CLUSTERING CUSTOMERS AND FORMING BUSINESS STRATEGY

BY YUXIN ZHAO

TABLE OF CONTENT:

INTRODUCTION

Background information:

In recent years, a certain fast food restaurant has begun to collect users' information to optimize sales and promote products through mobile phone ordering apps. One of their main focuses is how their customers are formed and structured.

What is the difference in their purchasing? This problem can be solved by using data filtering and modeling, which then allows us to combine the analysis results with the actual business environment and operating conditions.

Goal:

- Use the data collected to cluster customers
- Visualize result
- Form business strategy.

STATISTICS & ANALYSIS

Data:

This study randomly selected 10000 customers' data from a certain restaurant's mobile phone application (filtered and provided by the restaurent). Customers identities and their purchase history are collected in a twelve-week period. Users are analyzed by average days between purchases, total spending, average check, number of transactions, age, race, gender, income bucket, and purchase history of products. Data are processed by Python through Jupyter Notebook, which is also used to visualize result.

	avg_days_between	total_spend	avg_check	num_transactions	bagels_count
9953	4.941176	109.94	6.467059	17	0
3850	9.333333	52.32	5.813333	9	0
4962	42.000000	30.92	15.460000	2	0
3886	84.000000	7.49	7.490000	1	0
5437	10.500000	41.42	5.177500	8	0

five of the ten thousand random samples are listed in a 5× 51 table (abridged images above / on the left).

time_last	age	race	gender	income_bucket
8	18-34	AfAm	Male	60k–74k
16	50-64	White	Female	30k–49k
44	18-34	White	Female	50k–59k
29	35-49	White	Male	100k–149k
32	65+	White	Female	60k–74k

Then we convert categorical cols to numerics for modeling intake.

```
df_model = df_model.astype({'age':'category',
                            'race':'category',
                            'gender':'category',
                            'income_bucket':'category'})

cat_columns = df_model.select_dtypes(['category']).columns
cat_columns
```

```
Index(['age', 'race', 'gender', 'income_bucket'], dtype='object')
```

```
#this conversion will insure NaN  = -1
df_model[cat_columns] = df_model[cat_columns].apply(lambda x: x.cat.codes)
```

```
df.race.unique()
```

```
array(['White', 'AfAm', 'Hispanic', 'Asian', 'Native'], dtype=object)
```

```
df_model.race.unique()
```

```
array([0, 4, 2, 1, 3])
```

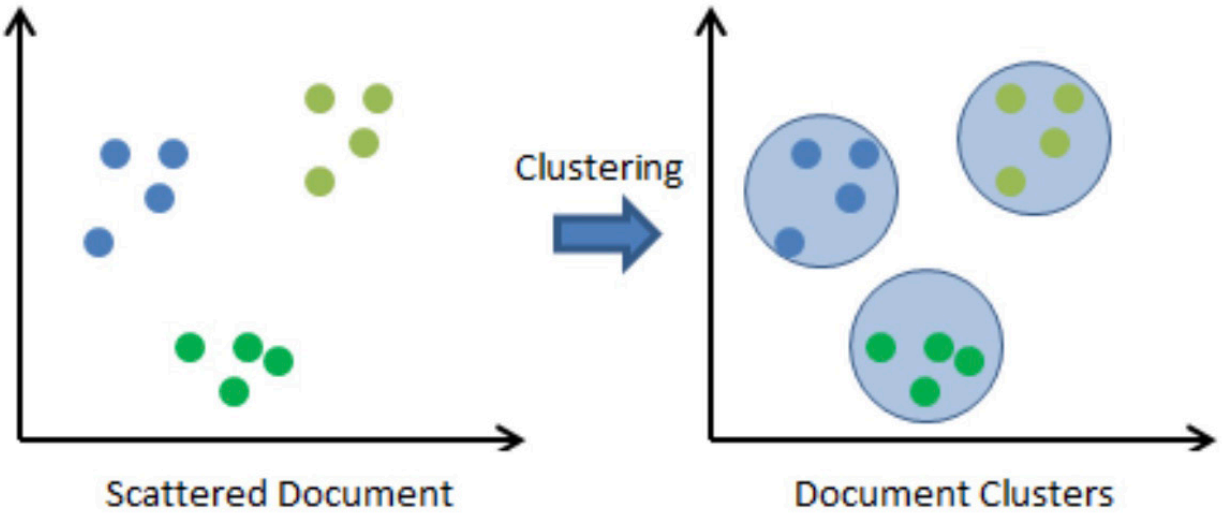
adjusted data are shown in the table below:

time_last	age	race	gender	income_bucket
8	0	0	1	4
16	2	4	0	2
44	0	4	0	3
29	1	4	1	0
32	3	4	0	4

