UC San Diego

COGS 118B Final project K-Means, PCA

Feifan Li
Binghan Shen
Rosy Xu
Xuhui Liu
Shaolong Li

UC San Diego

Introduction

1. Dataset description
2. Topic
3. Significance of the topic

UC San Diego

Introduction:
Our Topic

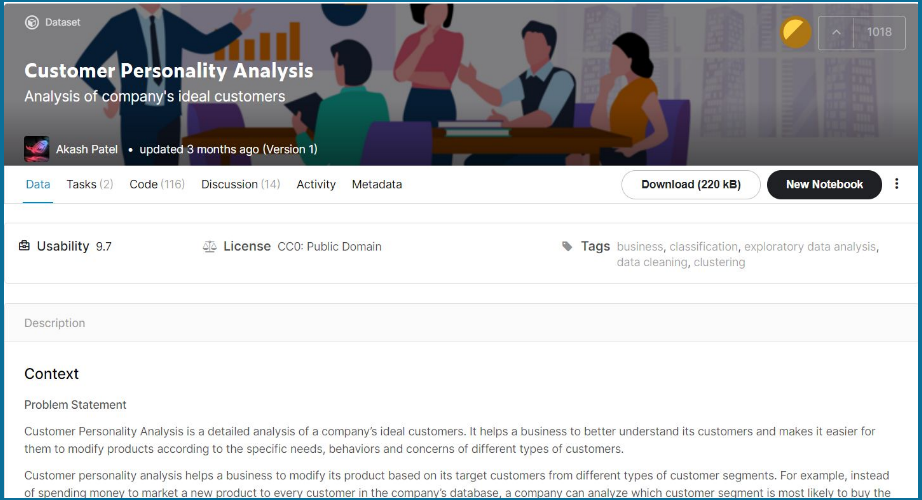
Assuming that we are a firm producing wine. Our topic is to cluster the customers by several selected features.

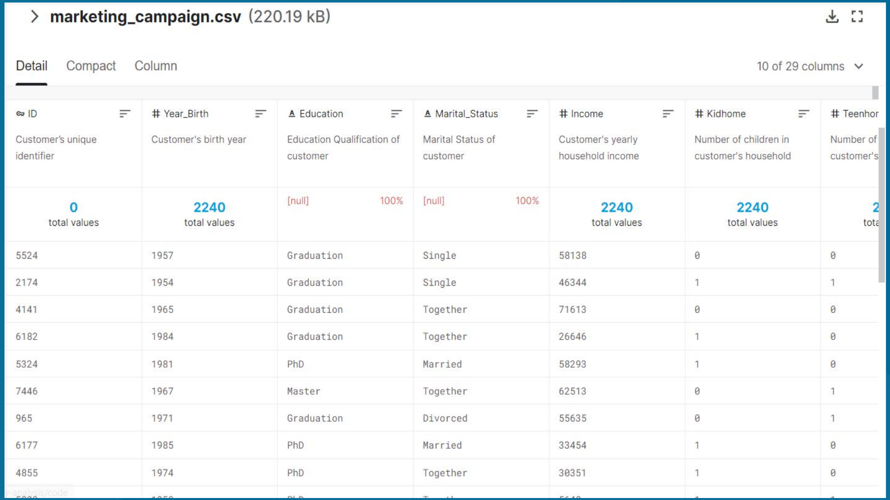
For each cluster, we adopt specific measures to figure out the customers' willingness to purchase our wine products.

Is there any other patterns worth attention?

Introduction: Dataset

Dataset we chose: https://www.kaggle.com/imakash3011/customer-personality-analysis





ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhor
Customer's unique identifier	Customer's birth year	Education Qualification of customer	Marital Status of customer	Customer's yearly household income	Number of children in customer's household	Number of customer's
0 total values	2240 total values	[null] 100%	[null] 100%	2240 total values	2240 total values	
5524	1957	Graduation	Single	58138	0	0
2174	1954	Graduation	Single	46344	1	1
4141	1965	Graduation	Together	71613	0	0
6182	1984	Graduation	Together	26646	1	0
5324	1981	PhD	Married	58293	1	0
7446	1967	Master	Together	62513	0	1
965	1971	Graduation	Divorced	55635	0	1
6177	1985	PhD	Married	33454	1	0
4855	1974	PhD	Together	30351	1	0

UC San Diego

Introduction:
Dataset

Our dataset contains 2240 rows and 29 column, after cleaning , we dropped the last 19 column since we only interested in the attributes Age, Education and Income of customers. We define each column of the dataset as one dimension. Our goal is to use K-means and PCA to build a model that classifies the customers according to the attributes listed above.

Five Dimensions are selected in the classification:

- Age: the age of the individual
- Kids: the number of kids in the individual’s family
- Teens: the number of teens in the individual’s family
- Enroll_age: how long has been the individual enrolled in our firm
- Rencency: Number of days since customer's last purchase

UC San Diego

Introduction: Significance of our project

What's the problem we want to solve?

How can we make our advertisements more targeted and effective? Nowadays, in marketing field, we are always wondering that to whom we should advertise our products.

Why is it important?

It will dramatically affect the profits that our firm could obtain. We do not want to advertise the wine product to the customers that have low willingness, which costs a lot but makes little profit in return.

UC San Diego

Introduction: Significance of our project

The central aim of our project is to cluster the customers based on some features provided and figure out their willingness to purchase wine. To be specific, we categorize the customers into several groups: low willingness, middle willingness, and high willingness. After we divide the customers according to their willingness, we can send out the corresponding advertisements. Hence, from the result of this project, the company can make their advertisements more targeted and effective.

