Customer Segmentation

Mingrui Zhang
zhang.mingr@northeastern.edu
Northeastern University
Boston, MA, USA

ACM Reference Format:
Mingrui Zhang. 2018. Customer Segmentation. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX).* ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Customer segmentation is a way to divide customers into groups based on common characteristics so that companies can make market decisions and develop specialized marketing strategies to different groups. For example, they can create and communicate targeted marketing messages that will resonate with specific groups of customers. In this project, we have a dataset consisting of customer information. We need to use unsupervised machine learning technologies to cluster customers in an appropriate way and try to interpret the clusterings and find out some valuable insights to help companies know more about their customers and make better marketing strategies. In this project, we will implement PCA and autoencoder to customer dataset to reduce dimensions and use Kmeans method and the Elbow method to find the optimal k number of clusters. Then, Silhouette score and DB index are metrics to evaluate quality of clustering. Finally we conclude customer characteristics for different clusterings to achieve customer segmentation.

The project has been done by Mingrui Zhang who is responsible to all the parts of researching, coding and reporting.

2 Related Work

Companies using customer segmentation technique because each client is distinct and their marketing efforts would be better by building an efficient marketing and business strategies. Companies also hope to obtain a deeper understanding of the preferences and needs of their customers wuth the concept of finding out what each segment finds most useful to tailor marketing products to that segment more correctly[1]. A customer segmentation by Karnika Kapoor has used a simple PCA to reduce dimension of dataset then use Kmeans for clustering. But it has limitations that it doesn't consider to find a optimal number of component for PCA and a better way rather than PCA to do dimensionality reduction.

3 Background Information and Proposed Approach

This section mainly discusses the approaches that we use to implement customer segmentation.

3.1 Principal component analysis

Principal component analysis(PCA) is a method for dimensionality reduction of large dimension datasets while minimizing information loss at the same time. It can project data points onto only the first few principal components to obtain lower dimensional data and preserve as much data's variance as possible at the same time. The first principal component can equivalently be defined as a direction that maximizes the variance of the projected data.

3.2 Autoencoder

Autoencoder is an unsupervised deep learning neural network that has a symmetric structure with encoder and decoder layers. It compresses data into low dimension in the encoder layer and try to reconstruct the original data in decoder layer with the minimal reconstruct error.

3.3 Kmeans

Kmeans is a clustering method. It firstly select k centroids and repeatly classify each data point to its nearest centroids type and then get new centroids from the average of data points among the same type until it converges.

4 Experiments

The experiments are run under google colab, a platform similar to jupyter notebook that anyone can write and execute arbitrary python code through the browser.

4.1 Datasets

The dataset is from kaggle. The URL is https://www.kaggle.com/imakash3011/customer-personality-analysis. It has 2240 customer records for us to analyze and each customer has 29 features within four main parts: customer's basic information, the total quantity a customer spent in six kinds of

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

products, promotion that customer ever used and the number of times customer bought products in four purchasing ways such as online, in store etc.

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer
0	5524	1957	Graduation	Single	58138.0	0	0	04-09-2012
1	2174	1954	Graduation	Single	46344.0	1	1	08-03-2014
2	4141	1965	Graduation	Together	71613.0	0	0	21-08-2013
3	6182	1984	Graduation	Together	26646.0	1	0	10-02-2014
4	5324	1981	PhD	Married	58293.0	1	0	19-01-2014

Figure 1. Sample of part dataset

Dataset is processed before putting into next steps with the following parts:

- `Data Cleaning`: Data is cleaned after dropping samples with missing values and outliers. We also drop some clearly redundant features.
- `Feature Engineering`: Ordinal features such as *Education* represented as education level transform into real values. New feature *ChildNum* is extracted from feature *Teenhome* and *Kidhome*. Date-time feature *Dt_Customer* which shows the date of customer's enrollment with the company extracts a new feature *Customerdays* to reveal the number of days customer enrolled with the company.
- `Data Transformation`: Standardization is an important step of data preprocessing, especially PCA will be applied to the dataset. In our case, StandardScaler is used to normalized the dataset. StandardScaler is typically done via following formula:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where μ is the mean value and σ is the standard deviation.

