# Instacart Market Basket Analysis

The Erdos Institute, Fall 2022 Bootcamp

**Sycamore team**: Thanos Kritikos, Haoran Li

# Overview

## Problem

- Use customer Instacart orders over time to predict which previously purchased products will be in a user's next order
- Understand the consumer trends and customer behavior

## Stakeholders

- Instacart – Grocery delivery app
- Online grocery shopping companies
- Customers

## Task

- Analyze Instacart data set
- Categorize customers behavior by their purchases
- Predict the future purchases with ML algorithms

# Table of contents

01

# Data Creation

Acquisition & preprocessing of data

# The Data

- The data were acquired through a competition on Kaggle that was completed in 2017: https://www.kaggle.com/competitions/instacart-market-basket-analysis/overview
- It is an anonymized dataset containing a sample of over 3 million grocery orders from more than 200,000 Instacart users.
- The week and hour of day the order was placed, a relative measure of time between orders, the priors orders and a training data set are provided.

# Data cleaning

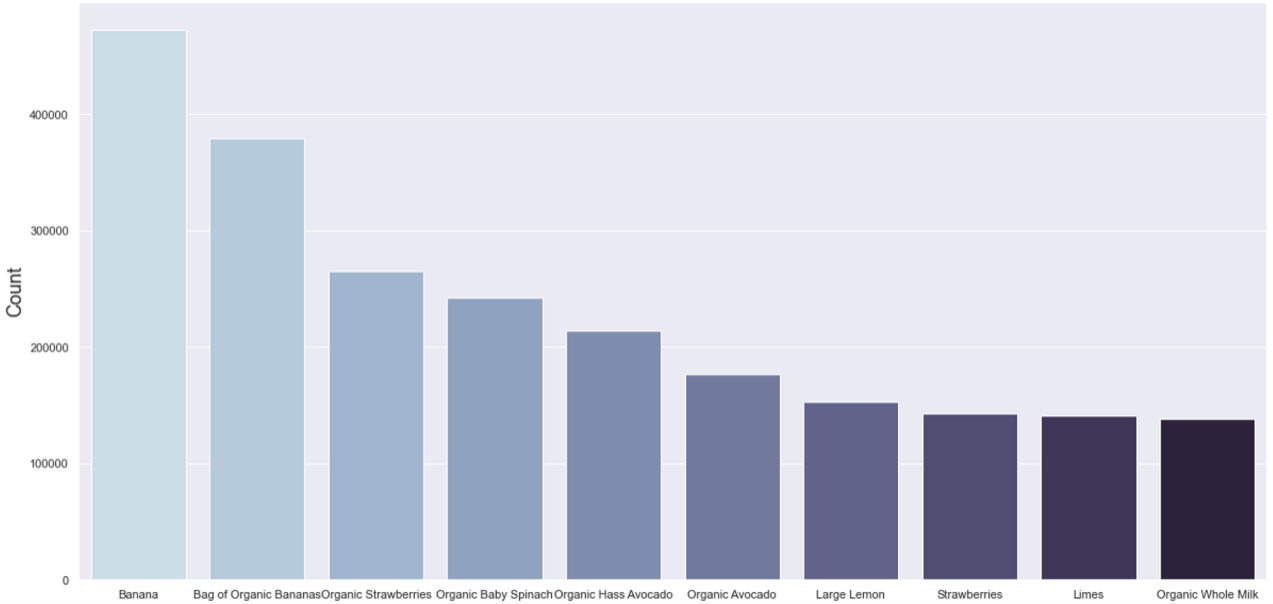
- To conduct an exploratory analysis the individual files were merged into one common file.
- The non-existent values for previous orders conducted, were replaced with 0.
- The data were organized into:
  - Orders
  - Products
  - Aisles
  - Departments

