Personalised recommendations to increase AOV of Instacart loyalists

Agenda

Introduction

- About Instacart
- Problem Statement

EDA

- Exploratory Data Analysis

Modelling

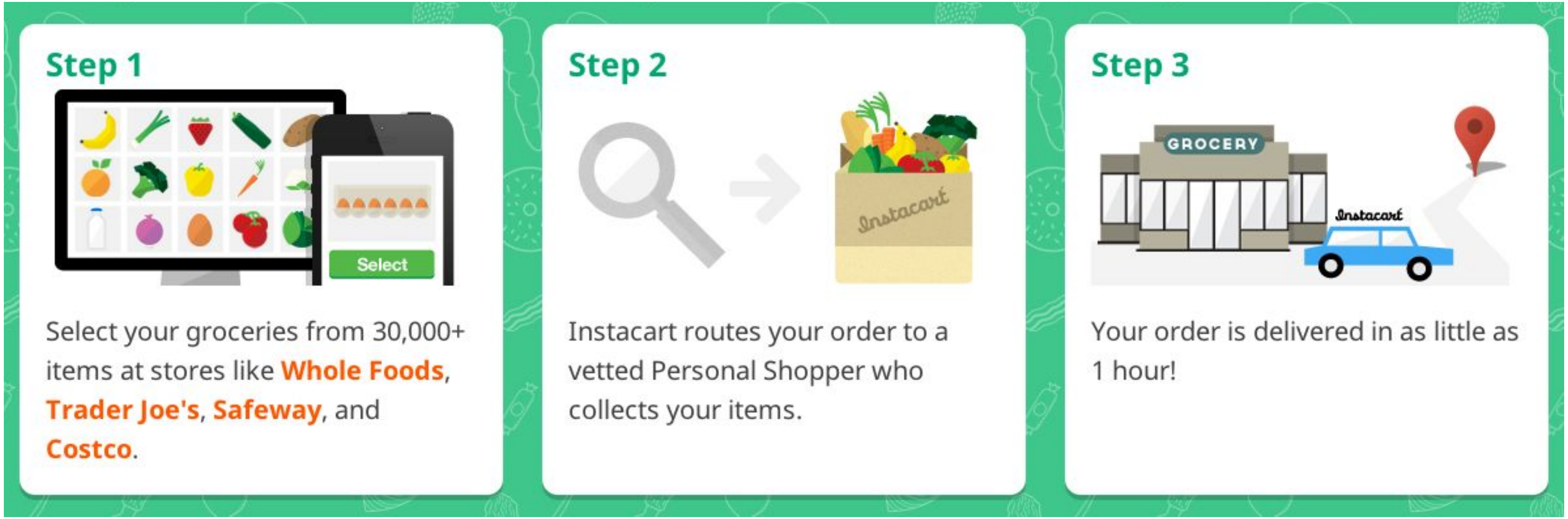
- Implementation of Recommender Systems
- Evaluation and Conclusion