UC San Diego

Related Works

Paper :

- *Application of K-Means Algorithm for Efficient Customer Segmentation: A Strategy for Targeted Customer Services*
- *RFM model for customer purchase behavior using K-Means algorithm*
- *Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster*

What's in common?

These paper all illustrate the common process of applying K-means for customer segmentation:

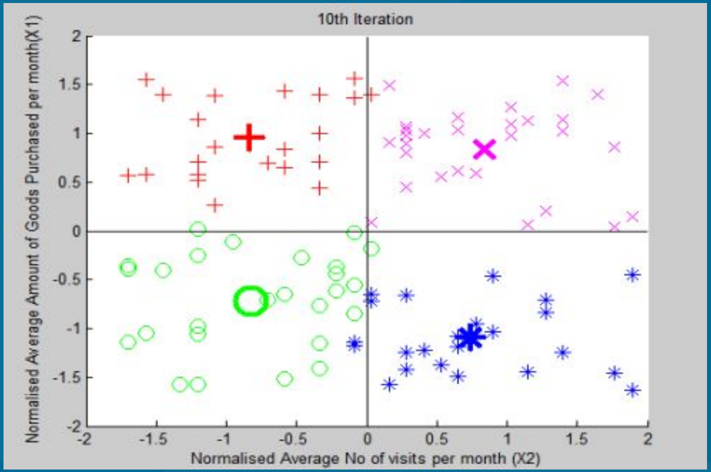
First, they determine k by using different methods. Second, they run K-means and get several clusters. Finally, they label the clusters and observe the patterns of each cluster. In conclusion, the process is similar to ours.

UC San Diego

Related Works

In the paper *Application of K-Means Algorithm for Efficient Customer Segmentation: A Strategy for Targeted Customer Services*, the authors used K-means to classify customers based on several features such as income, which is really similar to what we did in our project. However, their k means program is trained using a z score normalized two factor dataset, which we did not do for our K means. They have four clusters after running the k-means and they labeled these four clusters as High Buyers Regular Visitors (HBRV), High Buyers Irregular Visitors (HBIV), Low Buyers Regular Visitors (LBRV) and Low Buyers Irregular Visitors (LBIV). We also labeled our clusters after running the k means but our label is not as generalized as theirs. Also, they did their k means in Matlab, which is different from what we did in python. In short, both works are trying to cluster customers based on their features using k means in order to provide more precise and targeted customer service.

HBIV *Cluster +*	HBRV *Cluster* **X**
LBIV *Cluster* **O**	LBRV *Cluster **



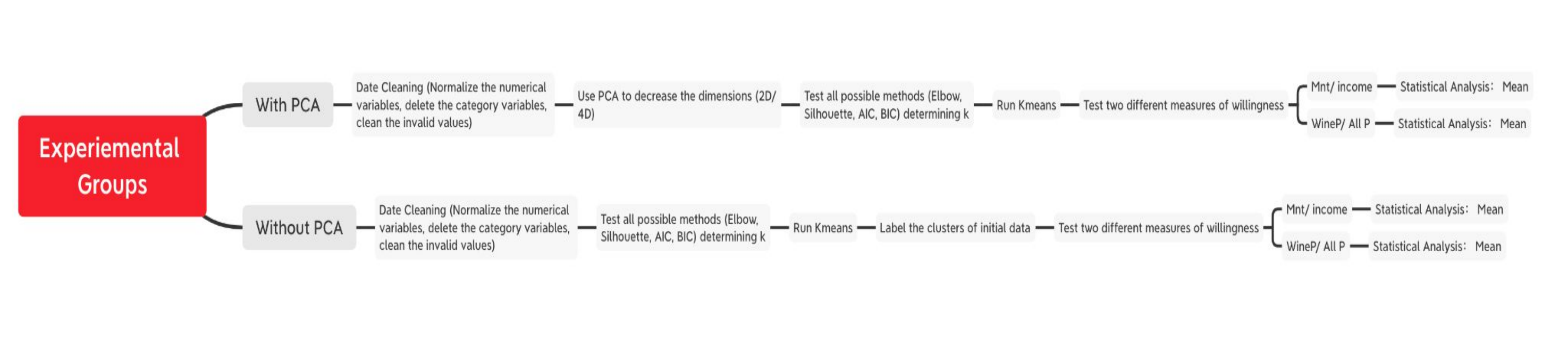
Source:
http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.736.3182

UC San Diego

Method

1. Flow chart
2. Description of the process
3. Algorithms we use(K-means and PCA)

UC San Diego


Experiemental Groups
With PCA
Date Cleaning (Normalize the numerical variables, delete the category variables, clean the invalid values)
Use PCA to decrease the dimensions (2D/ 4D)
Test all possible methods (Elbow, Silhouette, AIC, BIC) determining k
Run Kmeans
Test two different measures of willingness
Mnt/ income
Statistical Analysis: Mean
WineP/ All P
Statistical Analysis: Mean
Without PCA
Date Cleaning (Normalize the numerical variables, delete the category variables, clean the invalid values)
Test all possible methods (Elbow, Silhouette, AIC, BIC) determining k
Run Kmeans
Label the clusters of initial data
Test two different measures of willingness
Mnt/ income
Statistical Analysis: Mean
WineP/ All P
Statistical Analysis: Mean

UC San Diego

Method

How we design? -- Two critical control groups

- **Classification:** With PCA *V.S.* Without PCA
- **Measures:** Mnt/ income *V.S.* Wine P/All P

Method

How we design? -- Classification

With PCA *V.S.* Without PCA

The data set which we want to cluster is 5-D data, so we could not visualize it. In this case, determining the k would be difficult, since objective test (Elbow test, etc.) may not show an obvious result on consistency, and subjective test (directly observe the visualization) doesn't work either because we cannot visualize a 5-D data set.