After data processing, the dataset now has 18 features and 2215 samples. To find the correlation between each features, we also draw a heatmap as Figure 2. From the heatmap below, we find that *Income* has a negative correlation with *NumWebVisitMonth* and *ChildNum* and a positive correlation with all the features about the number of purchasing especially *MntWines* and *MntMeatProducts*. It means that people with higher income tend to have less children, buy more products like wines and meat and seldom go shopping online.

4.2 Principal component analysis

At first, we apply PCA to reduce dimensions. We have two methods to determine the number of principal components: Scree Plot and Kaiser's rule(Figure 3). From Figure 3, the clear elbow for variance drop is around PC4 to PC5. According to Kaiser's rule to only retain factors whose eigenvalues are greater than 1, keeping top 5 components seems to be a reasonable choice.

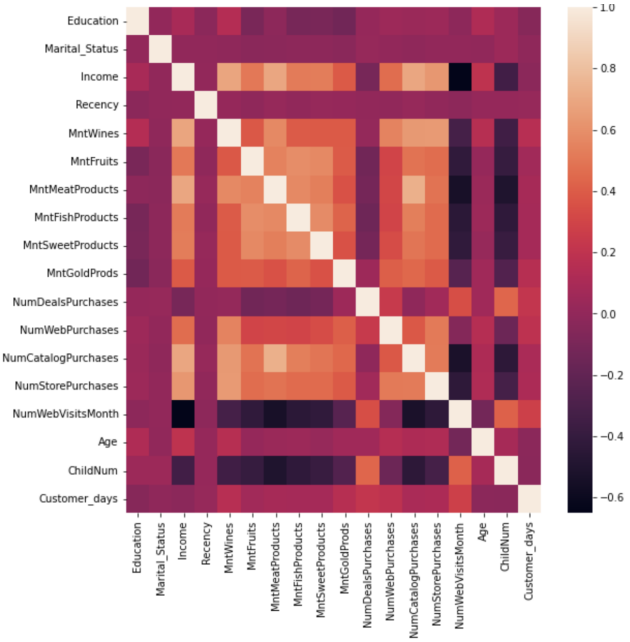

Figure 2. Heatmap of features in preprocessed dataset

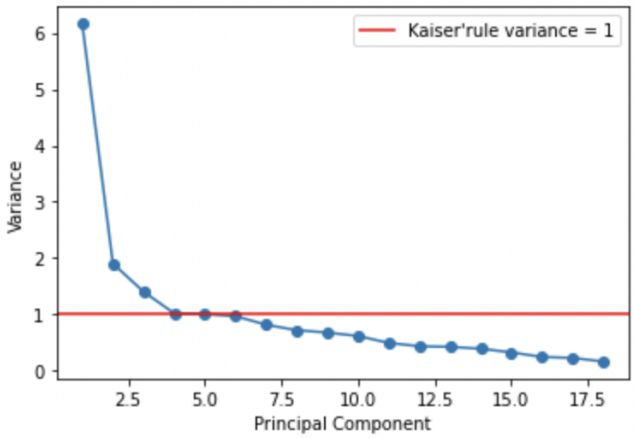

Figure 3. Scree Plot

We also draw the relationship with cumulative proportion of variance with principal components(Figure 4). The first component contain 35% of overall variance and top 5 components can retain around 65% information from original dataset.

Then we try to find important features that influence the Principle components and have a high absolute eigenvalue score. For those features that contribute very trivial to to eigenvectors should be considered to drop. From table 1, we can see that income is the most important feature for the first principal component.