MODEL AND CLUSTERS

Introduction of k-means:

k-means clustering is a method to partition groups with similar traits and assign them into clusters. It assigns objects to their closest cluster center according to the Euclidean distance function. Then it calculates the mean of all objects in each cluster.

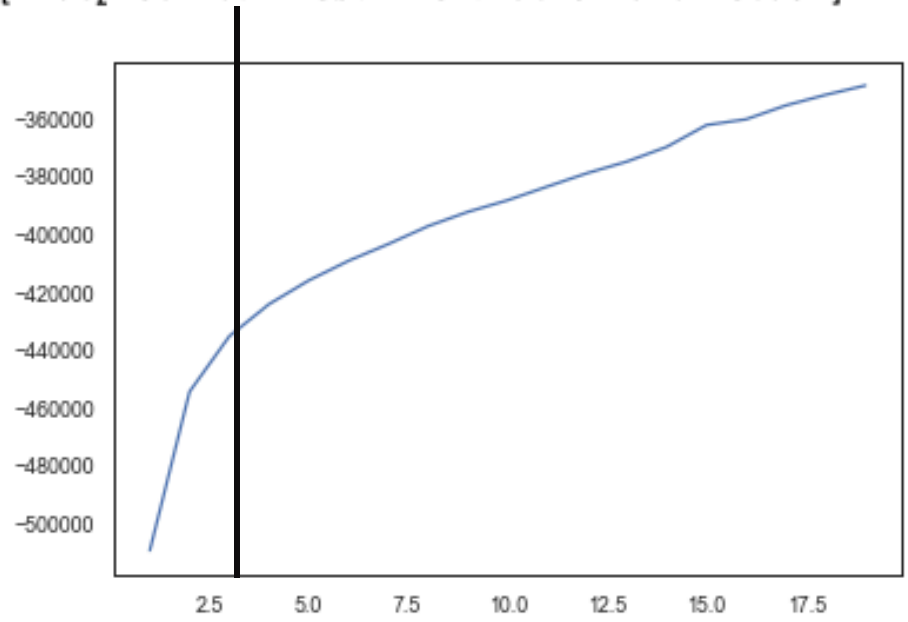


CLUSTERS

Because the sample size is relatively small, the k-means clustering model demonstrates that the restaurant's customers can be splitted into three main clusters. Then, using k-means, the characteristics of the three clusters are shown by statistics.

```
import pylab as pl
Nc = range(1,20)
kmeans = [KMeans(n_clusters = i) for i in Nc]
score = [kmeans[i].fit(final_model_array).score(final_model_array)
         for i in range(len(kmeans))]
pl.plot(Nc,score)
```

```
[<matplotlib.lines.Line2D at 0x1a2bf4eb70>]
```



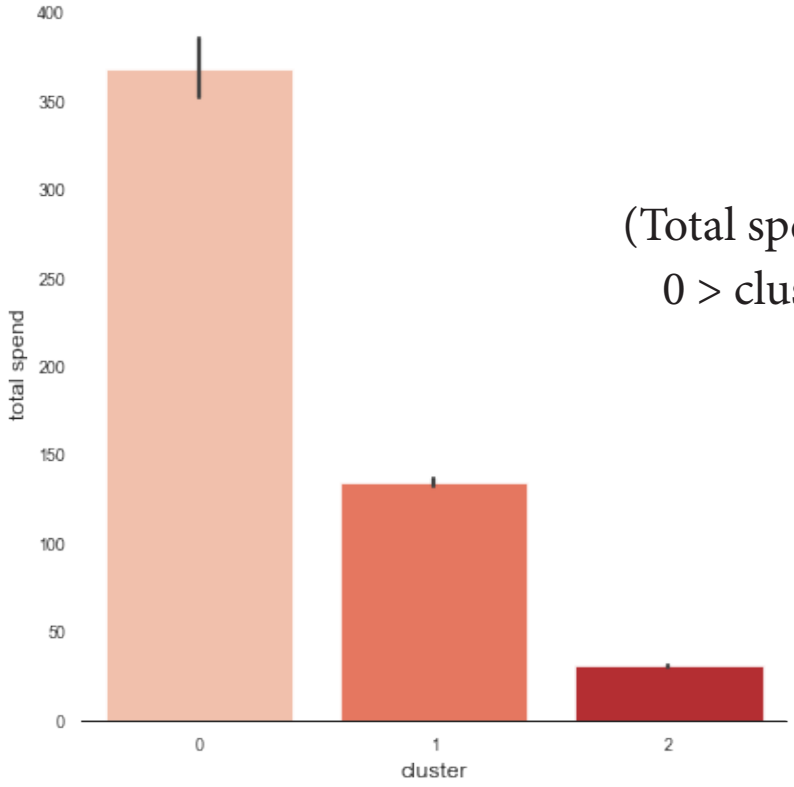
Samples are partitioned into three clusters.
Data are shown below:

Cluster Class	avg_days_between	total_spend	avg_check	num_transactions
0	1.827109	367.533663	7.327668	55.504950
1	5.878529	134.140852	8.461694	18.017620
2	34.720971	30.551237	7.716421	4.505209

3 rows × 51 columns

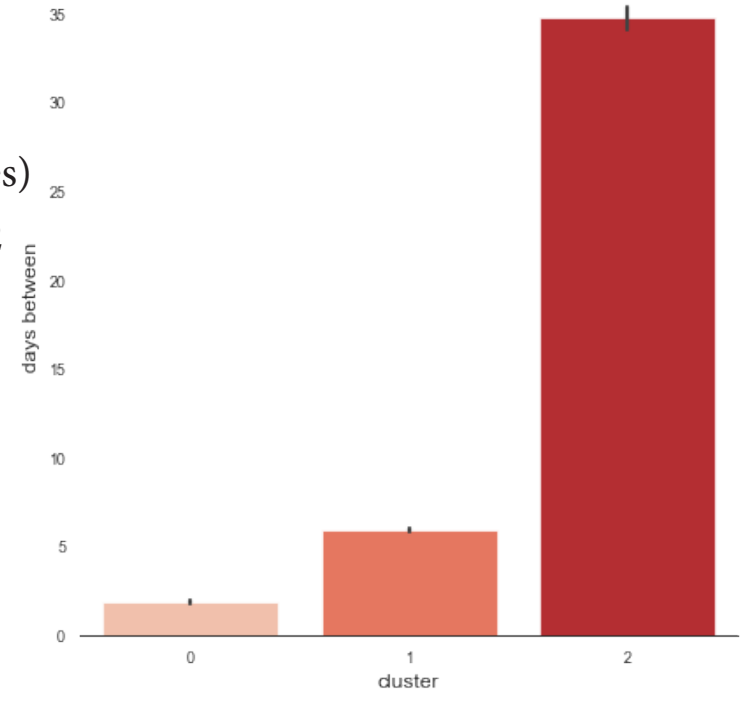
sundae_count	time_last	age	race	gender	income_bucket
0.591584	9.376238	1.415842	3.284653	0.366337	3.413366
0.227256	12.733909	1.376124	3.089177	0.406329	3.349515
0.051357	30.159648	1.505062	3.047395	0.425679	3.234336