Hence, to cluster the data better, besides clustering without PCA, we adopt PCA to cluster the dataset in the other control group. In this way, we can see which one is better.

Method

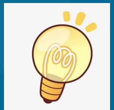
How we design? -- Measures

Mnt/ income V.S. Wine P/All P

In order to find out the purchasing willingness of each class after unsupervised clustering, we need to develop several measures to measure the purchasing willingness. Considering the features we can get in the original data set from Kaggle, we design these two measures:

- Mnt/income: the minimum expense on wine products of an individual over his/ her income.
- Wine P/All P: the expense on wine products over expense on other products (i.e. fruits, meat, fish, etc.)

After classification, we will use these two measures to compute the purchasing willingness of every individual, and observe the consistency of the two results.



Hint: Data of these two measures won't be included in the classification step!

UC San Diego

Method: algorithms

1. In this project, we mainly use two algorithms that are taught in class: K Means and Principal Component Analysis. We use K means to cluster the customers into separate categories by selected features. There are two main problems with Kmeans: first, we need to determine the appropriate value of K; a good K value should produce very consistent results. Another problem is which dimensions we should choose. Including or excluding different dimensions or features will produce different results.
2. Another algorithm that we use is Principal Component Analysis (PCA). By using it, we can reduce the dimension of our dataset, excluding the features that are least important. In the project, We run K-means on the dataset after PCA and compare its result with the one without using PCA. From this comparison, we can judge whether the use of PCA can improve effectiveness of K-means. The problem with PCA is that we need to figure out how many dimensions that we should reduce to, and this will directly affect the result of K-means.

Process and results

1. Without PCA
2. With PCA

UC San Diego

Process: Data Cleaning

Step1: drop the columns that we will not use: we only choose the first ten columns because they are mostly numerical values.

Since in the documentation, we only have explanation for the first 10 variables, we drop others

```
]: df2 = df1.iloc[:, :10]
```

Step2:we drop the rows that contain null values

Then, we deal with the missing values.

```
[6]: pd.isnull(df2).mean()
```

```
t[6]: ID                0.000000
      Year_Birth        0.000000
      Education         0.000000
      Marital_Status    0.000000
      Income            0.010714
      Kidhome           0.000000
      Teenhome          0.000000
      Dt_Customer       0.000000
      Recency           0.000000
      MntWines          0.000000
      dtype: float64
```

So, only Income contains missing values.
For simplicity, we assume it to be missing completely at random. (We didn't do hypothesis test to test this since the main task of this project is clustering not missing type analysis). So, we just drop these.

```
[7]: df3 = df2[~pd.isnull(df2["Income"])]
```

UC San Diego

Data Cleaning

Step3: Convert the year into age and standardize it

We will convert Year_Birth to age and standardize it.

```
8]: age = 2021 - df3["Year_Birth"]
    std_age = (age - np.mean(age))/np.std(age)
```

Step4: Standardize other columns: teens, kids, income, recency

We standardize income, kids and teens

```
std_income = (df3["Income"] - np.mean(df3["Income"]))/np.std(df3["Income"])
```

```
std_kid = (df3["Kidhome"] - np.mean(df3["Kidhome"]))/np.std(df3["Kidhome"])
```

```
std_teen = (df3["Teenhome"] - np.mean(df3["Teenhome"]))/np.std(df3["Teenhome"])
```

For Dt_Customer, we first find the first enrolled customer and calculate the day difference between others and the first enrolled customer. We standarize it.

```
min_date = min(df3["Dt_Customer"].apply(lambda x: pd.to_datetime(x)))
```

```
date1 = df3["Dt_Customer"].apply(lambda x: pd.to_datetime(x))
```

```
day_diff = (date1 - min_date).astype(str).apply(lambda x: x.split()[0]).astype(int)
```

```
std_day_diff = (day_diff - np.mean(day_diff))/np.std(day_diff)
```

We standardize Recency

```
std_recency = (df3["Recency"] - np.mean(df3["Recency"]))/np.std(df3["Recency"])
```

```
std_mntwines = (df3["MntWines"] - np.mean(df3["MntWines"]))/np.std(df3["MntWines"])
```

Note: We choose to standardize these dimensions because the scale of each dimension is different. Some features like income is around 50,000 but other features like kid is 0 or 1. Hence, if we do not standardize them, it will distort the dataset and influence the outcome of K-means

UC San Diego

Data Cleaning

Step 5: Use one-hot encoding to convert non-numerical value into categorical data

We will do One-hot encoding on Education

```
df3["Education"].value_counts()
```

```
Graduation    1116
PhD            481
Master         365
2n Cycle       200
Basic           54
Name: Education, dtype: int64
```

```
Graduation = (df3["Education"] == "Graduation").astype(int)
PhD = (df3["Education"] == "PhD").astype(int)
Master = (df3["Education"] == "Master").astype(int)
Cycle = (df3["Education"] == "2n Cycle").astype(int)
Basic = (df3["Education"] == "Bsic").astype(int)
```