02

# Exploratory Analysis

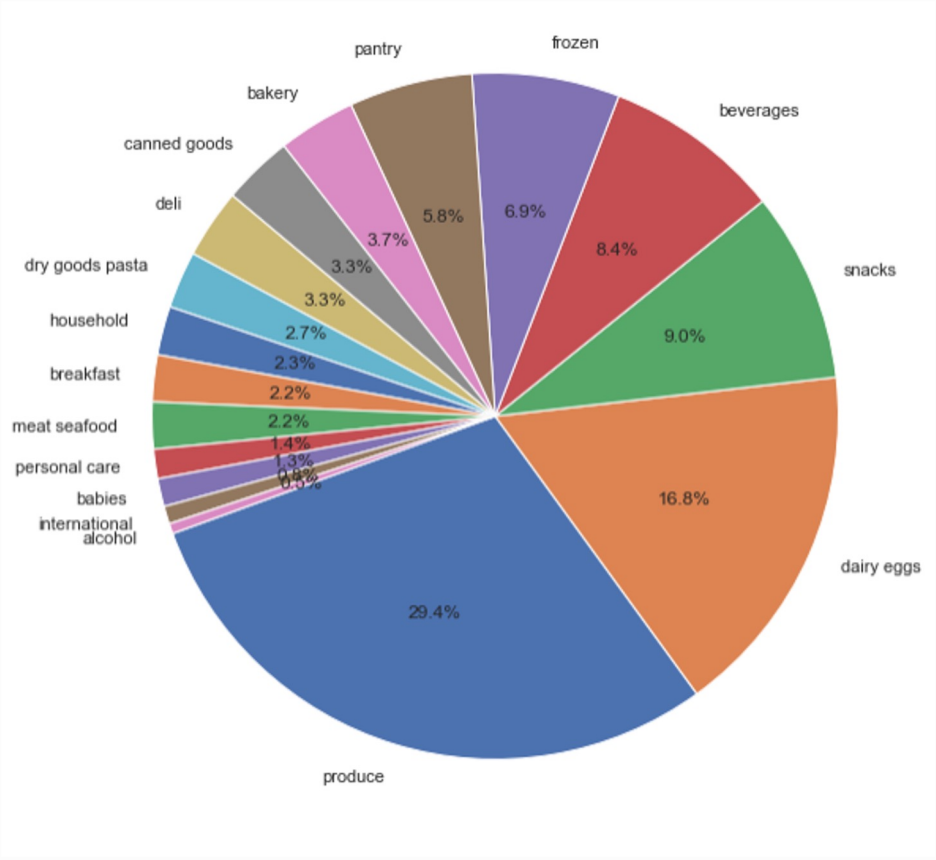
Analysis of consumer trends

# Top 10 products



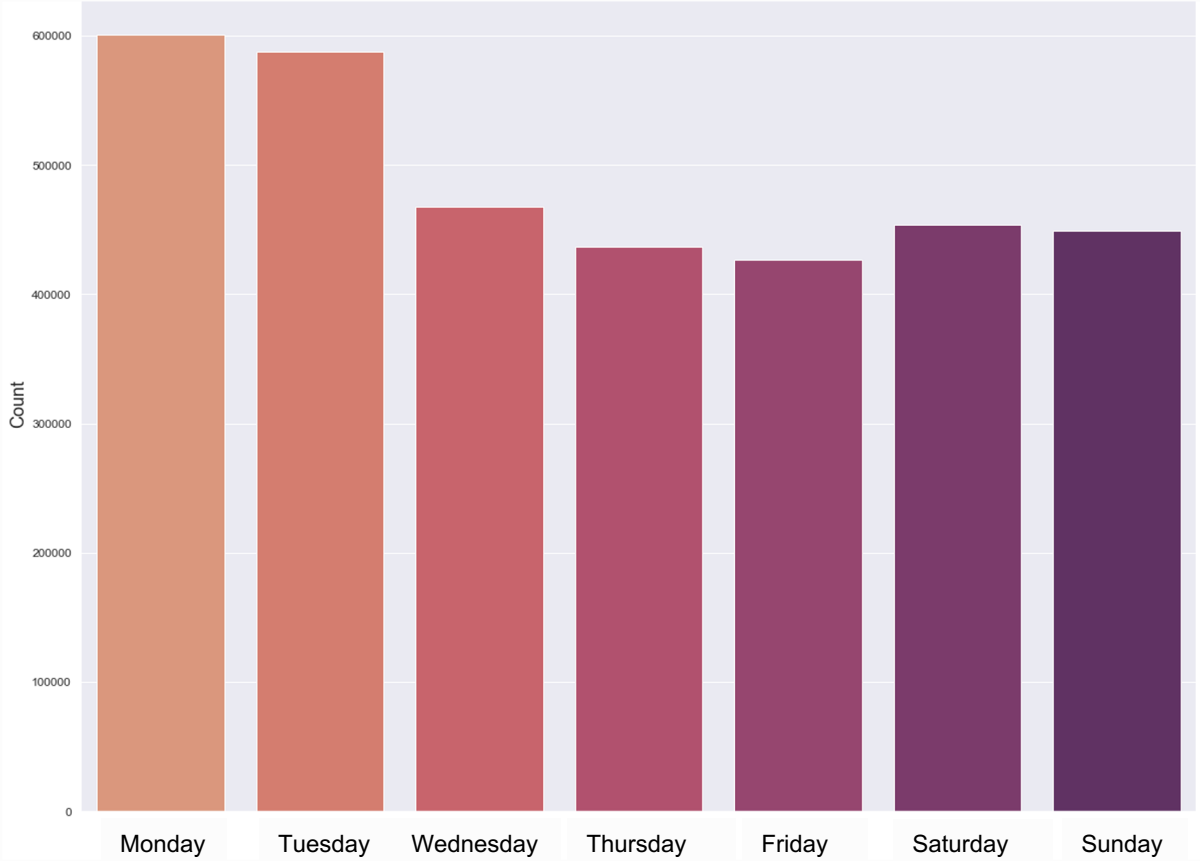

- Top product is Banana
