

E6893 Big Data Analytics:

Market Basket Analysis: What's in your shopping cart?

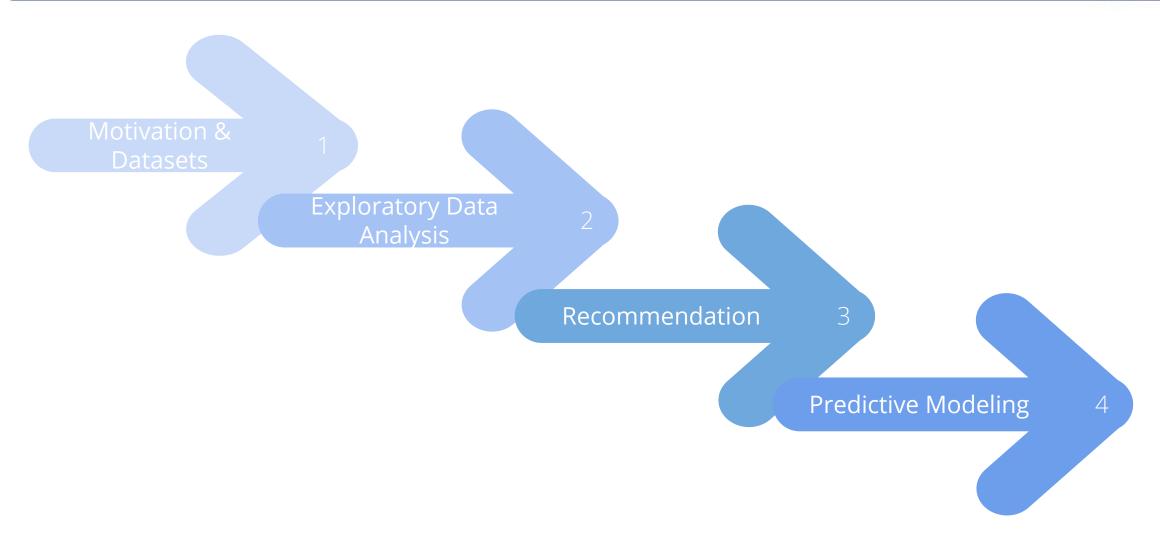
Team Members: Yini Zhang (yz3005)

Chenyun Zhu (cz2434)



Project Pipeline





Motivation



- Gain more users
- Provide delightful shopping experience to increase customer retention



Instacart is an American company that operates as a **same-day grocery delivery service**

- No need to go to grocery store to shop your favorites and save time
- Make it easy to fill your refrigerator and pantry with your personal favorites



Datasets



The Instacart Online Grocery Shopping Dataset 2017

- Relational set of files describing customers' orders
- 3,346,083 orders for 49,685 products

	order_id	product_id	add_to_cart_order	reordered
0	2	33120	1	1
1	2	28985	2	1
2	2	9327	3	0
3	2	45918	4	1
4	2	30035	5	0

product_id		product_name	aisle_id	department_id	
0	1	Chocolate Sandwich Cookies	61	19	
1	2	All-Seasons Salt	104	13	
2	3	Robust Golden Unsweetened Oolong Tea	94	7	
3	4	Smart Ones Classic Favorites Mini Rigatoni Wit	38	1	
4	5	Green Chile Anytime Sauce	5	13	

	order_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	days_since_prior_order
0	2539329	1	prior	1	2	8	NaN
1	2398795	1	prior	2	3	7	15.0
2	473747	1	prior	3	3	12	21.0
3	2254736	1	prior	4	4	7	29.0
4	431534	1	prior	5	4	15	28.0

aisle	sle_id	ai	department	rtment_id	de
prepared soups salads	1	0	frozen	1	0
specialty cheeses	2	1	other	2	1
energy granola bars	3	2	bakery	3	2
instant foods	4	3	produce	4	3
marinades meat preparation	5	4	alcohol	5	4