Step6: dataset after cleaning

	Age	Income	MntWines	Kids	Teens	Enroll_age	Recency	Graduation	PhD	Master	Cycle	Basic	Married	Together	Single	Divorced
0	0.986443	0.234063	0.978226	-0.823039	-0.928972	-1.974875	0.310532	1	0	0	0	0	0	0	1	0
1	1.236801	-0.234559	-0.872024	1.039938	0.909066	1.665141	-0.380509	1	0	0	0	0	0	0	1	0
2	0.318822	0.769478	0.358511	-0.823039	-0.928972	0.172132	-0.795134	1	0	0	0	0	0	1	0	0
3	-1.266777	-1.017239	-0.872024	1.039938	-0.928972	1.923298	-0.795134	1	0	0	0	0	0	1	0	0
4	-1.016420	0.240221	-0.391671	1.039938	-0.928972	0.821827	1.554407	0	1	0	0	0	1	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2211	0.151917	0.356642	1.197646	-0.823039	0.909066	-0.124749	-0.104093	1	0	0	0	0	1	0	0	0
2212	1.904422	0.467539	0.299208	2.902916	0.909066	1.940508	0.241428	0	1	0	0	0	0	1	0	0
2213	-1.016420	0.188091	1.787710	-0.823039	-0.928972	0.847643	1.450751	1	0	0	0	0	0	0	0	1
2214	1.069896	0.675388	0.364441	-0.823039	0.909066	0.843341	-1.417072	0	0	1	0	0	0	1	0	0
2215	1.236801	0.024705	-0.655568	1.039938	0.909066	-1.161680	-0.311405	0	1	0	0	0	1	0	0	0

2216 rows × 18 columns

UC San Diego

We wrote the K-means by hands

```
In [2]: def calc_sq_dist(df, kmus):
            out_df = pd.DataFrame()
            for i in range(kmus.shape[0]):
                icol = ((df - kmus.iloc[i,:])**2).sum(axis = 1)
                col_name = str(i)
                out_df[col_name] = icol
            return out_df
```

```
In [3]: def RunKMeans(df, K, maxiters):

            ### df is cleaned data without missing value
            ### K is the number of cluster you wish to achieve
            ### maxiters is the maximum iterations you wish to do

            N = df.shape[0]     ## N is the number of observations

            rnk_df = pd.DataFrame()      ## output dataframe

            rndinds = np.random.permutation(N)
            Kmus = df.iloc[rndinds[:K],:]     ## initial K means

            for iter in range(maxiters):

                sq_dists = calc_sq_dist(df, Kmus)     ## calculate the square distance

                ranks = sq_dists.idxmin(axis = 1)     ## determine the ranks by finding the min distance
                rnk_df = df.assign(rnk = ranks)

                KmusOld = Kmus
                Kmus = rnk_df.groupby("rnk").mean()     ## update the new K means

                if sum(abs(np.array(KmusOld).flatten() - np.array(Kmus).flatten())) < 1e-6:
                    break
            return rnk_df
```

Without PCA

Now the first problem is: how do we determine k?

Because our data has more than two dimensions, it is hard to visualize them. However, there are several backup methods: Elbow, Silhouette, density graph, AIC (Akaike Information Criterion), and BIC (Bayesian Information Criterion).

There are two possible optimal k by using Silhouette and Elbow.

a. Silhouette

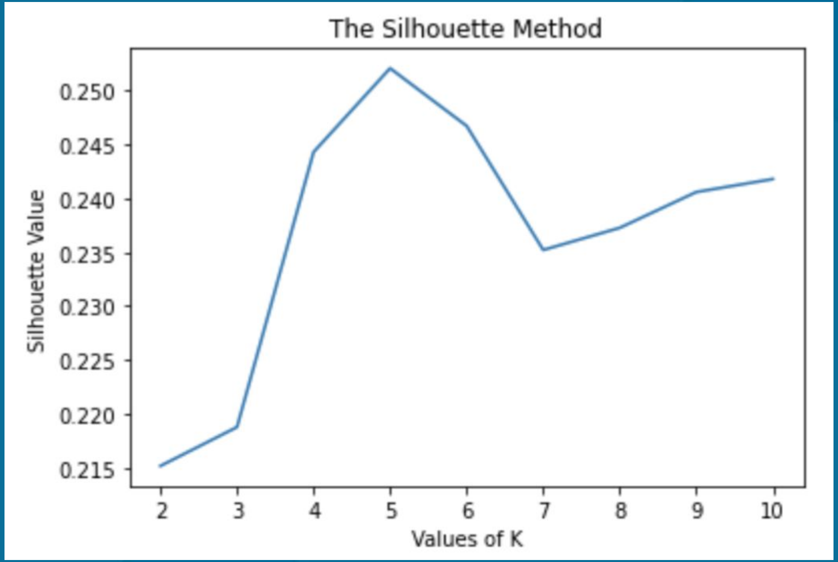
"The Silhouette value measures how similar a point is to its own cluster (cohesion) compared to other clusters (seperation)." -- Wikipedia

The higher the Silhouette value, the better is the kmean algorithm.

We used silhouette_score in python.

The Silhouette value reaches its global maximum at the optimal k.

From the graph, the optimal k is 5.



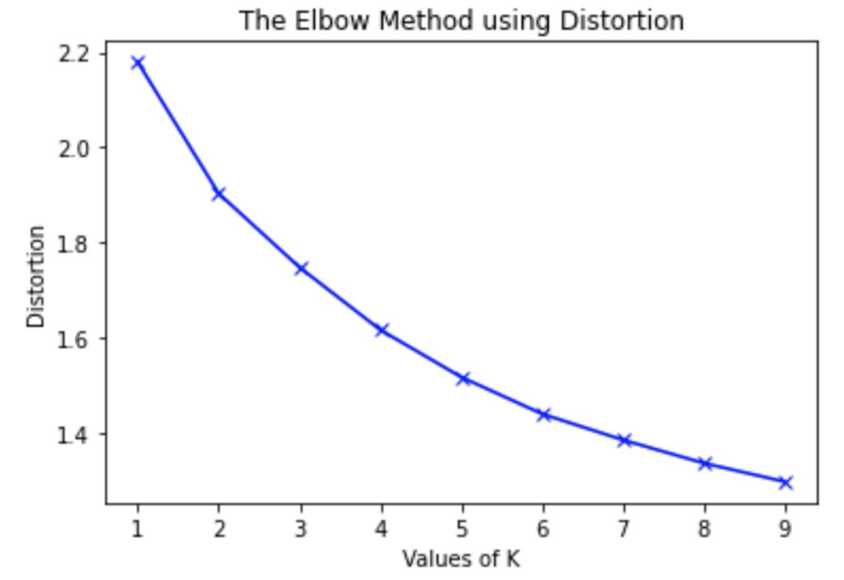
b. Elbow

The Elbow method measures the distortions (variances) of the k-mean.

