Clustering of Travel Review Ratings Against Each Category per User

SHERY SHAJAN

Faculty of Engineering,
Environment and Computing,
Coventry University
MSc Data Science and
Computational Intelligence
(ECT104) Stage 1
Coventry, United Kingdom
cislop@uni.coventry.ac.ukline 1: 5th

***Abstract*—The importance of travel reviews has become an important factor for travel related decisions. The Travel review readers perceive reviews posted by other customers on having several advantages over information from travel service providers. Also, most of the review readers think the other travelers' reviews are most likely to contain up to date information, enjoyable and reliable. Here I have taken the topic Travel Review Rating Dataset, the reason why I choose this topic is that I am a kind of person who love traveling. But sometimes I have problems like where should I visit? Are there any other places matched with my lifestyle? Often, I spent hours to search for interesting places to go out, which is a loss of time. What if we can build a recommended system which can suggest or predict you several interesting venues based on your preferences. With the information from TripAdvisor.com, Reviews on destinations in 10 categories mentioned across East Asia are considered. Here in my course work I am going to implement the Machine Learning techniques in the data set, which I got from Kaggle. My dataset consists of the reviews on destinations in 10 categories mentioned across East Asia. There are 980 Instances and 11 Attributes. The data applies clustering algorithms using Python programming language with the use of scientific libraries.**

Keywords— clustering, KMeans, Agglomerative Clustering, styling, DBSCAN

I INTRODUCTION

Identifying the interest of the travelers and their preferences on choosing a group of venues based on their interest is too complicated for the tourist providers or travel agencies. For a better understanding the service providers needs to know the traveler's behavior which helps the service providers to design their services and improve their strategies and satisfying their clients. In order to understand this in today's world we have n number of Travel service providers. Most consumers or the travelers will check their preferences and also will update their reviews in their respective sites.[1] Based on these travel review ratings we can group the review ratings given by each user using Machine Learning Techniques. As mentioned here we are using the data from Kaggle and the dataset is from tripadvisor_review.csv. As we know computers are faster than humans in processing data. So, when a traveler plans for a trip, based on his previous visits the computer can recognize and suggest him the places. These suggestions or predictions show how the machine learning is powerful. Here we will look at how we can cluster the bunch of data based on the user's interest and decide their preferences. [2]

II THE DATASET

Dataset was obtained from Kaggle Machine learning Repository. It contains 980 instances and 11 Attributes with no missing values. Each instance is one person's unique rating. The first attribute defines the User as User ID and remaining 10 categories are the destinations mentioned across East Asia. *TABLE1* displays the summary of the data set describing features and its types.

No.	Description	Type	Categorical value range
1	Attribute 1 : Unique user id		
2	Attribute 2 : Average user feedback on art galleries		
3	Attribute 3 : Average user feedback on dance clubs		
4	Attribute 4 : Average user feedback on juice bars		
5	Attribute 5 : Average user feedback on restaurants		
6	Attribute 6 : Average user feedback on museums		
7	Attribute 7 : Average user feedback on resorts		
8	Attribute 8 : Average user feedback on parks/picnic spots		
9	Attribute 9 : Average user feedback on beaches		
10	Attribute 10 : Average user feedback on theaters		
11	Attribute 11 : Average user feedback on religious institutions		

III DATA PREPARATION

Dataset a collection of data. Here in this section, we are analyzing the data and preparing for the data to apply the algorithms. On the first place as an initial data analysis, I imported all the necessary libraries (Like pandas, NumPy, matplotlib and seaborn) and loaded the dataset.

My initial inferences on the data are as below

- Original data was in the form of .CSV file.
- With the help of. head () function which retrieved the first five observations in the data set.
- Then I found the total number of rows and columns in the dataset – 980 entries
- Except User Id all other columns are having float64 and integer values.
- No attribute column is having null or missing values
- I have set the User Id and renamed the categories based on the venues.
- On describing the data, the summary statistics of the data was obtained.
- On verifying the overall rating, the values range is from 0 to 4, where 0 is being poor and 5 being Excellent.

These steps enabled me to get a good glimpse about the data. Then for analyzing the Overal1 all distribution, I have plotted the boxplot [fig1. Overall Rating Distribution], which shows the distribution of quantitative data in a way that facilitates comparisons. [5]