Introduction

About Instacart

Web and mobile-based

On-demand grocery delivery service



instacart

10021 Hi, Zi Xin Help 0

Big-selection grocers

NEW
Fairway
Delivery

NEW
The Food Emporium Shoppe
Delivery

NEW
The Market @ Mercedes House
Delivery

ALDI
Delivery • Pickup 18.9mi

The Food Emporium
Delivery

Key Food
Delivery • Pickup 6.0mi

Westside Market
Delivery

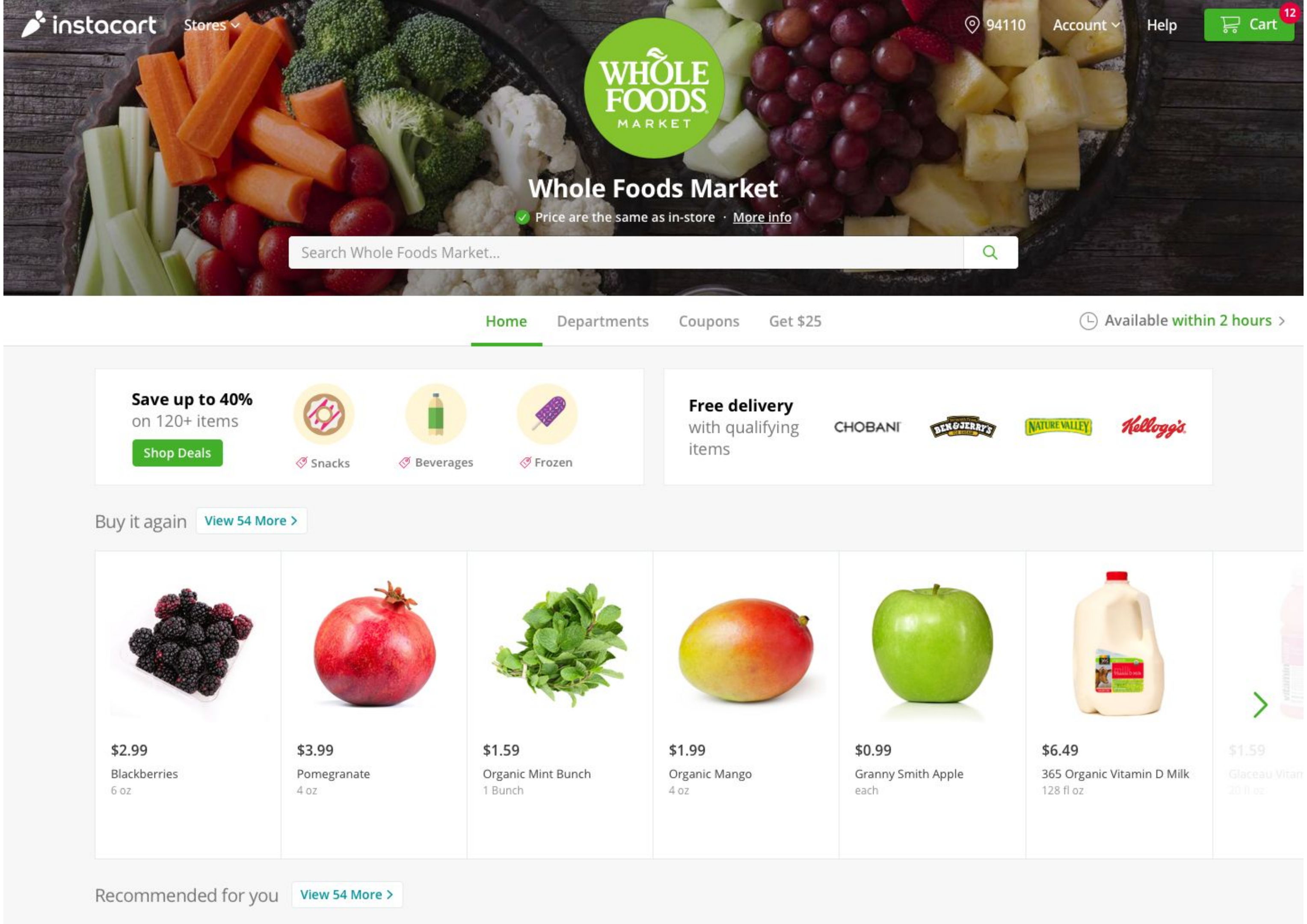
Food Universe
Delivery • Pickup 2.6mi

ShopRite
Delivery

Key Food Marketplace
Delivery

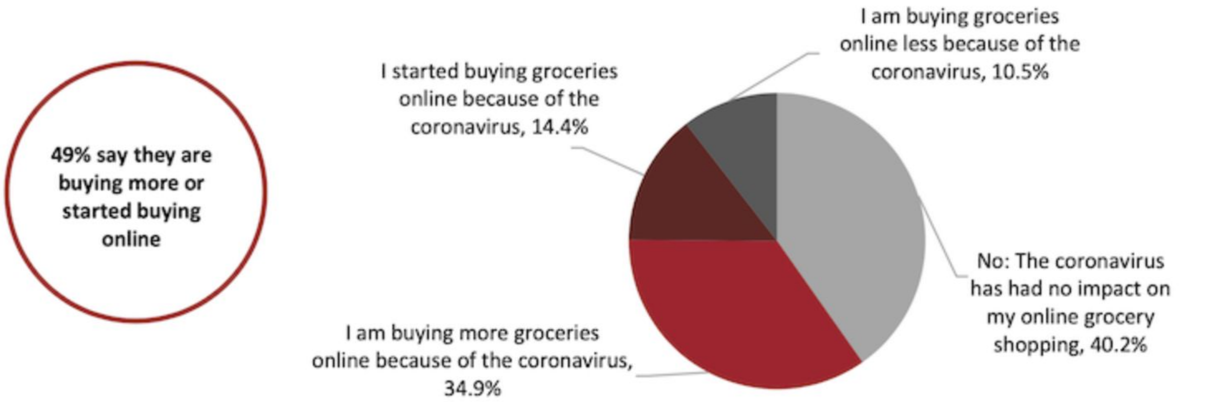
Wegmans
Delivery • Pickup 20.1mi

D'Agostino
Delivery


instacart
Stores
94110
Account
Help
Cart
12
WHOLE FOODS MARKET
Whole Foods Market
Price are the same as in-store · More info
Search Whole Foods Market...
Home
Departments
Coupons
Get $25
Available within 2 hours
Save up to 40%
on 120+ items
Shop Deals
Snacks
Beverages
Frozen
Free delivery
with qualifying items
CHOBANI
BEN & JERRY'S
NATURE VALLEY
Kellogg's
Buy it again
View 54 More
$2.99
Blackberries
6 oz
$3.99
Pomegranate
4 oz
$1.59
Organic Mint Bunch
1 Bunch
$1.99
Organic Mango
4 oz
$0.99
Granny Smith Apple
each
$6.49
365 Organic Vitamin D Milk
128 fl oz
$1.59
Recommended for you
View 54 More

Problem Statement – The Digital Fight is On

- As online grocery sales continue to surge, Instacart is facing increasing competition in the online grocery delivery space.
- Many traditional grocers are rapidly expanding their online operations. For example, Walmart has been expanding its online-order offering, with same-day delivery from 1,600 stores and pickup from 3,100 locations as of November 2019.



Base: 599 US Internet users aged 18+ who have bought groceries online in the past 12 months
Source: Coresight Research

Problem Statement – The Digital Fight is On

- As online grocery sales continue to surge, Instacart is facing increasing competition in the online grocery delivery space.
- Many traditional grocers are rapidly expanding their online operations. For example, Walmart has been expanding its online-order offering, with same-day delivery from 1,600 stores and pickup from 3,100 locations as of November 2019.

How can Instacart improve their customer experience so as to retain the loyalty of their top customer base?

Through personalised recommendations!

EDA


instacart
pickup
Kale&
Avocados&
Blueberries&
Chocolate&
Family.
instacart
instacart
pickup
Order online,
pickup here!
instacart.com