- All top products belong to the department of produce

# Departments Distribution

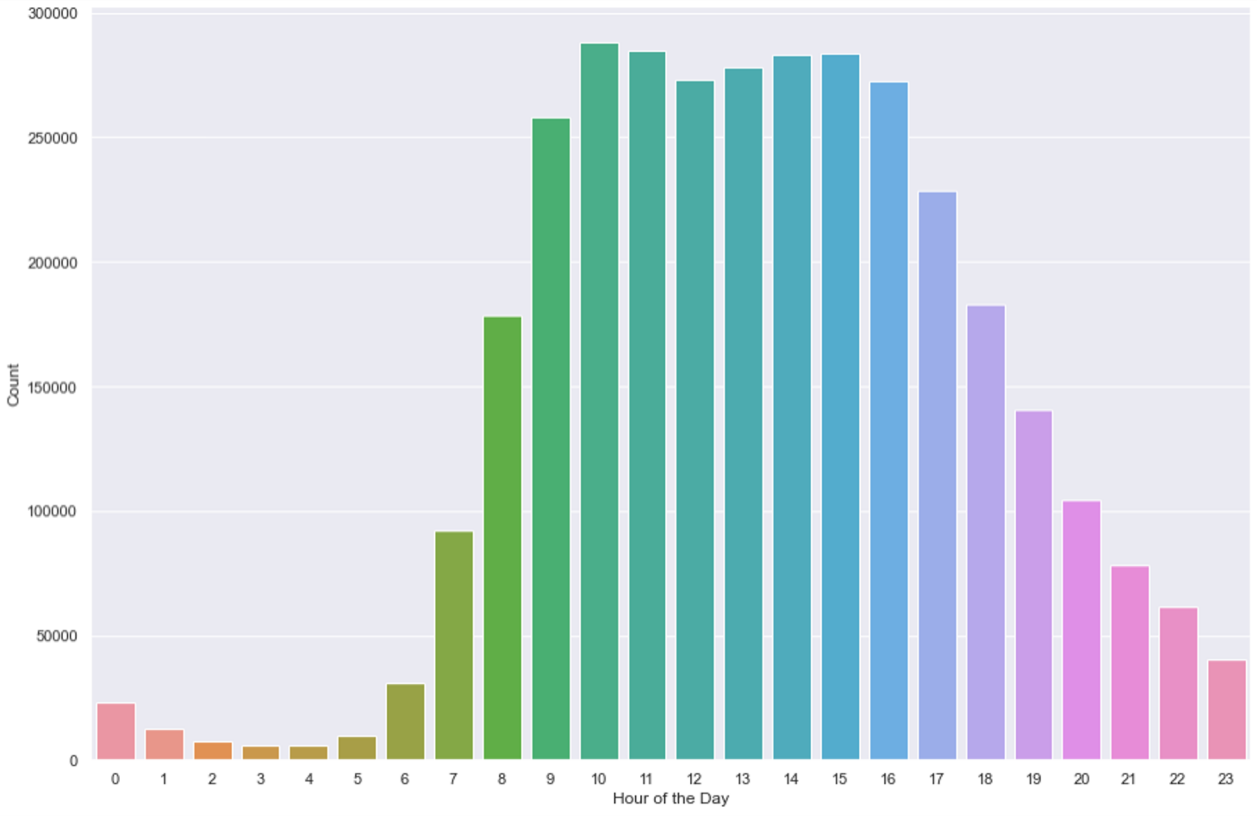



- Top department is Produce
- Top aisles for the produce department are fresh vegetables and fresh fruits