We desire smaller distortions. However, as distortion is negatively related with the values of k, we should find the elbow point of the graph.

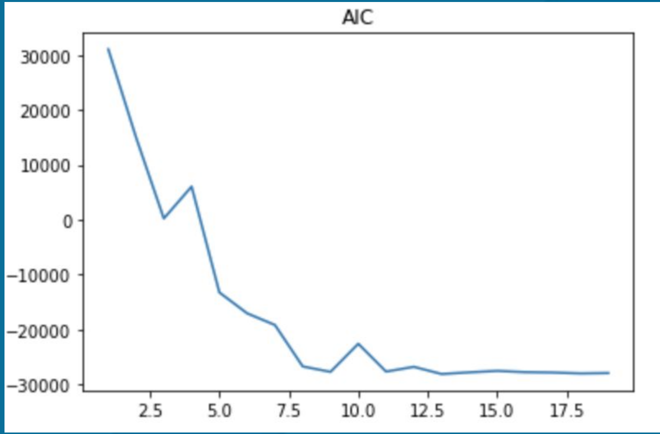
We used K-mean Model.inertia_ in python.

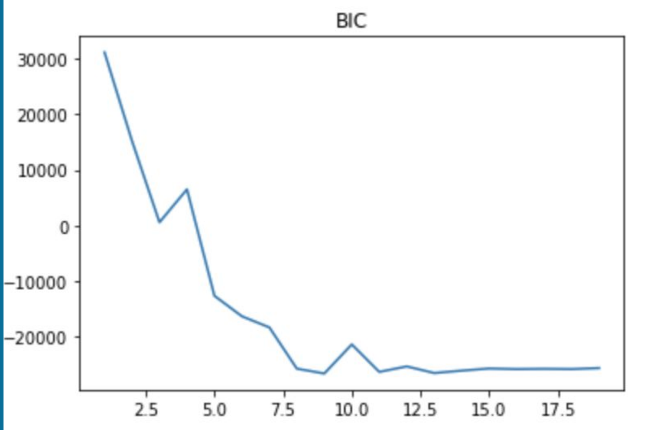
From the graph, the Elbow point is at 2. Thus, the optimal k is 2.



UC San Diego

Other Methods (Akaike Information Criterion, and Bayesian Information Criterion):





These methods produce extremely large k values, which does not meet our expectations

UC San Diego

*The data represents the standardized means of each cluster. After comparison between k=2 and k=5, 2 clusters work better for our data.

2 clusters:

Row Labels	Average of Age	Average of Kids	Average of Teens	Average of Enroll_age	Average of Recency	Average of Mnt/Income	Average of WinePrice/AllPrice
0	0.562853864	-0.218213494	0.621960226	-0.009154593	0.00842632	0.155302091	0.352911122
1	-0.807788073	0.313172334	-0.892615445	0.013138351	-0.012093159	-0.2228841	-0.506485632

5 clusters:

cluster	Age	Kids	Teens	Enroll_age	Recency	MntWines/Income	WinePrice/AllPrice
0	1.100530	-0.817144	-0.923156	0.067657	0.099283	0.003432	0.005077
1	0.296117	1.168880	1.001187	0.086490	0.014077	0.001105	0.036259
2	-0.802664	1.101674	-0.928972	0.026984	0.000372	0.038003	0.034721
3	0.405489	-0.823039	0.994029	-0.071646	-0.006825	-0.031031	-0.044529
4	-1.013280	-0.823039	-0.928972	-0.078256	-0.103551	-0.007063	-0.015544

There is little differences between these two clusters.

UC San Diego

Overview of K-mean with 2 clusters

Row Labels	Average of Mnt/Income	Average of WinePrice/AllPrice	Count of cluster
0	0.155302091	0.352911122	1306
1	-0.2228841	-0.506485632	910
Grand Total	**-7.15633E-16**	**-5.48238E-15**	**2216**

- The two clusters separated by Kmeans show that cluster 0 tends to spend large proportion of their money on wines, and cluster 1 tends to spend smaller proportion of their money on wines.

UC San Diego

Row Labels	Average of Widow	Average of Divorced	Average of Single	Average of Together	Average of Married
0	0.053598775	0.118683002	0.17611026	0.267228178	0.380551302
1	0.006593407	0.084615385	0.264835165	0.246153846	0.395604396
Grand Total	**0.034296029**	**0.104693141**	**0.212545126**	**0.258574007**	**0.386732852**

- Cluster 0 is composed of 5.4% of widow, 11.9% of divorced, 17.6% of single, 26.7% of together and 38.1% of married.
- Cluster 1 is composed of 0.6% of widow, 8.7% of divorced, 26.5% of single, 24.6% of together and 39.6% of married

UC San Diego

Row Labels	Average of Graduation	Average of PhD	Average of Master	Average of Cycle
0	0.491577335	0.246554364	0.179173047	0.074272588
1	0.520879121	0.174725275	0.143956044	0.113186813
Grand Total	**0.503610108**	**0.217057762**	**0.164711191**	**0.090252708**

- Cluster 0 is composed of 49.2% of graduation, 24.7% of PhD, 17.9% of master and 7.4% of cycle.
- Cluster 1 is composed of 52.1% of graduation, 17.5% of PhD, 14.4% of master and 11.3% of cycle.