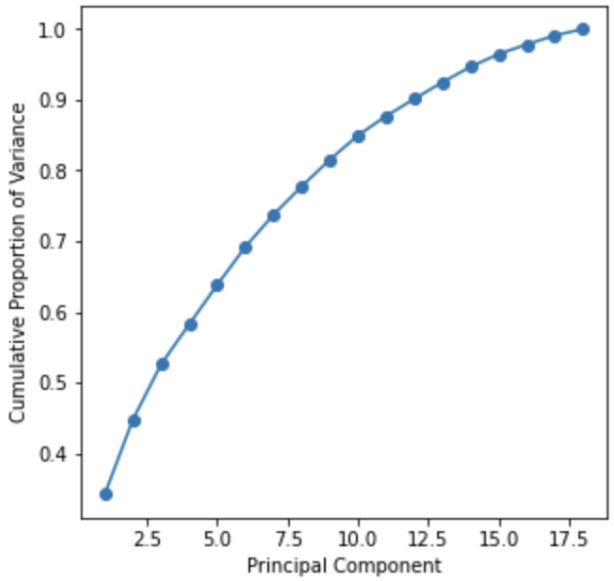

Figure 4. Distribution between cumulative proportion of variance and principal components

Table 1. Top 2 features contributions to Top 5 principal components

PC	Top1 feature	Top2 feature
1	Income	NumCatalogPurchase
2	NumDealsPurchases	NumWebPurchases
3	Education	Age
4	Recency	MaritalStatus
5	MaritalStatus	Recency

4.3 Autoencoder

PCA is a linear dimensionality reduction method while autoencoder can have non-linear transformation to matrix. Therefore, in our project, autoencoder is implemented to compare the quality of clustering with PCA.

We construct fully connected autoencoders using the Keras tensorflow and compile the neural network with mean square error loss and Adam optimizer with the default Keras parameters. we also set hyperparameters epochs=500, batch_size=256 to train autoencoders. To find the optimal encoder to compare with PCA, we build three autoencoders with layer [10,3,10],[10,4,10],[10,5,10] and analyze the trend of reconstruct error(in this case mean square error loss) during training process.Below is the table that shows the results(Table 2). Nodes means the number of nodes in hidden layer. The number in slow stage shows the approximate MSE dropped slowly during training. After considering the training time and MSE model get, an antoencoders with 5 hidden layer may be suitable to compare with reduced PCA matrix.

Table 2. MSE for autoencoders in different stages

Nodes	Start stage	Slow stage	Final stage
3	1.2446	0.68	0.6475
4	1.2518	0.68	0.6616
5	1.2422	0.65	0.6475
6	1.2475	0.63	0.6271

4.4 Kmeans Clustering

To investigate how many clusters are suitable for both reduced PCA matrix and encoder from autoencoder, we use Kmeans method to find the distribution between distortion(Sum of squared distances) and number of clusters and The Elbow method to find the optimal k.