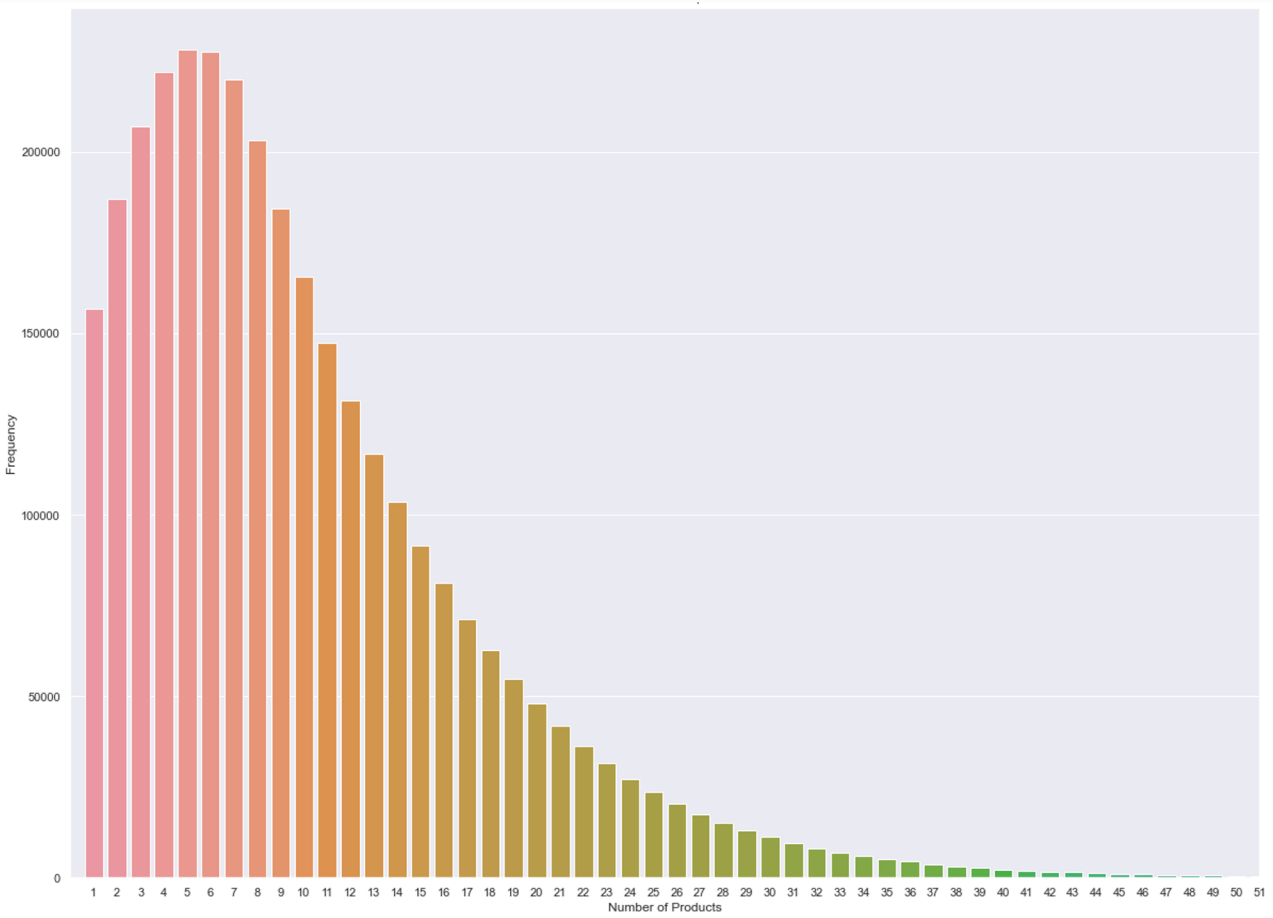
# Week Distribution



# Hour Distribution

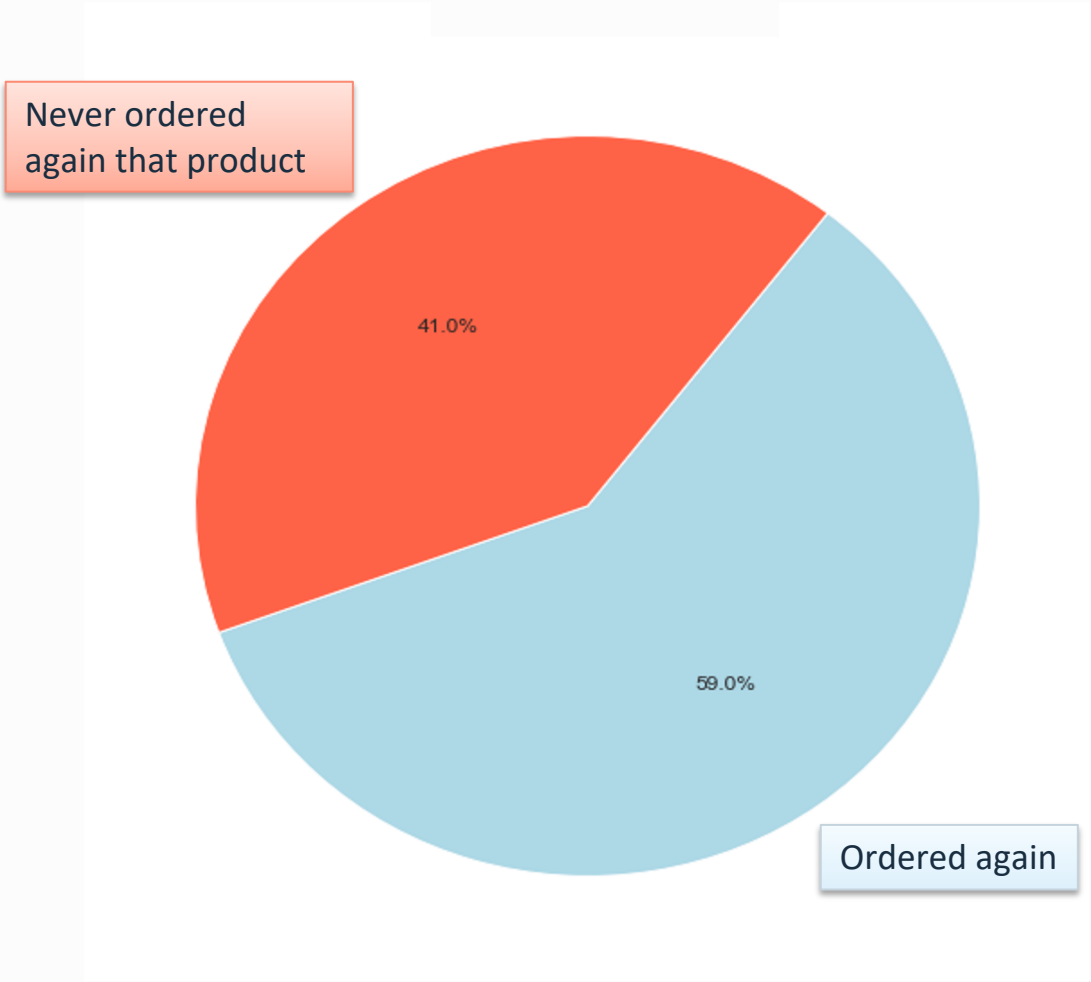


# Number of products per order




- The customers usually buy around 5 products