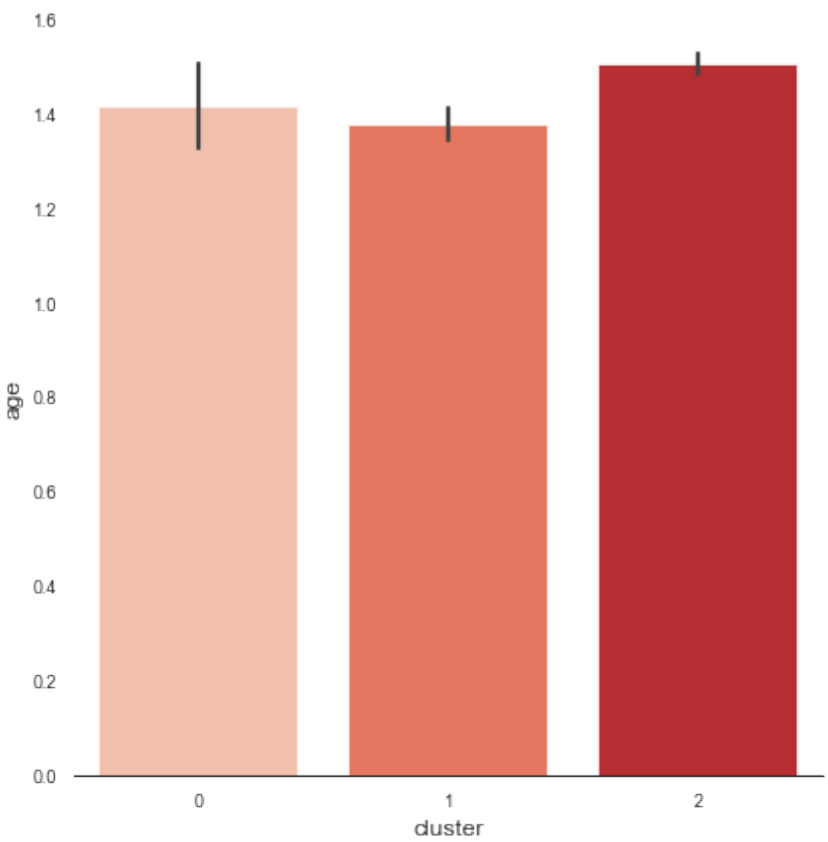
DATA VISUALIZATION



(Total spending of clusters)
0 > cluster 1 > cluster 2

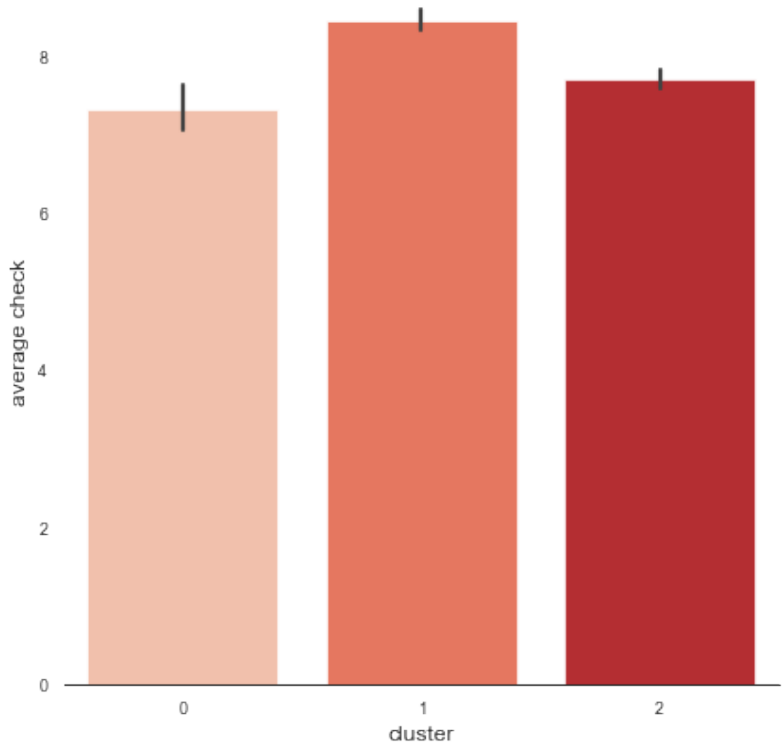
(Average days between purchases)
cluster 0 < cluster 1 < cluster 2