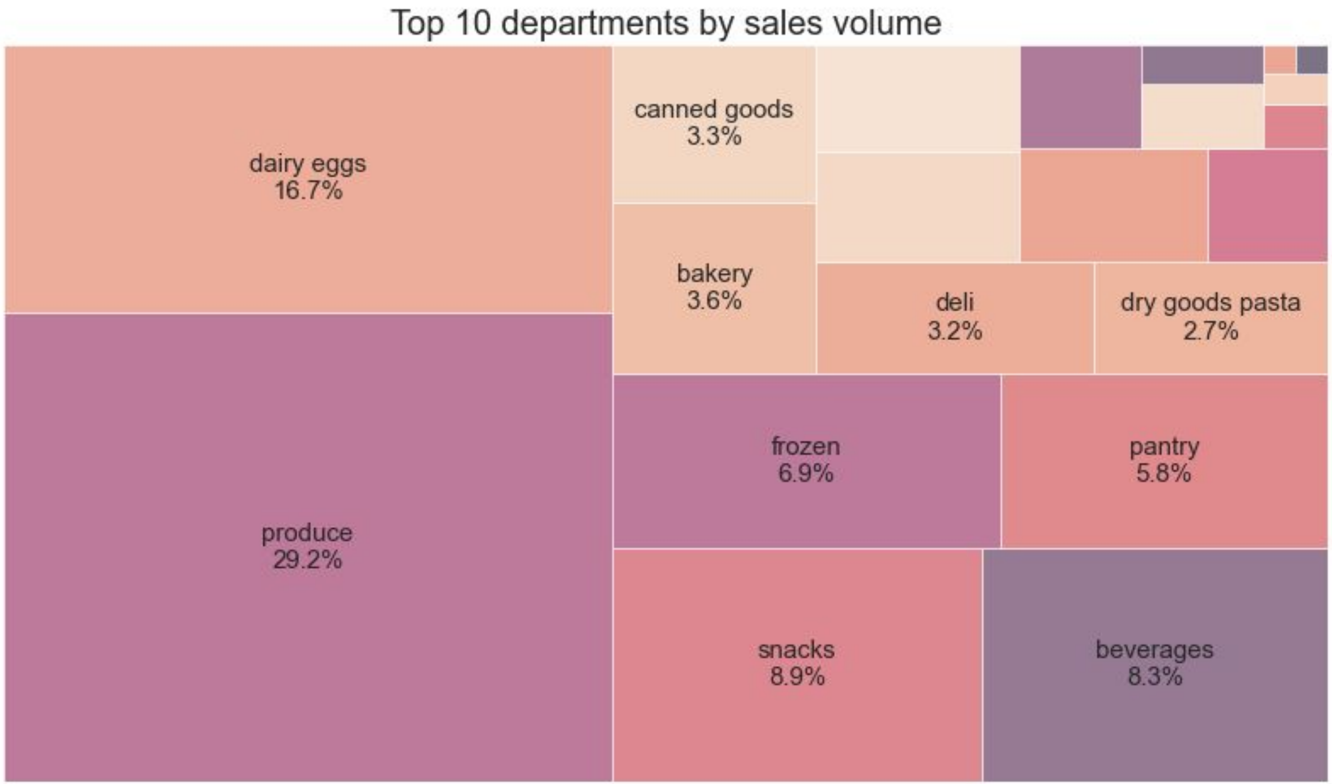
Dataset

33,819,106 rows, 12 columns

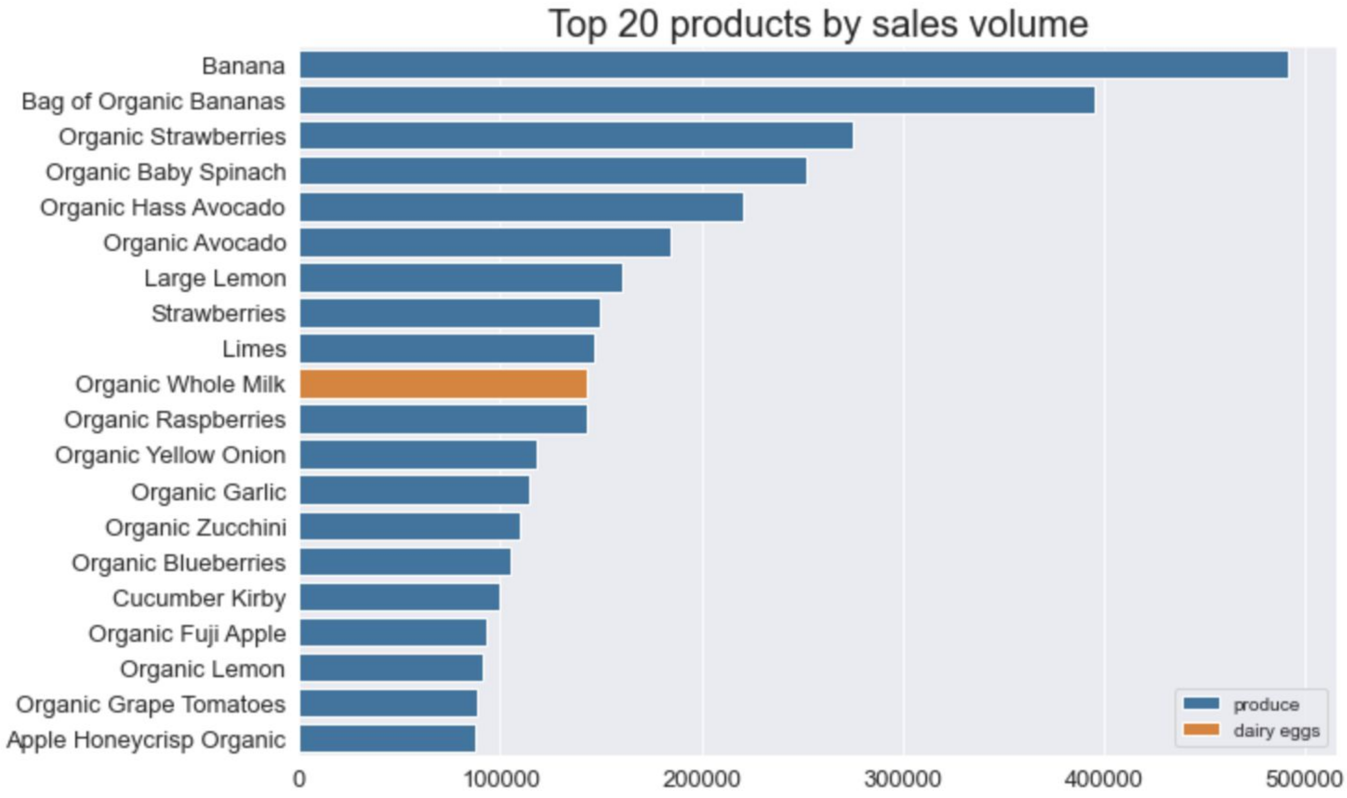
- Orders: 3.34 mil
- Users: 206,209
- Products: 49,688