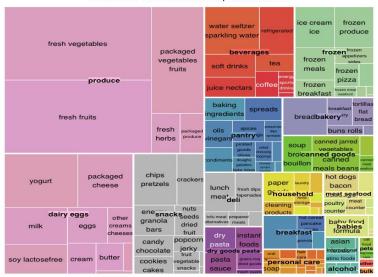


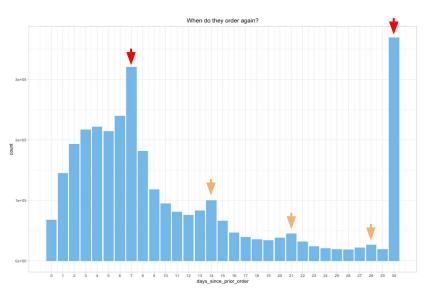
Exploratory Data Analysis

Exploratory Data Analysis

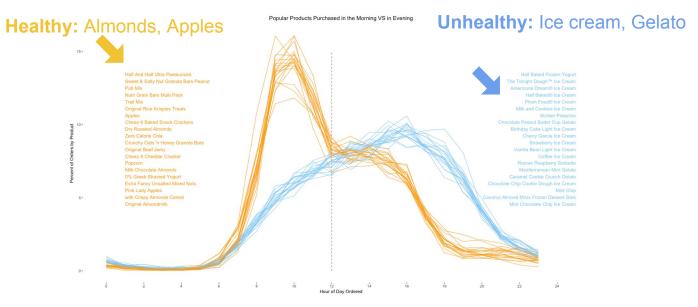














RECOMMENDATION



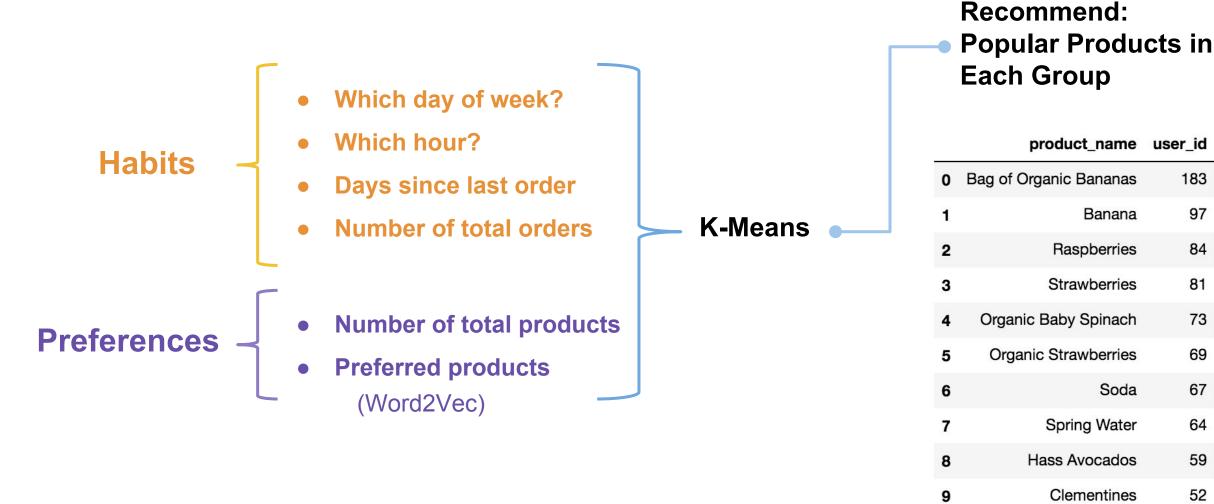
Recommend new products



IN GROUPS

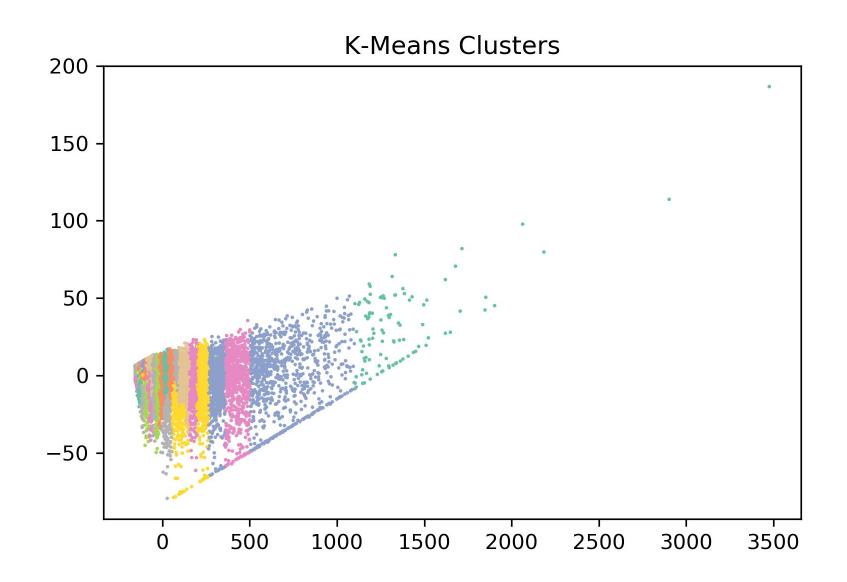
Recommend New Products: Steps





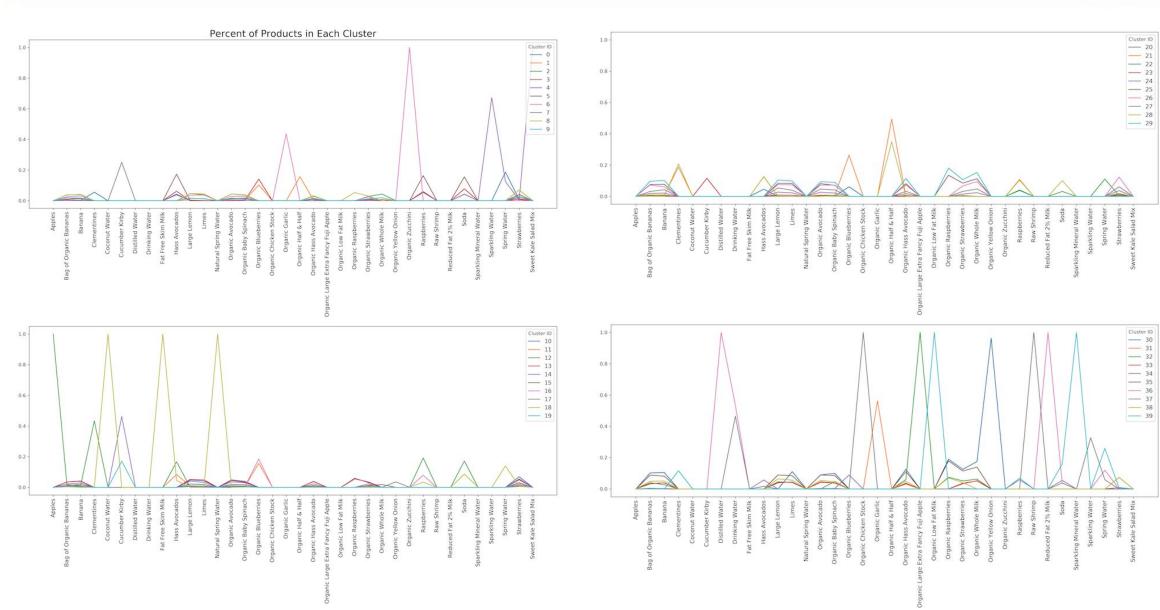
K-Means Visualization





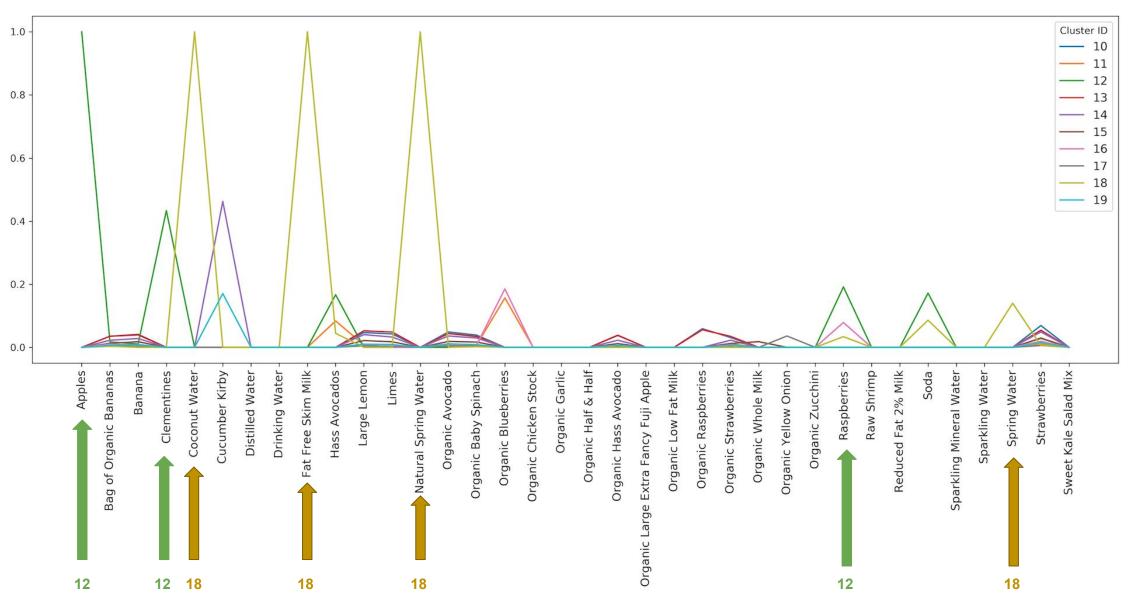
Popular Products in Each Group





Popular Products in Cluster 10-19

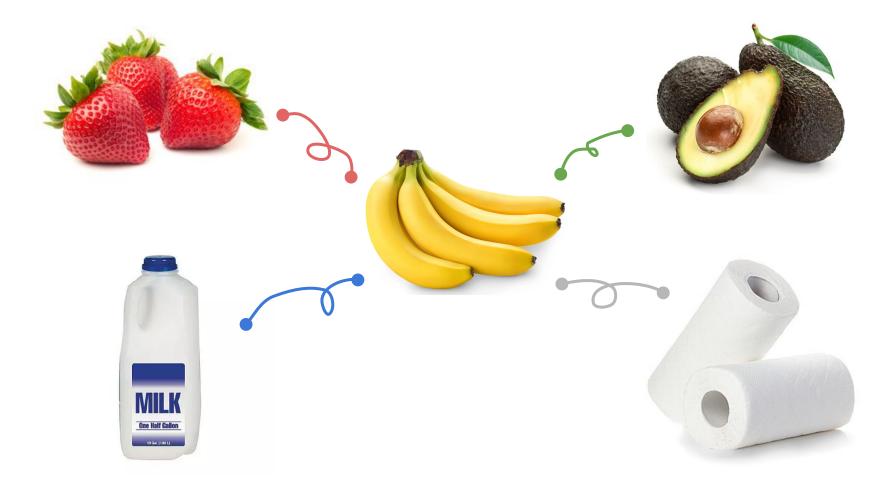




Recommendation Part II

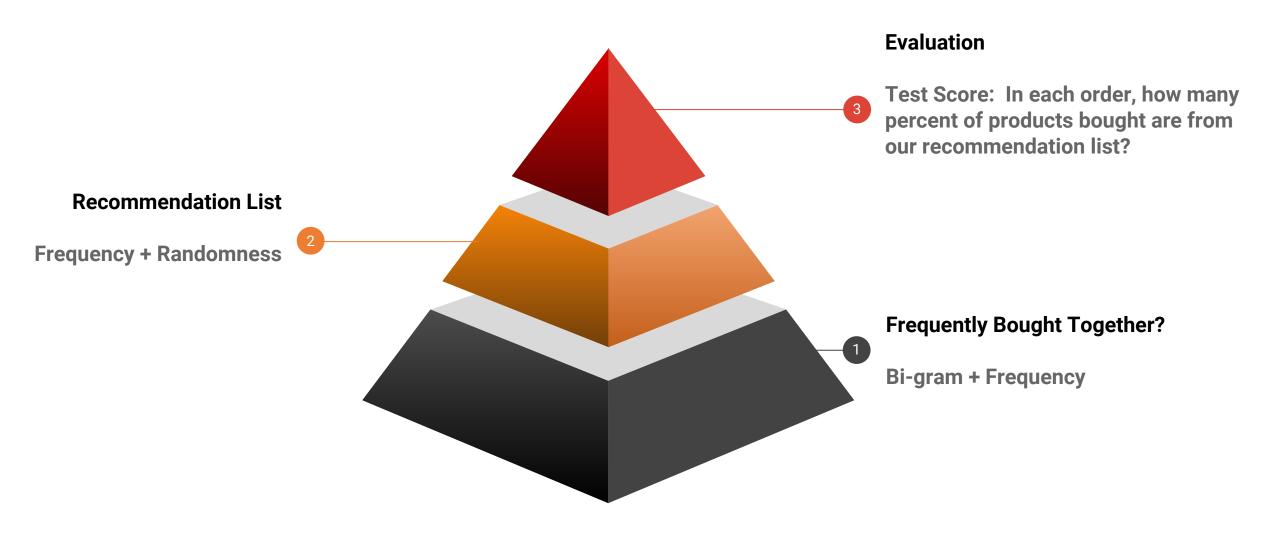


Recommend product bundles