# Probability of a product reordering



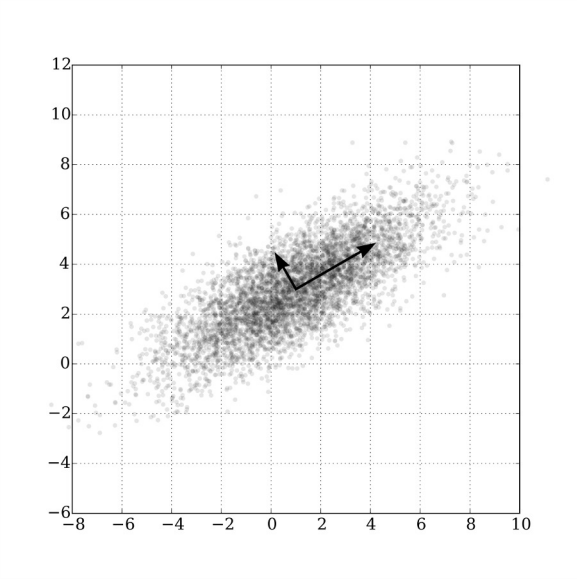
03

# Clustering

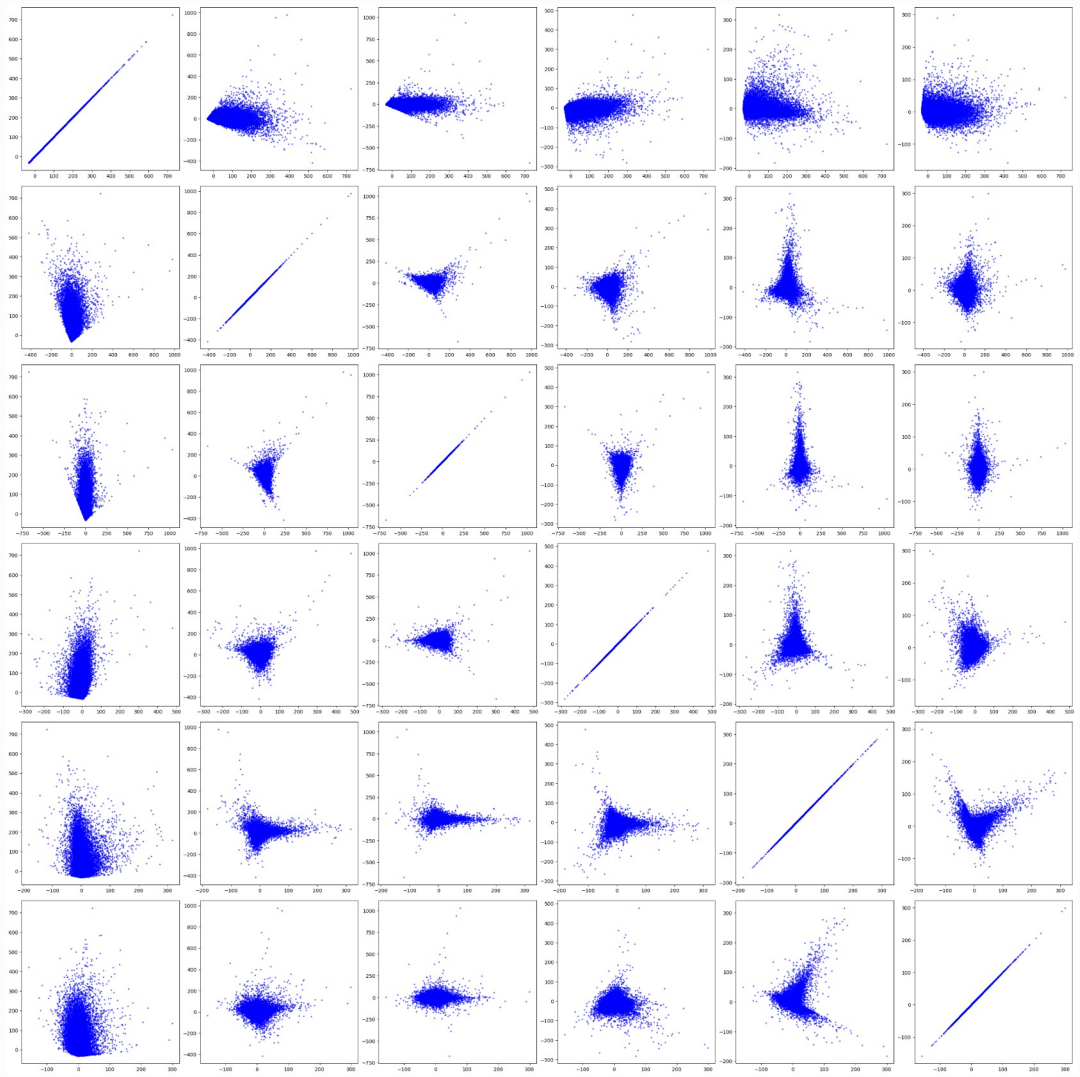
Analysis of consumer purchasing behavior