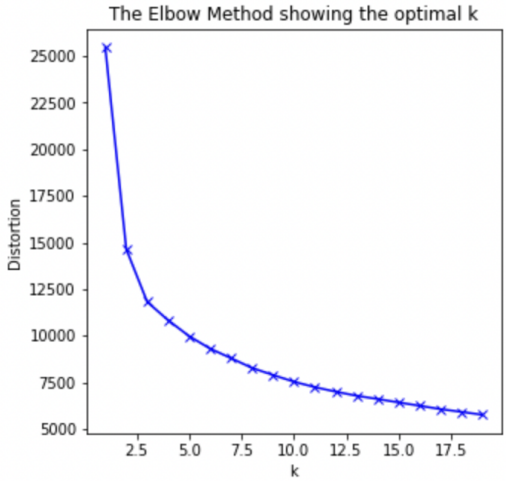

Figure 5. The Elbow method for PCA

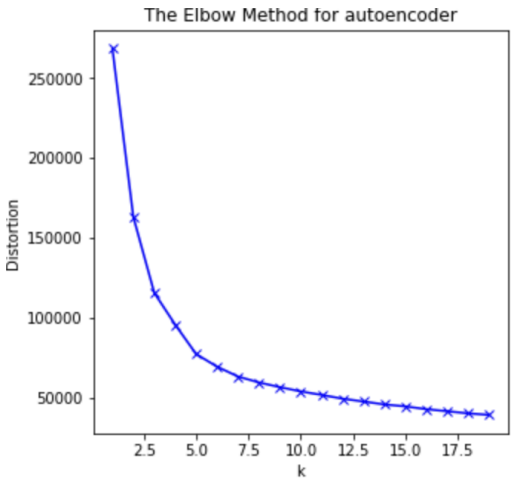

Figure 6. The Elbow method for autoencoder

From Figure 5 and 6, we find that distortion of PCA is around only 10% of autoencoder's. It shows that the samples reduced by autoencoder into 5-dimensions are more sparse and far away from each other. For both PCA and autoencoder, the optimal clustering number is 5.

4.5 Evaluation

We use two evaluation metrics to evaluate how good clustering is. The first one is Silhouette. It measures how well the features are clustered and it measures the similarity of an object to its own cluster compared with other clusters. The range of silhouette is between -1 and +1. The higher silhouette value is, the object is better matched to its own cluster and more poorly matched to neighboring clusters. We will use average silhouette value to estimate the overall performance of clustering. The second one is Davies-Bouldin(DB) index. This index produces an average measure of similarity of each cluster with its most similar cluster. The lower the value of the DB index, the better the quality of clustering is.

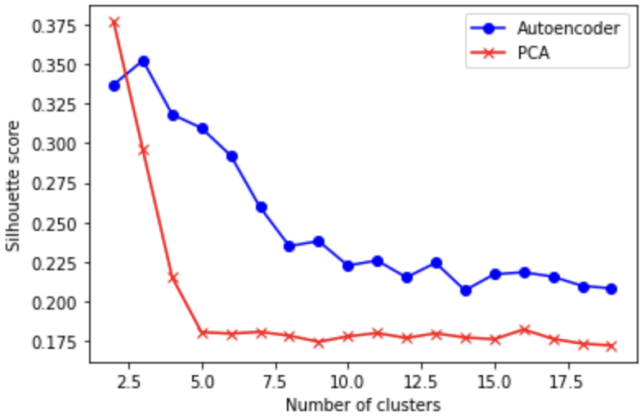

Figure 7. Distribution between number of clusters and Silhouette score

From figure 7 and 8, we can see that at most of times, Silhouette score of autoencoder is larger than that of PCA, and they both have a decreasing trend as number of clusters increases and are in range of [0.175,0.375] which is a relatively good score. In addition, DB index of autoencoder is smaller than that of PCA. Therefore, we can conclude that in our project, dimensionality reduction from autoencoder is better than that of PCA. But PCA still performs well as the scores are very close to the autoencoder's.

4.6 Customer Segmentation

Now, we use 5 dimensions reduced matrix of autoencoder and kmeans algorithm to classify 4 cluster(4 is a suitable number with lowest DB index).Here are some distributions of four clusters with original features. From those figures, customer's characteristics can be described clearly.

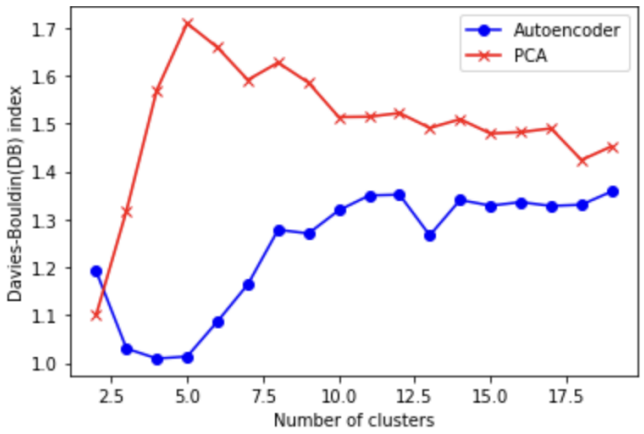

Figure 8. Distribution between number of clusters and Davies-Bouldin(DB) index

Cluster 0: most only have one child, basic education degree, not usually use coupons, have middle-upper level income, love spending money in meat and gold products.