UC San Diego

Result Analysis

Cluster	age on avg	avg recency	avg Income
0	58.92	49.26	55455.10491
1	42.50000001	48.66	47643.45276

- We first look at the average of age, recency, and income for each clusters.
- People who have higher income and larger age is likely to be in cluster 0.

UC San Diego

Result Analysis

	Widow	Divorced	Single	Together	Married
Cluster 0	92%	67%	49%	61%	58%
Cluster 1	8%	33%	51%	39%	42%

- We calculate the proportion of being cluster 0 and cluster 1 for each relationship state respectively.
- If our future customer is widow, we have 92% possibility that he/she is from cluster 0. In other words, he or she has 92% possible to spend more proportion of their income on purchasing wines.
- Similarly, it is more likely for a divorced, together or married customer to spend more income on wine, while it is less likely for a single customer to spend more income on wine.

UC San Diego

Result Analysis

	Master	PhD	Graduation	Cycle
Cluster 0	64%	67%	58%	49%
Cluster 1	36%	33%	42%	52%

- We calculate the proportion of being cluster 0 and cluster 1 for each diploma respectively.
- If our future customer is PhD, we have 67% possibility that he/she is from cluster 0. In other words, he or she has 67% possible to spend more proportion of their income on purchasing wines.
- Similarly, it is more likely for a master or graduation customer to spend more income on wine, while it is less likely for a cycle customer to spend more income on wine.

UC San Diego

Result Analysis

Cluster	#Kids on avg	#Teens on avg
0	0.325	0.844
1	0.61	0.02

- We calculate the average number of kids or teens for cluster 0 and cluster 1.
- If our future customer does not have kids, it is more likely that they will spend more proportion on wines. If our future customer have kids, it is more likely that they will spend less proportion on wines.
- On the opposite, If our future customer does not have teenagers, it is more likely that they will spend less proportion on wines. If out future customer have teenagers, it is more likely that they will spend more proportion on wines.

UC San Diego

Result Analysis

- Beside of looking at the characteristics of people in each cluster, we could also look at our two measures $\frac{\text{Amount spent on wines}}{\text{Income}}$ and $\frac{\text{Amount spent on wines}}{\text{Amount spent on all products}}$

Row Labels	Average of WineProducts/AllProducts	Average of Mnt/Income
0	0.352911122	0.155302091
1	-0.506485632	-0.2228841

- From the table, since $\frac{\text{Amount spent on wines}}{\text{Amount spent on all products}}$ has a larger range, which means that using this index is more distinguishable compared to $\frac{\text{Amount spent on wines}}{\text{Income}}$

UC San Diego

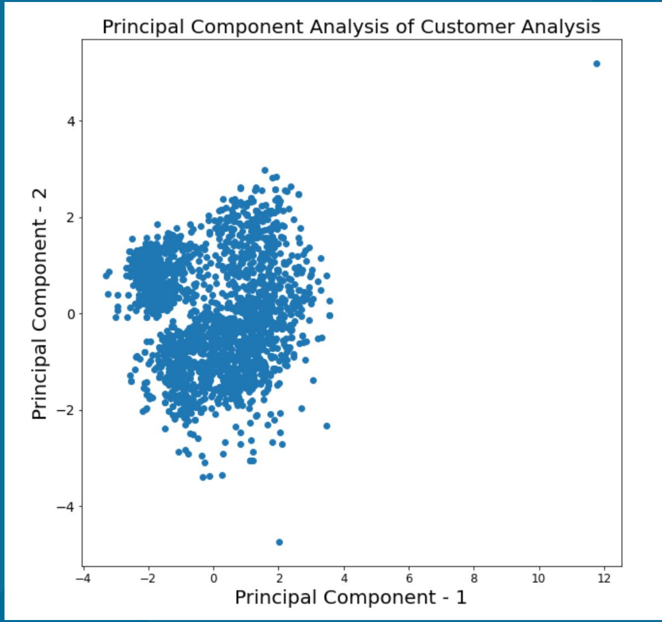
Summary

- Characteristics of People who are likely to have higher desire on purchasing wines:
  - People who are older around 59 years old
  - People who are richer around $55455 income
  - People who are widow
  - People who is PHD
  - People who has one teenager
- We could also use $\frac{\text{Amount spent on wines}}{\text{Amount spent on all products}}$ to distinguish the clusters of each customer.

With PCA

1. Our first step is to decrease the dimensionality of the dataset. We reduce the number of principal components to 2, so we can visualize the dataset.

From the graph here, we can see that the general distribution of the dataset, and it approximately has three main clusters. Hence, our optimal k value could be 2 or 3.



UC San Diego

2. We run the PCA on all the components and see its variance ratio.

```
# run PCA to get all components
pca2 = PCA(n_components=7)
pca2.fit(df10)
```

```
PCA(n_components=7)
```

```
##print out the varaiance ratio
pca2.explained_variance_ratio_
```

```
array([0.30335721, 0.18368963, 0.14854715, 0.1386951 , 0.09117064,
       0.0789861 , 0.05555415])
```

From the ratio, we can see that the first four columns have relatively large variance, while the last three are quite small. In this case, we decide to set the number of our principal components equal to 4.