# Principal component analysis (PCA)

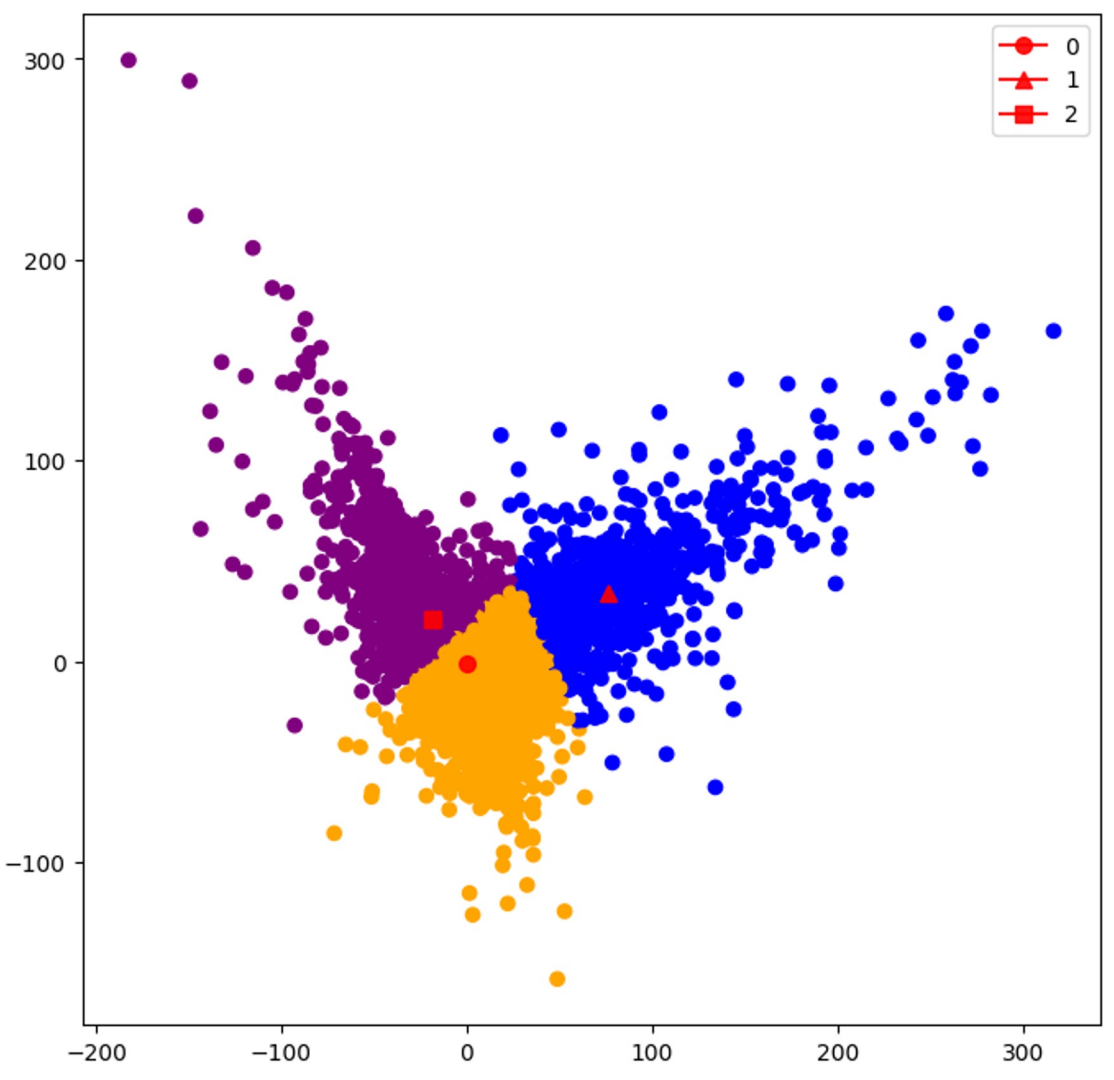
**PCA** is an unsupervised learning technique for reducing the dimensionality of a dataset. It can be accomplished by linearly transforming the data into a new coordinate system with fewer dimensions than the source data.



# PCA for six components



# K-means clustering of the distribution at (5,6)



# Identifying purchasing pattern

If we take a closer look into the clusters, we notice that the customers from cluster

- **0:** are "**generic**", meaning that they don't seem to have a preference
- **1:** are in great need of **baby food formula**
- **2:** are prone to buy **fresh produce**

```
The top 10 aisles are of clustering 0 is

aisle
fresh fruits                      16.240971
fresh vegetables                  15.738973
packaged vegetables fruits         8.438962
yogurt                             6.410504
packaged cheese                    4.812776
milk                               4.221710
chips pretzels                     3.439355
soy lactosefree                    2.914071
bread                              2.863071
water seltzer sparkling water      2.840107
Name: 0, dtype: float64

The top 10 aisles are of clustering 1 is

aisle
baby food formula                 93.260958
fresh fruits                      73.364934
fresh vegetables                  54.837920
yogurt                            34.314985
packaged vegetables fruits        32.680938
milk                              21.994903
packaged cheese                   21.955148
water seltzer sparkling water     12.240571
bread                             11.246687
soy lactosefree                   10.597350
Name: 1, dtype: float64

The top 10 aisles are of clustering 2 is

aisle
fresh fruits                      61.289183
fresh vegetables                  52.483850
water seltzer sparkling water     35.252547
yogurt                            25.238890
packaged vegetables fruits        18.768697
energy granola bars               10.169087
soy lactosefree                    9.774984
refrigerated                       8.765012
chips pretzels                     8.520919
milk                               8.158248
Name: 2, dtype: float64
```