Cluster 1: most don't have child, have a high education degree, seldom use coupons, have a high level income, love spending money on meat but not gold products.

Cluster 2: have multiple children, basic education degree, usually use coupons, low level income, spend little money on meat but love buying gold products.

Cluster 3: have multiple children, middle education degree, often use coupons, very low level income, spend little money on meat and gold products.

Due to the limit of pages, I only select some distinctive distributions. We can conclude a clearer description if we look into the whole features.

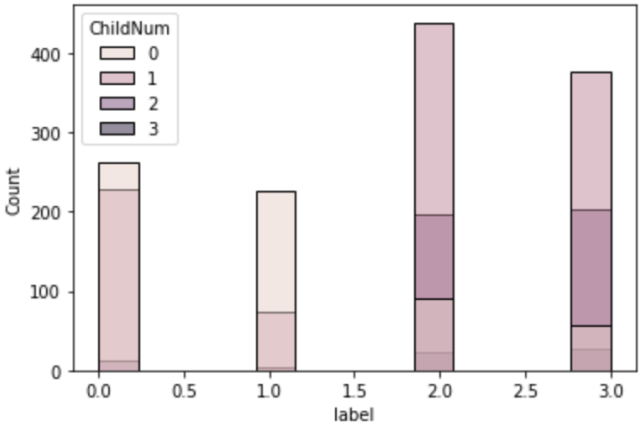

Figure 9. Distribution between clusters and the number of children they have

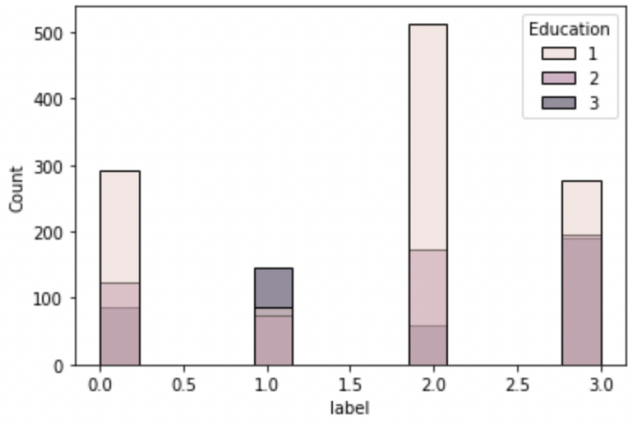

Figure 10. Distribution between clusters and education(1:Basic 2:Master 3:PHD)