Recommendation Product Bundles: Steps





Bi-grams



Popular Bundles:

```
Organic_Hass_Avocado + Bag_of_Organic_Bananas : 26
Banana + Organic_Avocado : 36
Banana + Organic_Fuji_Apple : 25
Banana + Honeycrisp_Apple : 23
Banana + Organic_Strawberries : 27
Bag_of_Organic_Bananas + Organic_Strawberries : 31
Bag_of_Organic_Bananas + Organic_Hass_Avocado : 30
Bag_of_Organic_Bananas + Organic_Baby_Spinach : 23
Organic_Avocado + Banana : 22
Large_Lemon + Limes : 24
```



Generate Recommendation List



1. Sort bi-gram frequencies

For example, if we have {'apple': {'strawberry': 5, 'avocado':5, 'banana': 7, 'milk': 1}}, it will be sorted as

```
'apple'+'banana': 7
'apple'+'avocado': 5
'apple'+'strawberry': 5
'apple'+'milk': 1
```

2. Specify number of recommendation k

- k = 3 : banana, avocado, strawberry
- k = 2 : banana + randomly pick one in {avocado, strawberry}
- k > 4 : all four + new recommendation of 'banana'

```
print(getRecommend("Organic_Mint_Bunch", 15))

['Organic_Italian_Parsley_Bunch', 'Garlic', 'Organic_Carrot_Bunch', 'Fresh_Cauliflower', 'Organic_Cilantro', 'Organic_Baby_Spinach_Salad', 'Organic_Cilantro_Bunch', 'Organic_Thyme', '100%_Pressed_Apple__Fruit_Juice', 'Organic_Mountain_Forest_Honey_Light_Amber', 'Organic_Cilantro', 'Large_Lemon', 'Organic_Mint', 'Organic_Basil', 'Organic_Garlic']
```

Evaluation





Score = right predictions / len(order)

```
scores = TestScore(test_data)
print("=====> Mean Test Scores: ", numpy.mean(scores))
=====> Mean Test Scores: 0.182374730607
```



Predictive Modeling

Feature Engineering







user total distinct product

ratio_of_daysSincePr iorOrder_avgDaysBet wOrders

15 Newly Added **Features**

product_total_reorder

length_between_orde

product_ reorder_ rate

average_po sition_for_t he_product

user_average_ite

ms_per_cart

num_order_user_ purchase_specific _product

user_product_hour

user_total_unique_item

Algorithms



- Faster training speed and higher efficiency
- Lower memory usage
- Better accuracy
- Parallel and GPU learning supported
- Capable of handling large-scale data