order_id	user_id	order_number	order_dow	order_hour_of_day	days_since_prior_order	product_id	add_to_cart_order	reordered	product_name	department	aisle
2539329	1	1	2	8	NaN	196	1	0	Soda	beverages	soft drinks
2398795	1	2	3	7	15.0	196	1	1	Soda	beverages	soft drinks
473747	1	3	3	12	21.0	196	1	1	Soda	beverages	soft drinks

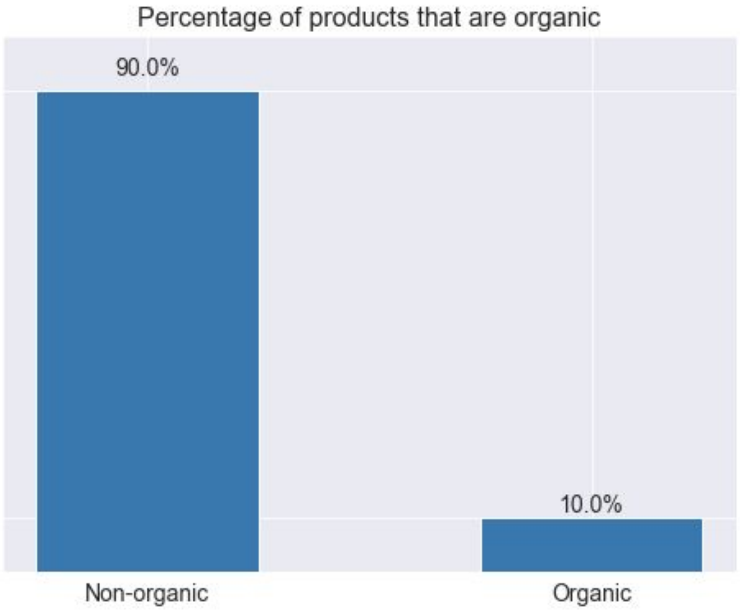
Sales by department



Top-selling items

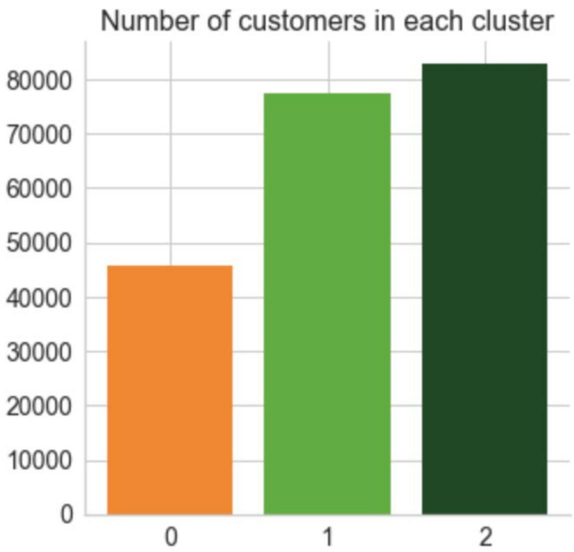


Organic items contribute a disproportionate amount to sales



Zooming in on Instacart's most loyal customers

Customer segments based on past transaction behaviour



Data reduction

EDA + customer clustering → Data reduction → Utility matrix for recommenders

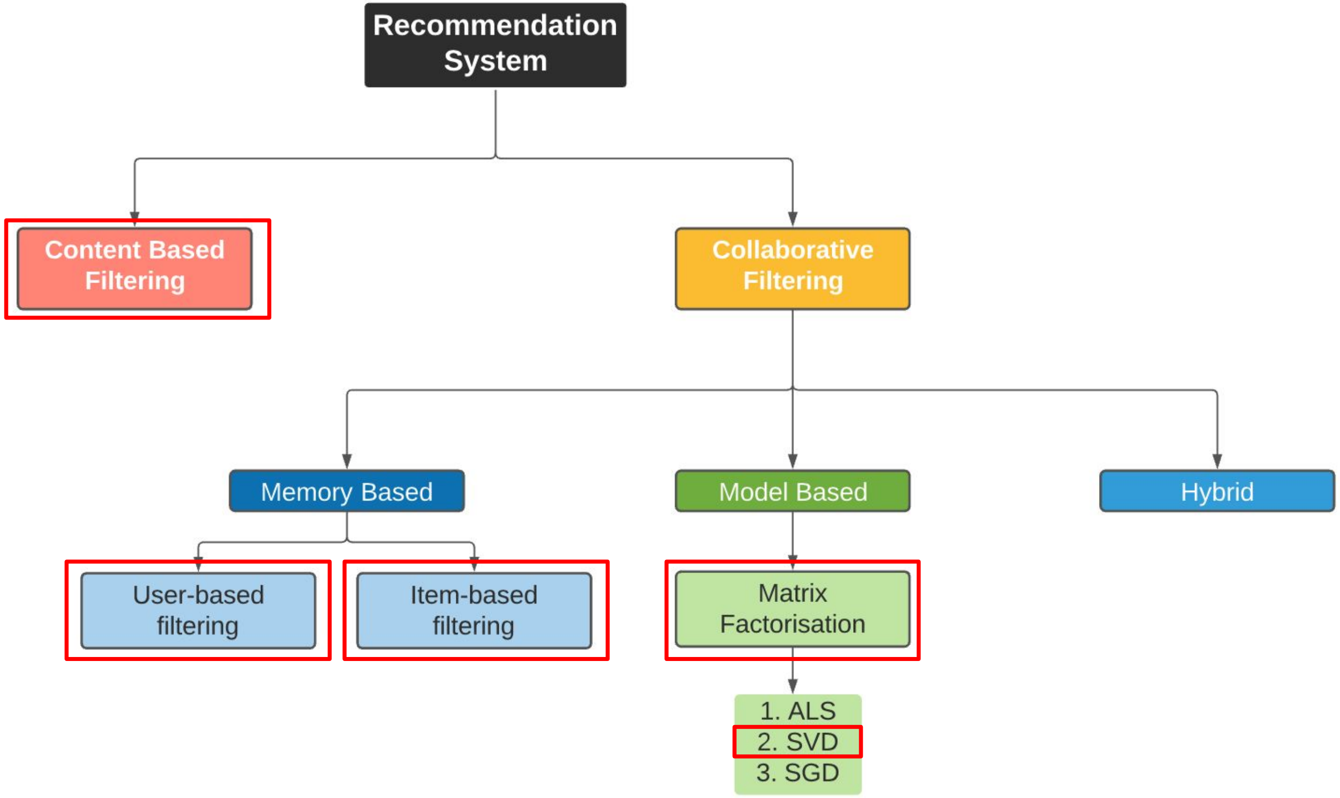
Orders: 3.34 mil
Users: 206,209
Products: 49,688

1. Drop products that were ordered <150 times
2. Drop orders with <4 items
3. Include only users with >20 orders, and keep only their last 20 orders

Orders: 830,980
Users: 41,549
Products: 16,859

Modelling

Recommender system types



General workings of Matrix Factorization

1. Decompose utility matrix

Decompose *M* into *U* and *V* such that *U*·*V* closely approximates *M* for known entries