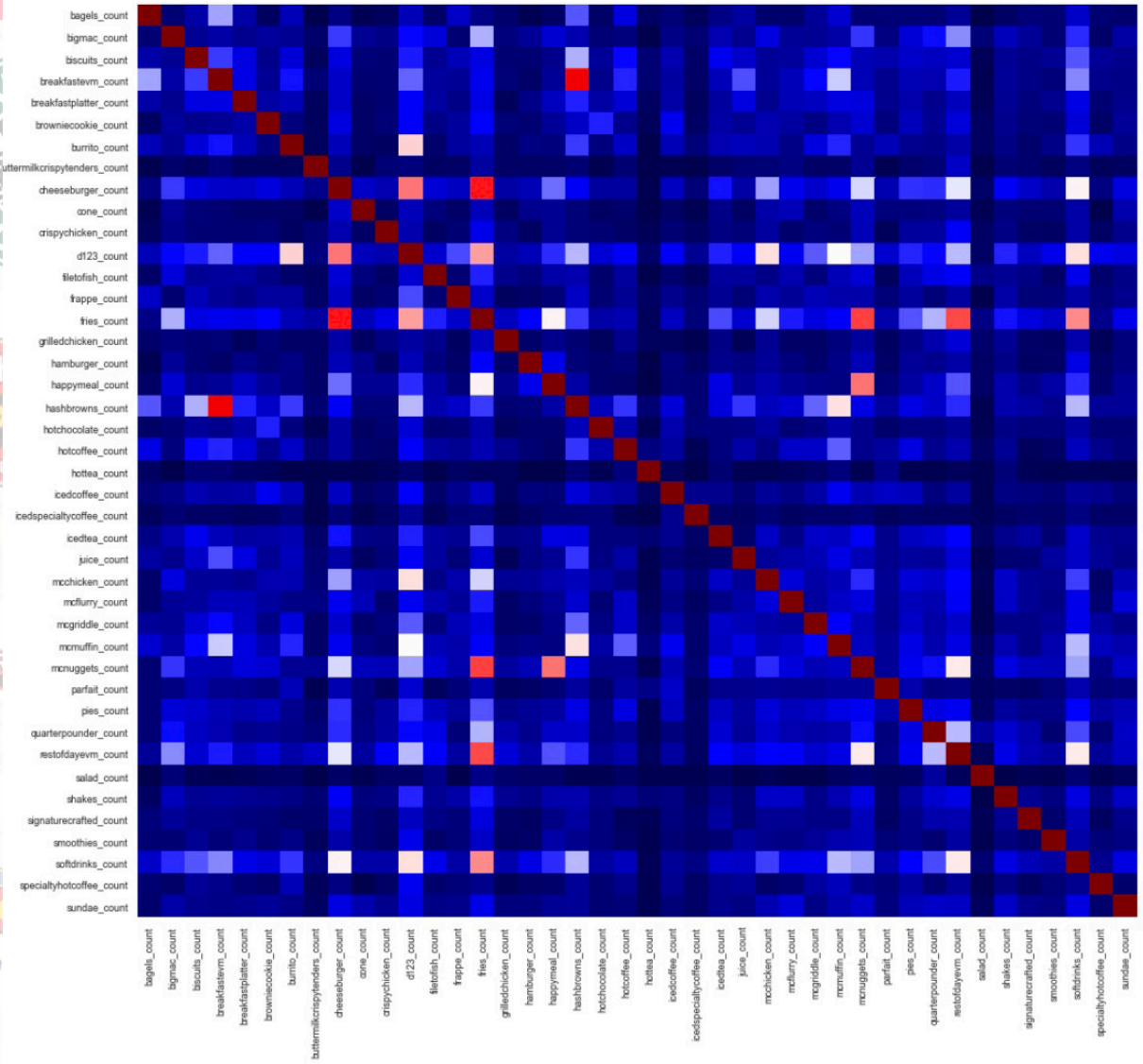




(Cluster 2 has the greatest age)

(Cluster 1 has the greatest average check)





High positive corrrelations:

hashbrowns + breakfastevm

nuggets + happymeal

fries + cheese burger / nuggets / restofdayevm / soft drinks

d123 + cheese burger / fries

CONCLUSION

CLUSTER 0 - "Young Salaryman" :

High total expense and frequency.

Median income.

Low average check and age.

(They are so busy that they only have time to order fast food, but they have to save money in each meal.)

CLUSTER 1 - “Student” :

High average check. Low age and income

(They have fun and do not need to worry about saving money, because most of them use their parents’ money.)

CLUSTER 2 - “Family”:

High income and age.

Low frequency.

(They seldom eat at fast food restaurants)

BUSINESS STRATEGY

"Young Salaryman":
(increase frequency)

- pre-ordering
- offer them annual card with discounts
- offer money saving combos (based on correlation graph)

"Student":
(promote products)

- share pictures of products on social media to get an ice cream
- write a review on social media to get 5% off
- get an extra burger for a full purchase of $12

"Family":
(increase average spending)

- encourage them to buy family combos
- get a coupon for next visit for every full purchase of $10
- add two dollars to get an ice cream for kids