Reference: https://github.com/Microsoft/LightGBM

Desired Output



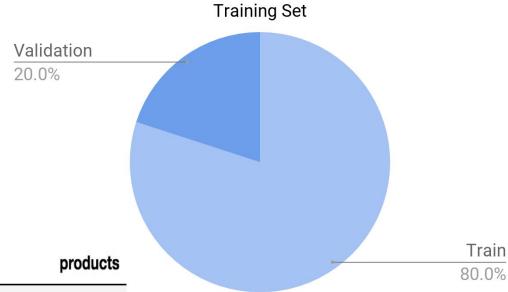
	order_id	products
0	525192	45066 13198 21137 40852 47272 29993 27690 3217
1	880375	17794 21903 14992 2078 32030 34358 28985 4799
2	2614670	46676 4957 13176 2966 14233 33754 33787
3	2436259	37131 1073 26172 5955 18653 21616
4	2906490	24838 15120 24852

What's the **threshold probability** to select product into the corresponding order?

Threshold Probability



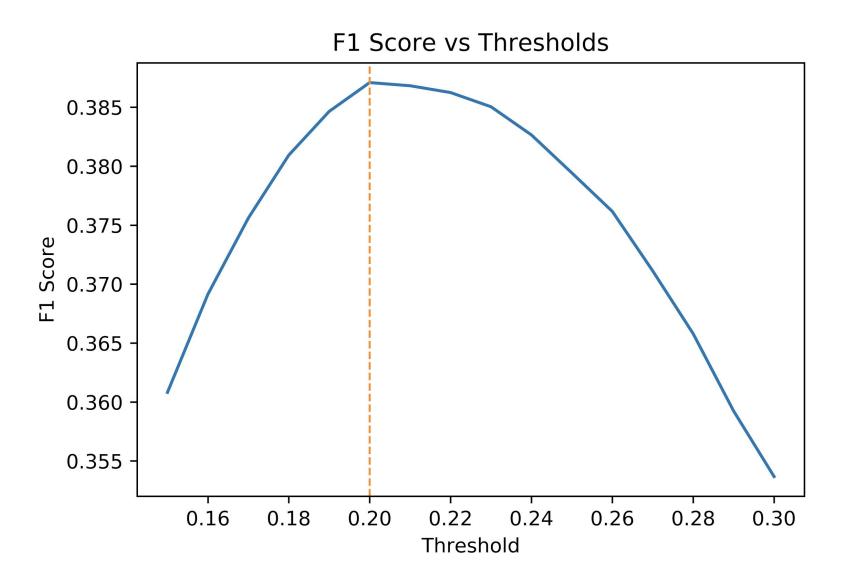
• Produce equivalent output with the train ground truth data.



	order_id	true	products
0	525192	45066 13198 40852 47272 43967 29894 17638 37999	45066 13198 21137 40852 47272 29993 27690 3217
1	880375	21903 15937 41540 23165	17794 21903 14992 2078 32030 34358 28985 4799
2	2614670	32263 37947 46676 21137 1323 4920 46906 45446	46676 4957 13176 2966 14233 33754 33787
3	3038639	33290 14874 21555 37973 42075 117 9385 48857 1	14874 21555 32347 42075 18531 47209 27243 2570
4	613340	31915 2228 49383 5876 43789 27966 13176 16249	43014 17948 47209 35951 21137 5876 8518 34126

Threshold Probability







Our Result	XGBoost	Light GBM	
'Equivalent' Result	0.383	0.384	
Kaggle Result	0.373	0.373	

Our Scores vs Kaggle Highest Scores

Submission and Description	Private Score	Public Score
lightgbm_0.20(6th)_nodow.csv a day ago by Chenyun Zhu	0.3723498	0.3729572
xgboost_0.2(2nd).csv 21 hours ago by Chenyun Zhu	0.3715731	0.3725694

#	△pub	Team Name	Score @	Entries
1		胡萝卜	0.4091449	62
2	======================================	===== KEEP OUT	0.4082039	138
3	_	sjv	0.4081041	76

Next Steps



- Get more correlated features
- Systematically tuning the models to improve accuracy

Q & A?