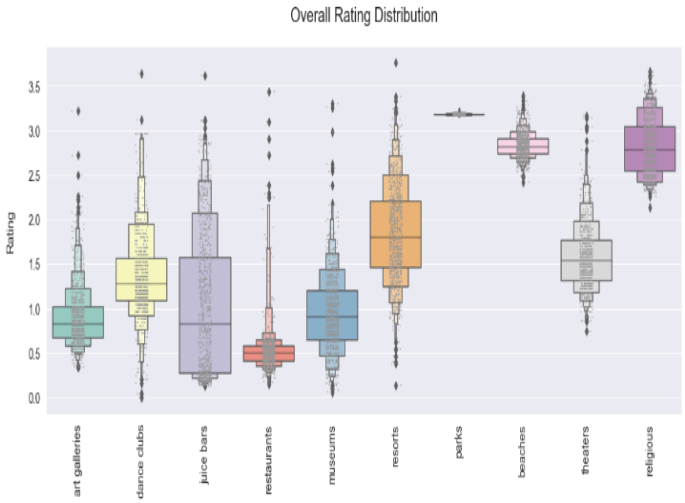

fig1. Overall Rating Distribution

In order to check the how closely the variables are correlated to each other. That is, it explains how one or more variables are related to each other. Thus, correlation analysis helped to quantify the degree to which the attributes are related. On plotting the data, I was able to find a Linear relationship between attributes. The same is defined in below figure [fig2. Correlation Mapping].

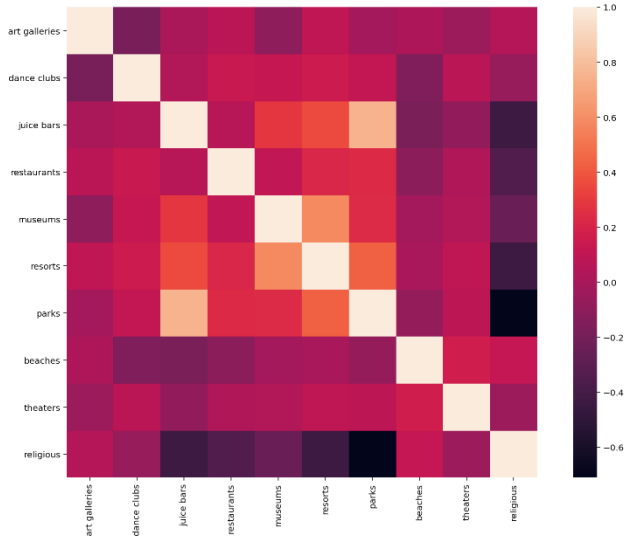

fig2. Correlation Mapping

Here this shows the stated correlation between the places a user wishes to travel. The line of 1.00s going from the top left to the bottom right is the main diagonal, which shows that each variable always perfectly correlates with itself. This matrix is symmetrical, with the same correlation is shown above the main diagonal being a mirror image of those below the main diagonal.

The below diagram shows the average rating of each category in below figure [fig5. Average rating of each category]

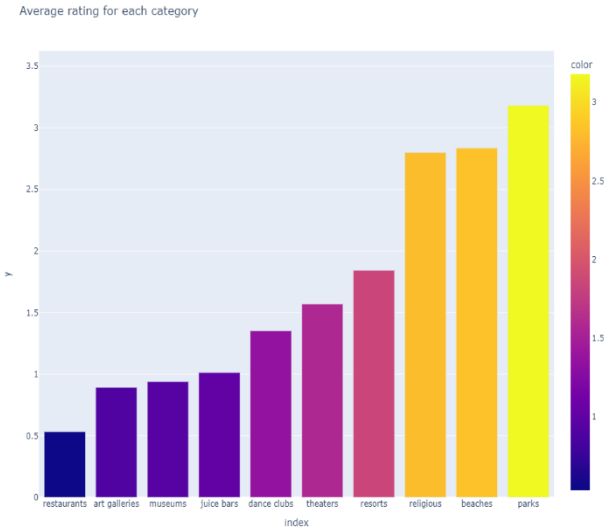

fig5. Average rating of each category

IV APPLICATION OF MACHINE LEARNING CLASSIFICATION TECHNIQUES

We can focus on segmenting the user reviews and then cluster them into different preferences.

IV.A K-Means Clustering

The widely used algorithm.This is a calculation that is generally utilized and material. The initial step is to choose various bunches haphazardly, every one of which is

addressed by a variable 'k'. Then, each bunch is allocated a centroid, i.e., the focal point of that group. Characterize the centroids as distant from one another as conceivable to lessen variety. After every one of the centroids with respect to off from every information point is relegated to the bunch whose centroid is at the nearest distance. Here we can discover right number of grouping calculations and look at their outcome.

We would first be able to go with chief compound investigation; [10] this will assist with tracking down the low dimensional portrayal of the perceptions that clarify a decent part of the difference. On playing out the calculation on PCA segment with scaled information, just 2 rule segments which address just 42% of unique data [* Refer Code fig3.Selecting appropriate bunch number with elbow method]. Since total clarified change proportion is 42%, this may not outcome so great bunching execution. [4] So, I will progress forward K-Mean bunching examination on PCA_scaled information with number of groups =4[fig4.Result of K-Means Clustering]. On clustering the results, my inference is as below.