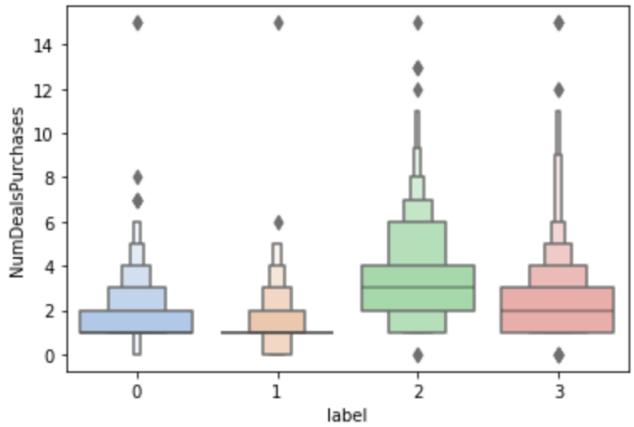

Figure 11. Distribution between clusters and amount of coupons used

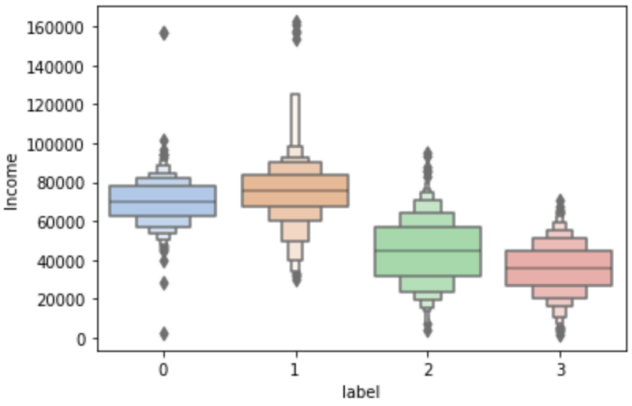

Figure 12. Distribution between clusters and Income

5 Conclusion and Future Work

In this project, we use two dimensionality reduction methods: PCA and autoencoder applied into a dataset containing customer information. For the dataset itself, we can find that income has negative correlation with number of children. For PCA, there are five principal components that have variance greater than 1 and represent 65% of overall original dataset. Feature *Income* contributes the most to Top 1 component. For autoencoder, we experiment four kinds of layers and find model with hidden layer of 5 nodes has relatively low MSE and better overall performance. Then we use Kmeans to cluster. For both methods, the 'elbow value' is around 5. To get a more accurate result on quality of cluster, Silhouette score and DB index is implemented. Compared with PCA, autoencoder performs better on both metrics which means dimensionality reduction from autoencoder is better on our dataset. Finally, we plots multiple distribution between clusters and original features to achieve customer segmentation and then conclude some significant characteristics of each cluster.

Though the whole process is very complete and compact, there are still some limitations and a lot can be done in

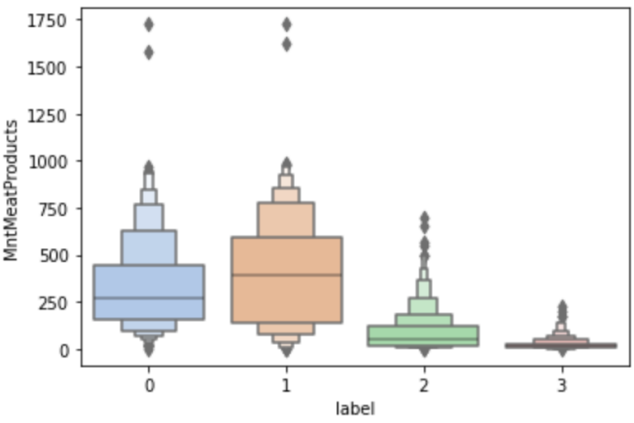

Figure 13. Distribution between clusters and Amount of meat products purchased

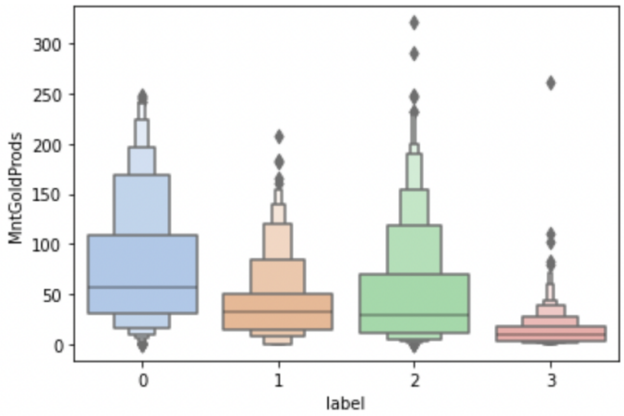

Figure 14. Distribution between clusters and Amount of gold products purchased

the future work. First, the processed dataset has only 18 features which is not a large dimension. It can more logical to have a baseline of clustering result on the dataset without dimensionality reduction compared with reduced one. In addition, we only try simple four autoencoder models. In the future work, we can use more complicated autoencoder like Convolutional autoencoder to achieve better performance.

6 References

[1] Alkhayrat, M., Aljnidi, M. Aljoumaa, K. A comparative dimensionality reduction study in telecom customer segmentation using deep learning and PCA. J Big Data 7, 9 (2020)
[2] Ferré, Q., Chèneby, J., Puthier, D. et al. Anomaly detection in genomic catalogues using unsupervised multi-view autoencoders. BMC Bioinformatics 22, 460 (2021)