3. Run the PCA and get the result

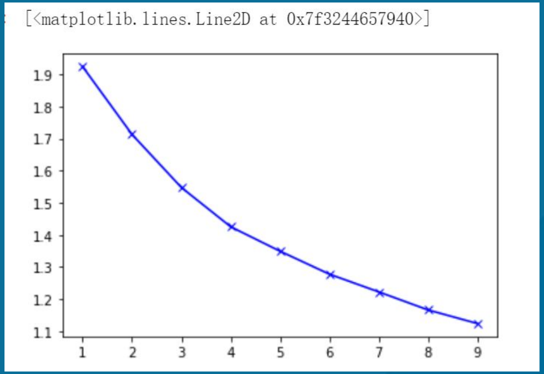
	principal component1	principal component2	principal component3	principal component4
0	1.009155	0.599199	-1.484474	0.861259
1	-0.566072	-1.731228	1.329067	-0.586936
2	0.690986	0.675897	0.867258	0.266582
3	-1.553015	0.641515	1.804550	-0.611720
4	-0.771120	0.845144	-0.319718	-1.893810
...	...	...	...	...
2211	1.009677	-0.287158	-0.082824	0.255029
2212	-0.400435	-1.950033	1.081970	-1.303376
2213	0.725388	1.401870	-0.265498	-1.797742
2214	0.929499	-0.970743	1.559623	0.607947
2215	-0.216558	-1.273067	-0.723354	1.107095

2216 rows × 4 columns

And we will use this dataset to run K-means

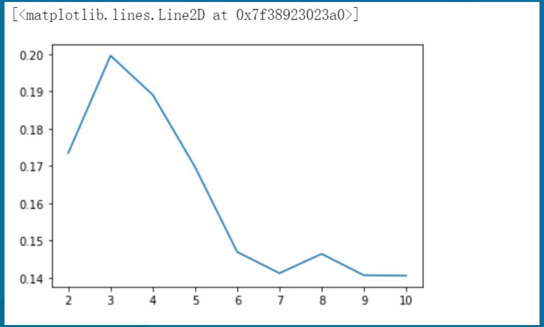
UC San Diego

4. Use elbow graph to determine k value



From the graph, we can see that the elbow point is not very clear, so we choose silhouette score to determine the k value

5. Use Silhouette score to determine k value



From this graph, the silhouette score reach the maximum at k=3. Hence, our optimal k value should be 3, which corresponds to our prediction before

UC San Diego

6. Run K-means on the dataset after PCA

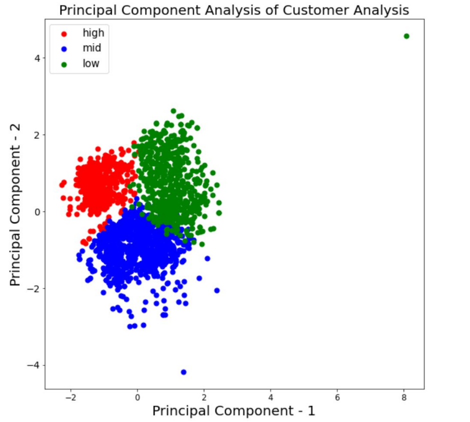
	Age	Kids	Teens	Enroll_age	Recency	MntWines/Income	MntWine/All_Products	cluster2
0	0.986443	-0.823039	-0.928972	-1.974875	0.310532	1.327251	-0.289712	0
1	1.236801	1.039938	0.909066	1.665141	-0.380509	-0.965033	-0.225226	1
2	0.318822	-0.823039	-0.928972	0.172132	-0.795134	0.260233	0.395571	0
3	-1.266777	1.039938	-0.928972	1.923298	-0.795134	-0.927390	-1.101683	2
4	-1.016420	1.039938	-0.928972	0.821827	1.554407	-0.379266	-0.214065	2
...	...	...	...	...	...	...	...	...
2211	0.151917	-0.823039	0.909066	-0.124749	-0.104093	0.244655	-0.609511	0
2212	1.904422	2.902916	0.909066	1.940508	0.241428	-0.886288	0.229550	1
2213	-1.016420	-0.823039	-0.928972	0.847643	1.450751	0.564772	-0.196591	0
2214	1.069896	-0.823039	0.909066	0.843341	-1.417072	-0.530673	-0.367344	1
2215	1.236801	1.039938	0.909066	-1.161680	-0.311405	-0.992639	-0.915513	1

2216 rows × 8 columns

Note: Cluster2 indicates which cluster the customer belongs to. And the measures (6th and 7th columns) did not run through the Kmeans. they were appended later.

7. Visualize the clustering

```
Out[79]: <matplotlib.legend.Legend at 0x7f32106742e0>
<Figure size 432x288 with 0 Axes>
```



From the graph, we can see that the outcome of K-means is pretty good

UC San Diego

7. take a look at the mean

cluster2	Age	Kids	Teens	Enroll_age	Recency	MntWines/Income	MntWine/All_Products
0	0.010366	-0.754778	-0.494017	-0.067100	0.069723	-0.037571	-0.033949
1	0.525092	-0.000908	0.895579	0.045051	-0.037379	0.009818	0.000581
2	-0.836453	0.815890	-0.874467	0.001601	-0.016489	0.025296	0.035721

UC San Diego

Result Analysis:

1. From the two measure columns (MntWine/Income, MntWine/All) in the statistics above, we can see that
   Cluster0 = low willingness
   Cluster1 = middle willingness
   Cluster2 = high willingness

2. Moreover, we can convert the normalized data back to the original to see the patterns. Specifically, the group of customers with low willingness has a average age of 52, and most of them do not have kids. This clustering makes sense because most elder people did not drink much wine, less likely to purchase wine. However, for the customers with high willingness, they have average of 40, and most of them have kids. From the comparison, we may conclude that the younger customers have higher willingness of buying wine and they usually have kids. Maybe number of kids can indicate the richness of the family. Therefore, if we want to sell wine products, we should consider sending advertisements to younger customers who have kids.

