.272642 | 0.012369 | 1.231562 |
| 4 | (45007) | (22935) | 0.068135 | 0.071228 | 0.010043 | 0.147391 | 2.069297 | 0.005189 | 1.089330 |

For instance, from association rule mining we can see organic yellow onion (24964) and organic garlic (22935) have a lift value of 3.27. We can inference that customers who purchased organic yellow onions are more likely to buy organic garlic. Hence by placing onions and garlic in a same aisle, we can see higher sales between them, ultimately leading to increase in revenue generation.

To build on the idea of recommended purchases to users, a collaborative based recommender system was built. The results obtained using cosine similarity as a measure with and without using SVD are as follow:

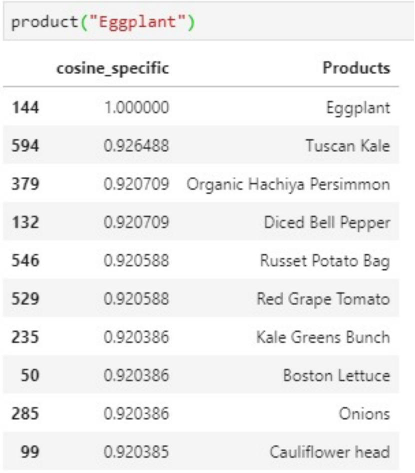
Recommendations obtained using SVD



```
product("Eggplant")
```

| | cosine_specific | Products |
|---|---|---|
| 144 | 1.000000 | Eggplant |
| 26 | 0.999787 | Baby Fingerling Potatoes |
| 257 | 0.999776 | Lettuce |
| 594 | 0.999767 | Tuscan Kale |
| 99 | 0.999762 | Cauliflower head |
| 173 | 0.999760 | Fresh Wrap Organic Cucumber |
| 201 | 0.999760 | Green Chard |
| 625 | 0.999760 | Yellow Squash |
| 286 | 0.999756 | Opo Squash |
| 122 | 0.999753 | Cucumber |

Recommendation obtained using cosine similarity measure without using SVD



```
product("Eggplant")
```

| | cosine_specific | Products |
|---|---|---|
| 144 | 1.000000 | Eggplant |
| 594 | 0.926488 | Tuscan Kale |
| 379 | 0.920709 | Organic Hachiya Persimmon |
| 132 | 0.920709 | Diced Bell Pepper |
| 546 | 0.920588 | Russet Potato Bag |
| 529 | 0.920588 | Red Grape Tomato |
| 235 | 0.920386 | Kale Greens Bunch |
| 50 | 0.920386 | Boston Lettuce |
| 285 | 0.920386 | Onions |
| 99 | 0.920385 | Cauliflower head |

From the results we observe that there is a significant difference in the recommended products using the two methods. Hence, evaluating the root mean squared error of the two model we observed a noteworthy reduction of .18 in the error by using SVD to decompose matrix before calculating the cosine similarity.

Hence, by deploying the collaborative based filtering using SVD as the final model for product recommendation we were able to observe a RMSE of .37. The business revenue tends to increase as a consequent of increase in sales with customers being recommended potential addition to their basket.

## Conclusion

To summarize, the results depicts that a sparse dataset matrix could well cluster with K- means algorithm on scaled data. Silhouette index evaluated the best clustering measure. The results also support that it is a good practice to preprocess data when performing cluster analysis. With market basket analysis useful relations between products were derived based on customer purchases and product associations. The algorithm is easy to run and analyze. More and more business organizations are adopting ways to use these analyses to gain insights into hidden relationships. Above all, collaborative filtering provided a powerful methodology to recommend new product or items to user. Cosine similarity was used to get better recommendation results than reducing. MAE Data analysis in combination with marketing and business can result in growth of customer base for Instacart business while simultaneously increasing their revenue.

For future work, other clustering techniques could be tested and compared to the K-means implementation on the basis of clusters generated. Since the customer data is continuously increasing more efficient clustering needs to be employed to deal with exponential data. Also, for the recommendation system with more well detailed metadata about the products we could use a hybrid approach combining content based and collaborative based system for more accurate product recommendations to customers. Feedback from customers can be used to evaluate the accuracy of the system.

## References

"The Instacart Online Grocery Shopping Dataset 2017", Accessed from https://www.instacart.com/datasets/grocery-shopping-2017

"Comparative Study on Apriori Algorithm and Fp Growth Algorithm with Pros and Cons", http://www.ijcstjournal.org/volume-4/issue-4/IJCST-V4I4P28.pdf

"Singular Value Decomposition (SVD) & Its Application In Recommender System", https://analyticsindiamag.com/singular-value-decomposition-svd-application-recommender-system/