(Number of latent factors *d* such that RMSE is minimised)

$$\begin{array}{cc} & \text{items} \\ \text{users} & \begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \end{bmatrix} \end{array} = \begin{array}{c} \text{User latent factors} \\ \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \end{array} \times \begin{array}{c} \text{Item latent factors} \\ \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix} \end{array}$$

General workings of Matrix Factorization

1. Decompose utility matrix

Decompose *M* into *U* and *V* such that *U*·*V* closely approximates *M* for known entries
(Number of latent factors *d* such that RMSE is minimised)

$$\begin{array}{cc} & \text{items} \\ \text{users} & \begin{bmatrix} 5 & 2 & 4 & 4 & 3 \\ 3 & 1 & 2 & 4 & 1 \\ 2 & \mathbf{3} & 3 & 1 & 4 \\ 2 & 5 & 4 & 3 & 5 \\ 4 & 4 & 5 & 4 & \mathbf{4} \end{bmatrix} \end{array} = \begin{array}{c} \text{User latent factors} \\ \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ u_{41} & u_{42} \\ u_{51} & u_{52} \end{bmatrix} \end{array} \times \begin{array}{c} \text{Item latent factors} \\ \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{21} & v_{22} & v_{23} & v_{24} & v_{25} \end{bmatrix} \end{array}$$