3. Another pattern we can observe from the statistics is that compared with the customers with high willingness, customers with low willingness have small enrollment age and long recency. Low enrollment age indicates that they are new customers and long recency implies that their shopping frequency is low. Hence, when we sell our wine products, these customers should not be our primary target.

4. By comparing the two measure columns (MntWine/Income, WineProducts/AllProducts), we can discover that the second measure (WineProducts/AllProducts) has more differences between clusters. This means that the second measure can represent customers' purchasing willingness in a more distinguishable way, which makes it a better measure compared with the first one (MntWine/Income).

5. Compared with the result from without PCA and with PCA, their patterns have some differences. For example, for without PCA, customers with high willingness of purchasing wine usually have larger age around 59. However, for with PCA, the customers with high desires of purchasing wine have younger age around 40. Thus, some of the patterns are similar, and others are different. We believe that this is because PCA exclude some dimensions, which influences the outcome of clustering. In conclusion, using PCA before K-means is not very appropriate, which will affect the clustering.

Discussion

What did we learn?

1. The whole process and the difficulties we may have of K-mean algorithm
2. The problems may exist in unsupervised learning.
3. Lessons from PCA

What could we do better?

1. Measures selection
2. Processing categorical data
3. Supervised learning
4. Choose dimensions

UC San Diego

What did we learn?

1. The whole process and the difficulties we may have of K-mean algorithm

In this project, we get a chance to go through the whole process of K-mean algorithm. First, we should properly select the data and we know from this project that the binary variable (0-1) cannot be taken into K-mean algorithm. After that, we should determine the proper k and we may have difficulties in this part because Elbow taught in the lecture may not always work. To overcome the difficulties, we learn that PCA or other k-determining methods (such as AIC, BIC, Silhouette, etc.) would work. Finally, because K-mean algorithm is an unsupervised learning algorithm, we should figure out the patterns of each cluster.

UC San Diego

What did we learn?

2. The problems may exist in unsupervised learning

The algorithm we used in this project are all unsupervised learning algorithms. Unlike supervised learning, we cannot collect or produce data from the previous experience. One thing we learned from the this project is that most of the time unsupervised learning don’t classify the data into clusters that we familiar with, we have to look carefully into each cluster to find the underlying pattern the algorithms have found out. The first few times of clustering may be unintuitive but all we need to do is to try different value for k until we find a significant difference.

UC San Diego

What did we learn?

3.Lessons from PCA

In this project, we use PCA to reduce the dimension of the original dataset and run K-means on that. But the result does not meet our expectation. PCA has some influences on the outcome of clustering. As professor mention in the class, PCA will exclude several dimensions and rotate the dataset to reduce the dimension. And this could be regarded as the primary reason why the clustering result is not good. However, PCA proves to be very useful in visualizing high dimensional data. As you can see in the previous slides, PCA allows us to visualize the clustering result of K-means, which is clear and easy to comprehend. In conclusion, we should be more careful when we use PCA to reduce the dimensionality of the dataset. Even though it contains the maximum variance, it still modify the dataset to some extent.

UC San Diego

What did we learn?

4. Multiple ways to choose optimal k

In this project, we use Elbow, Silhouette, AIC(Akaike Information Criterion), and BIC(Bayesian Information Criterion) respectively to find the optimal k. Though there are different methods, each method indicates different optimal k. Thus, we need to try each method and find which optimal k is truly optimal for our dataset.

UC San Diego

What could we do better?

1. Measure Selection

In this project, we only use two measures to represent customer’s willingness to purchase wine: MntWine/Income and WineProducts/AllProducts. In our project plan, we expect to use about 4 or 5 measures to represent the purchasing willingness and compare their outcomes to see which one is the most representative. Hence, if we are given with more time, we should come up with more measures to make the project less biased and more convincing. Moreover, we can invite the students who have deep understanding in statistics, and he can give more valuable and reasonable measures.

What could we do better?

2. Processing categorical data

We used one-hot coding to encode the categorical data. For example, married will be 1 and not married will be 0. However, encoding in this way has a problem: we cannot run K-means on categorical variables because computing Euclidean distance for such data is meaningless. to solve this problem, we may need to adjust our K-means algorithm. After some research, we found that an algorithm called K-mode may help us. Hence, if we have more time, we would like to use this algorithm to process categorical data and observe its effect.

UC San Diego

What could we do better?

3. Supervised learning

Another improvement that we can do is supervised learning. Classification by using unsupervised learning may not always classify the data in the way we want. In this case, we want to find purchasing willingnesses of each group are distinguished from each other. But the unsupervised learning cannot always satisfy our requirement.

Hence, what more we can do is to split our dataset into training part and testing part. We run the K-means on the training part to cluster the customers and summarize its patterns; then, we apply these patterns to the test part to predict which cluster they belong to and we can see if they really belong to this cluster. In this way, we can judge the accuracy and effectiveness of the Kmeans, which will make our project more meaningful.

UC San Diego

What could we do better?

4. Choosing Dimensions

Another improvement that we can make is to try different combinations of dimensions. In this dataset, there are 7 dimensions that we can use. thus, when we run the Kmeans, if we include or exclude several dimensions, the outcome of the Kmeans will also change. We should try different combinations of the dimensions to see which one will make a better clustering. this part is very time consuming because we may need to try about 50 kinds of combination and see their outcomes.

UC San Diego