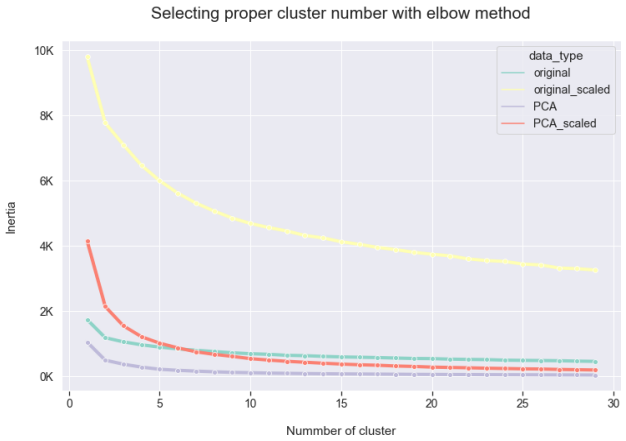

fig3.Selecting proper cluster number with elbow method

- Cluster#0(Green): User who prefer Entertainments like beaches, theater and dance clubs.
- Cluster#1(Orange): Users are like food lovers, who likes resorts, museums, parks, juice bars, and restaurant.
- Cluster#2(Light Blue)- Art Lovers, dedicated to art gallery.
- Cluster#3(Pink) – religious

K-Means clustering result in 4 clusters(segments) of user as follow figure [fig4.Result of KMeans Clustering].

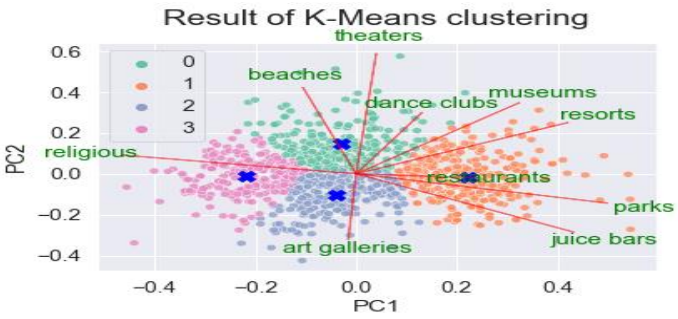

fig4.Result of KMeans Clustering

IV.B Agglomerative Clustering

A solo learning strategy in AI model that derives the information design with no direction and name. It's anything but an individual from Hierarchical Clustering family which work by consolidating each and every bunch with the interaction that is rehashed until all the information have become one group. To acquire the ideal number of bunches, the quantity of groups should be decreased from at first start n group, i.e., n approaches the complete number of information focuses). For discovering the closeness between two bunches we are joining the two groups by registering them. We have utilized the Dendrograms for showing the plan of the groups created by the relating investigations and is likewise used to notice the yield of progressive (Agglomerative) bunching [11] in figure [fig6. Dendrogram]

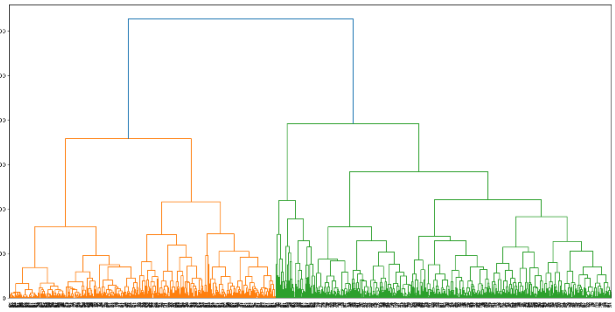

fig6. Dendrogram

IV.C DBSCAN

Density Based Clustering is again a solo learning strategy. This is Density based bunching. It can find groups of various shapes and sizes from an enormous measure of information, which is containing clamor and exceptions. [7] Calculations start by picking a point (one record) x from your dataset aimlessly and allot it's anything but a bunch 1. Then, at that point it checks the number of focuses is situated inside the ε (epsilon) distance from x. Assuming this amount is more noteworthy than or equivalent to minPoints (n), thinks about it as center point, then, at that point it will pull out every one of these ε-neighbors to a similar bunch 1. It will then, at that point inspect every individual from bunch 1 and track down their separate

ε - neighbors. On the off chance that some individual from bunch 1 has n or more ε-neighbors, it will grow group 1 by putting those ε-neighbors to the bunch. It will keep extending bunch 1 until there are no more guides to place in it. [8] Here on applying the DBSCAN over the categorized data we can see 3 clusters as output in figure [fig7.DBSCAN cluster representation]. The homogeneity is approximately 1.0 and the completeness calculated is approximate to 0.647. And the Silhouette Coefficient is 0.296