2. Recompose utility matrix

Use the entry in the product *UV* to estimate corresponding blank entries in *M*

Recommender implementation

Each model generates 20 final recommendations for the target user

User-based

CANDIDATE GENERATION

1. Find user's 20 nearest neighbours
2. All past purchases of these 20 neighbours form potential recs

RANKING

1. Rank potential recs based on sales
2. Select the top 20 bestselling items as final recs

Item-based

CANDIDATE GENERATION

1. Select user's top 20 purchases
2. For each item, find 10 most similar items
3. 200 potential recs in total

RANKING

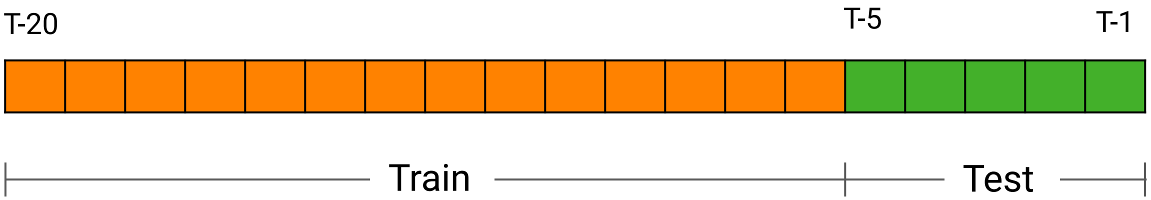
Same procedure as above

SVD

Select the top 20 items with the highest scores from the reconstructed utility matrix.

Train-test-split at the user level

E.g. user 32589's last 20 orders



Evaluation Metrics

$$\text{recommender system precision:} \quad P = \frac{\text{\# of our recommendations that are relevant}}{\text{\# of items we recommended}}$$

20 items

$$\text{recommender system recall:} \quad r = \frac{\text{\# of our recommendations that are relevant}}{\text{\# of all the possible relevant items}}$$

All items in user's last 5 orders
(test set)

Evaluating the recommender systems

Collaborative filtering models performed the best.

model	eval_set	F1	precision	recall
content-based	train	0.045174	0.116083	0.035937
content-based	test	0.046903	0.068783	0.038880
baseline	train	0.108100	0.264833	0.074087
baseline	test	0.114044	0.173317	0.091570
CF user-based	train	0.111239	0.270417	0.076457
CF user-based	test	0.117076	0.176683	0.094254
CF item-based	train	0.122490	0.289817	0.084993
CF item-based	test	0.129298	0.192200	0.105910
CF SVD	train	0.098514	0.239900	0.066857
CF SVD	test	0.114802	0.173033	0.093709

Baseline: Popularity model
Non-personalised – recommend the top 20 best-selling products

Further notes

Recommendations need to be made contextual

In reality, e-commerce retailers employ different product recommendation strategies for different pages on the site, such as:

- **Homepage**: "Recently viewed" / "Buy it again" / "Recommended For You"
- **Category pages**: "Most Popular in Category" / "Recommended For You" (category-specific)
- **Product detail pages (PDPs)**: "Similar Products" / "Often Bought Together"
- **Cart pages**: "Often Bought Together" (showcasing products that are slightly cheaper than those in a user’s cart can lead to quick purchase decisions)
- **Search results page**: Results returned from search queries can also be considered a form of recommendation. These items are ranked by probability of purchase.

More granularity needed to truly personalise the recommendations

The recommendations generated may not be user-specific enough – we need to set additional rules to generate smarter recommendations.

- We wouldn't want to recommend non-vegan items to a vegan customer.
- Consumers who have high average order values can be recommended more highly profitable items in order to maximise revenue.
- We don't want to recommend items from the user's most recent basket, especially for items that are not weekly purchase items.

Consider diversity of recommendations too

- My recommendation systems are biased towards recommending items that have relatively high sales volume, so they are probably unable to surface truly novel items that have not been discovered by many other people.
- We can improve the diversity of recommendations by recommending these long-tail items to increase the novelty factor for the user.
- Include recommendations from stores customers may have never shopped from previously. This is appropriate for customers who have a high unique-items-to-total-items ratio.

The End :)