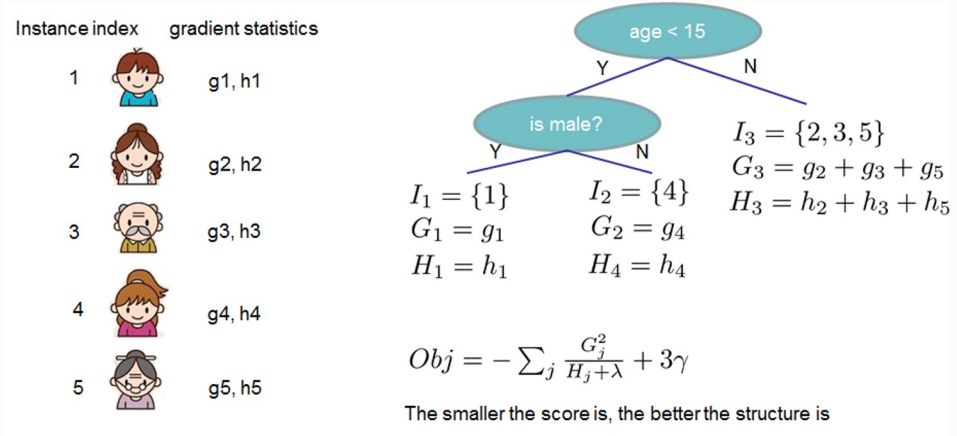
04

# Machine Learning

Prediction of the future purchases

# Extreme gradient boosting (XGBoost)

XGBoost is a supervised learning method based on gradient boosted trees. XGBoost works as Newton-Raphson in function space and a second order Taylor approximation is used in the loss function.



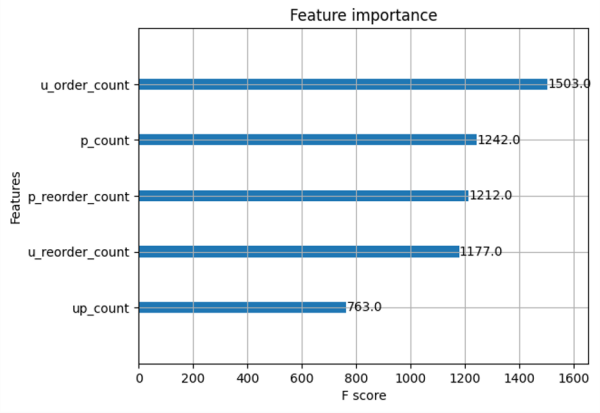
# Important features for customer's future purchases

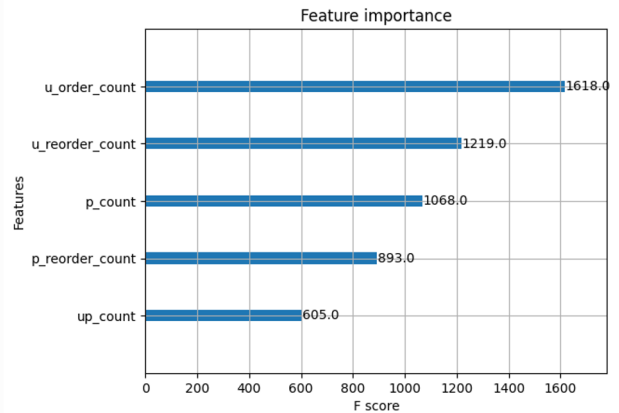
## The critical factors are:

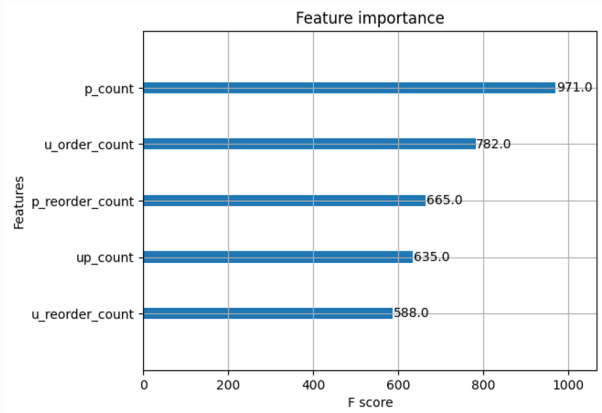
1. Purchases for every user & product combination
2. Reorders for every user & product combination
3. Reorders in the last five orders for every user & product combination
4. Total purchases for every user
5. Total reorders for every user
6. Total purchases of every product
7. Total reorders of every product

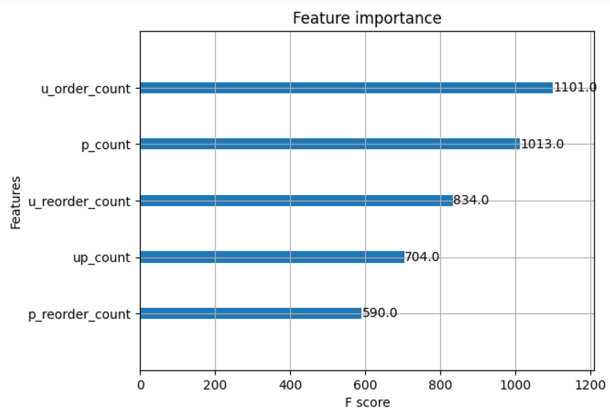
# Feature plots of XGBoost models

We plot the feature importance for the non clustered and cluster type 0, 1, 2 data, respectively