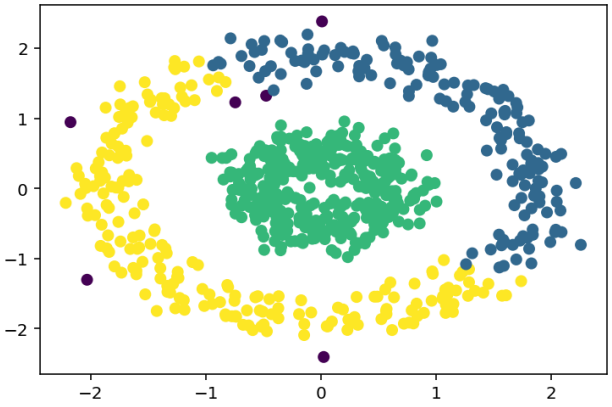

fig7.DBSCAN cluster representation

V DISCUSSION AND CONCLUSION

Finally, from this course work we can conclude on how the clustering algorithms can help to propose most relevant venues. Here we consider the user reviews from TripAdvisor. On applying the various cluster algorithms, we can find the different ways of clustering the travel reviews. The preference of each set of people. As we know this clustering of travel review is not an easy task as it has many discrepancies. Therefore, we are able to get an estimated results on applying the machine learning techniques to the data set. Distribution of rating in each attraction categories are not the same. Some of them have wide range distribution while some are distributing narrowly in low rating region.
We also compared each data preparation methods and found out that with Standardized data and reduced dimension with PCA. We are able to divide review user into 4 separable groups using KMeans Clustering and 3 separable groups based on DBSCAN clustering and Agglomerative clustering. With information of how different user group prefer different attraction. We can further use this information to build recommender system where we can recommend specific type of attractions to specific user to enhanced their traveling experience and boost revenue for attraction point.

VI REFERENCES

[1] Marinella Petrocchi (2018) A Study On Online Travel Reviews Through Intelligent Data Analysis [online] from <https://www.researchgate.net/publication/327293886_A_study_on_online_travel_reviews_through_intelligent_data_analysis>

[2] Michela Fazzolari Information Technology & Tourism (2018) A Study On Online Travel Reviews Through Intelligent Data Analysis. [online]

[3] Kyung Hyan Yoo (2007) Online Travel Review Study - Role & Impact of Online Travel Reviews [online]

[4] Iswarya Nagappan (2019) Travel Review Ratings [online] from <https://www.kaggle.com/ishbhms/travel-rating-reviews-analysis>

[5] Titov Vladislav (2020) [online] Travel-Review-Rating-Clustering

[6] Jupyter Team (2015) What Is the Jupyter Notebook? — Jupyter
Notebook 5.2.2 Documentation [online] available from <https://jupyternotebook.readthedocs.io/en/stable/examples/Notebook/What%20is%20the%20Jupyter%20Notebook.html>

[7] Nagesh Singh Chauhan, kdnuggets (2020) An introduction to the DBSCAN algorithm and its Implementation in Python [online] from <https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html>

[8] Pavan Kumar Raja (2020) Great Learning Team [online] from <https://www.mygreatlearning.com/blog/dbscan-algorithm/>

[9] Tutorials point Machine Learning - Hierarchical Clustering, Clustering Algorithms [online] from <https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_clustering_algorithms_overview.htm>

[10] Dr. Michael J. Garbade (2018) Understanding K-means Clustering in Machine Learning [online] from <https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1>

[11] Cory Maklin Hierarchical Agglomerative Clustering Algorithm Example in Python [online] from <https://towardsdatascience.com/machine-learning-algorithms-part-12-hierarchical-agglomerative-clustering-example-in-python-1e18e0075019>

VII APPENDIX

In [1]:
```
import pandas as pd
import numpy as np
import os
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
#Loading Dataset
df_trvl_rvw = pd.read_csv('tripadvisor_review.csv')
df_trvl_rvw = df_trvl_rvw.set_index('User ID')
df_trvl_rvw.head()
```

Out[2]:

User ID	Category 1	Category 2	Category 3	Category 4	Category 5	Category 6	Category 7	Category 8	Category 9	Category 10
User 1	0.93	1.8	2.29	0.62	0.80	2.42	3.19	2.79	1.82	2.42
User 2	1.02	2.2	2.66	0.64	1.42	3.18	3.21	2.63	1.86	2.32
User 3	1.22	0.8	0.54	0.53	0.24	1.54	3.18	2.80	1.31	2.50
User 4	0.45	1.8	0.29	0.57	0.46	1.52	3.18	2.96	1.57	2.86
User 5	0.51	1.2	1.18	0.57	1.54	2.02	3.18	2.78	1.18	2.54

Below is the attachment of Final Code- contains python code used to pre-process the data and apply the clustering techniques, alongside the original data.

Clustering_PART1.pdf

Clustering_PART2.pdf