# Hyperparameter tuning with GridSearchCV for optimal prediction

```
paramGrid = {"max_depth":[5,10],
             "colsample_bytree":[0.3,0.4]}
xgbc = xgb.XGBClassifier(objective='binary:logistic', eval_metric='logloss', num_boost_round=10)
gridsearch = GridSearchCV(xgbc, paramGrid, cv=3, verbose=2, n_jobs=1)
model = gridsearch.fit(X_train, y_train)
print("The best parameters are: /n",  gridsearch.best_params_)
# Store the model for prediction
model = gridsearch.best_estimator_
```

```
Fitting 3 folds for each of 4 candidates, totalling 12 fits
[22:43:29] WARNING: /Users/runner/work/xgboost/xgboost/python-package/build/temp.macosx-10.9-x86_64-cpython-38/xgbo
ost/src/learner.cc:767:
Parameters: { "num_boost_round" } are not used.

[CV] END ..................colsample_bytree=0.3, max_depth=5; total time= 1.2min
[22:44:42] WARNING: /Users/runner/work/xgboost/xgboost/python-package/build/temp.macosx-10.9-x86_64-cpython-38/xgbo
ost/src/learner.cc:767:
Parameters: { "num_boost_round" } are not used.

[CV] END ..................colsample_bytree=0.3, max_depth=5; total time= 1.2min
[22:45:51] WARNING: /Users/runner/work/xgboost/xgboost/python-package/build/temp.macosx-10.9-x86_64-cpython-38/xgbo
ost/src/learner.cc:767:
Parameters: { "num_boost_round" } are not used.

[CV] END ..................colsample_bytree=0.3, max_depth=5; total time= 1.0min
[22:46:53] WARNING: /Users/runner/work/xgboost/xgboost/python-package/build/temp.macosx-10.9-x86_64-cpython-38/xgbo
ost/src/learner.cc:767:
Parameters: { "num_boost_round" } are not used.
```



```
[CV] END ..................colsample_bytree=0.4, max_depth=10; total time= 1.4min
[22:56:56] WARNING: /Users/runner/work/xgboost/xgboost/python-package/build/temp.macosx-10.9-x86_64-cpython-38/xgbo
ost/src/learner.cc:767:
Parameters: { "num_boost_round" } are not used.

[CV] END ..................colsample_bytree=0.4, max_depth=10; total time= 1.4min
[22:58:20] WARNING: /Users/runner/work/xgboost/xgboost/python-package/build/temp.macosx-10.9-x86_64-cpython-38/xgbo
ost/src/learner.cc:767:
Parameters: { "num_boost_round" } are not used.

The best parameters are: /n {'colsample_bytree': 0.3, 'max_depth': 5}
```

# Predictions and scores

We train XGBoost models on the unclustered data as well as the clustered data, which result in 4 models, and their predictions and scores are as follows:

| | Non clustered model | Clustered 0 model | Clustered 1 model | Clustered 2 model |
|---|---|---|---|---|
| Number of predictions | 6785793 | 580703 | 15809 | 42369 |
| Accuracy | 86.30% | 79.46% | 90.33% | 87.89% |
| F1 score | 0.3712 | 0.2816 | 0.2892 | 0.3566 |

The clustered model generally has better accuracy and lower F1 scores as well.

# Predictions for the most valuable customers

Considering the most valuable customers as anyone who has had more than 100 orders, we apply our model, and the accuracies are:

| Accuracy | Non clustered model | Cluster 0 model | Cluster 1 model | Cluster 2 model |
|---|---|---|---|---|
| Cluster 0 data | 91.23% | 91.32% | | |
| Cluster 1 data | 94.01% | | 94.05% | |
| Cluster 2 data | 90.44% | | | 92.51% |

The clustered 2 model is greatly better at predicting customers who likes to buy fresh produce

# Product model precision

## Some reordering items that we got right

- Organic Grade A Free Range Large Brown Eggs,
- Organic Yellow Onion,
- Organic Cucumber,
- Organic Tomato Cluster,
- Free & Clear Unscented Baby Wipes
- …

## Some products that we have missed

- Organic Lemon,
- Organic Red Onion,
- Organic Sunday Bacon
- …

The list goes on…

# Summary

## Problem

- Use customer Instacart orders over time to predict which previously purchased products will be in a user's next order
- Understand the consumer trends and customer behavior

## Successes

- Analyzed the customer trends
- Provided an algorithm for the customer categorization
- Provided 2 techniques for customer order prediction

## Future Work

- Investigating further how XGBoost features vary on different data set
- Test more ML algorithms like lightGBM, feedforward neural network, recursive neural network etc.

# Thank you! Any questions?

**Linkedin profiles:**

https://www.linkedin.com/in/haoran-li-80027b23b/

https://www.linkedin.com/in/thanasis-kritikos-41665279/