User Clustering of Travel Ratings

Aman Singh, Riddhi Barbhaiya, Karl Jang

12/13/2020

Background:

With the advent of mobile devices allowing users to access the Internet ubiquitously, travel guidelines have largely transitioned from passive mass media to interactive social media. Users' public impressions can now influence others' decisions on their activities. Although such a phenomenon is not limited to the following method of reviews and geographical surroundings, ratings are the easiest to be interpreted quantitatively as they represent overall perception of places and highly-developed continents may generate better results because most people, both locals and tourists, can afford mobile devices. Therefore, in this project, we analyze average Google review ratings for 24 different categories of attractions across Europe, such as bars, restaurants, and gyms, ranging from 0 to 5 stars.

In this project, we cluster subgroups of people by their ratings across three different categories for two main reasons. First, to see if they similarly rate identical categories of attractions. Second, in order to analyze whether similar categories of attractions have comparable review ratings. Clustering people based on how they rate places may also allow us to pattern what people generally prefer or defer, inferred from incomplete ratings data. In turn,we could use this information to notify and create a relevant recommendation system determined by the individuals' previous ratings.

Statistical Question of Interest:

In order to elaborate both of our aforementioned justifications, we first explore how to best cluster users based on their review ratings. To achieve this goal, we use k-means clustering and generate silhouette index. For inferring about the users' characteristics based on our clustering, we graph the relations in 3-dimensional scatterplots.

How can we best cluster users based on their review ratings?

What can we infer about users from our clustering?

Analysis Plan:

Descriptive Analysis:

We will summarize the distribution of ratings for each category to understand how users typically rate this category. We then create a correlation matrix between the categories to identify patterns of relationships. This will help us to understand how the categories co-vary.

Main Analysis:

For this project, we primarily use k-means to cluster the data. We then evaluate the quality of the clusters with the silhouette index. In order to use k-means to find x clusters, we pick x points to be the center of the clusters. Then, for each point, we calculate the distance from the point to each of the centers picking the lowest and assigning it to a cluster, subsequently recalculating the centers as the results update. We repeat this process until the centers have been found. We essentially want the points within clusters to be homogeneous and each cluster to be distinct from the others. To measure this property, we use the silhouette index. A silhouette coefficient is computed for each point and then to get the index for all data

points, we average the coefficient. The silhouette coefficient is calculated as follows:

$$S_i = \frac{b_i - a_i}{max(a_i, b_i)}$$

,where a_i is the average dissimilarity of point i from all other points in the cluster, and b_i is the smallest average distance between point i and other points in each cluster.

Initially, we cluster all ratings across all 24 categories using k-means. By doing so, our clustering has a silhouette indexed very close to 0, indicating that the within cluster variation is widely prevalent and the distance from other clusters is small.

We then perform clustering for three categories for a total of 27 combinations, with 9 being chosen by the study group and 18 chosen at random. This was done to analyze clustering patterns within smaller subsets of the data, also producing more coherent clusters. We utilize silhouette index while clustering to pick the numbers of clusters to adopt, between 2 to 7. Then, we visualize the clustering of the three best combinations and the three worst in terms of silhouette index to locate clusters.

In addition to the two methods of clustering mentioned above, we also attempt to use spectral clustering to see if there are any nonlinear clusters in the data that is not accounted for by k-means. Unfortunately, we are unable to perform spectral clustering due to its intensive, time-demanding nature in running the code. Our computer would freeze up due to the size of the data set and the requirement of a considerable computing power.

Results:

Descriptive Analysis:

We run summary statistics to learn more about the data set. This correlation graph shows how each feature is correlated to another feature in the data set, with the blue color showing a negative correlation, and the red color showing a positive correlation. The summary

table shows all the features in the data set with there respective mean, standard deviation, minimum, maximum, the first quartile and the third quartile:

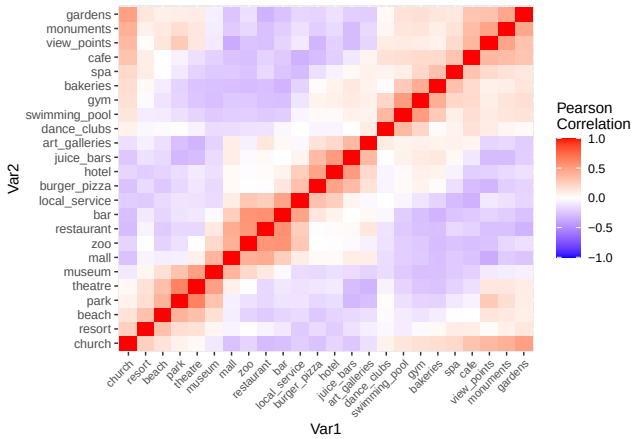


Table 1:

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
church	5,456	1.456	0.828	0	0.9	1.8	5
resort	5,456	2.320	1.421	0	1.4	2.7	5
beach	5,456	2.489	1.248	0.000	1.540	2.740	5.000
park	5,456	2.797	1.309	0.830	1.730	4.092	5.000
theatre	5,456	2.959	1.339	1	1.8	4.3	5
museum	5,456	2.893	1.282	1.110	1.790	3.840	5.000
mall	5,456	3.351	1.413	1	1.9	5	5
zoo	5,456	2.541	1.111	0.860	1.620	3.190	5.000
restaurant	5,456	3.126	1.357	0.840	1.800	5.000	5.000
bar	5,456	2.833	1.308	0.810	1.640	3.530	5.000
local_service	5,456	2.550	1.382	0.780	1.580	3.220	5.000
burger_pizza	5,456	2.078	1.249	0.780	1.290	2.282	5.000
hotel	5,456	2.126	1.407	0.770	1.190	2.360	5.000
juice_bars	5,456	2.191	1.577	0.760	1.030	2.740	5.000
art_galleries	5,456	2.206	1.716	0.000	0.860	4.440	5.000
dance_clubs	5,456	1.193	1.107	0.000	0.690	1.160	5.000
swimming_pool	5,456	0.949	0.974	0.000	0.580	0.910	5.000
gym	5,456	0.823	0.950	0	0.5	0.8	5
bakeries	5,456	0.970	1.204	0.000	0.520	0.860	5.000
spa	5,456	1.000	1.194	0	0.5	0.9	5
cafe	5,456	0.966	0.930	0	0.6	1	5
view_points	5,456	1.751	1.599	0	0.7	2.1	5
monuments	5,456	1.531	1.316	0	0.8	1.6	5
gardens	5,456	1.561	1.172	0	0.9	1.7	5

Clustering Results:

For each combination in Table 2, we computed k-means clustering for k = 2,…,7 and then calculated the silhouette index. We chose 7 as the max number of clusters because for most of the clusters, the silhouette index declines after 6. Table 2 shows all the combinations we considered, the resulting number of clusters, and the best silhouette index. Note that the three rows highlighted in blue are the best clusterings based on the silhouette index, and the three others highlighted in red are the worst:

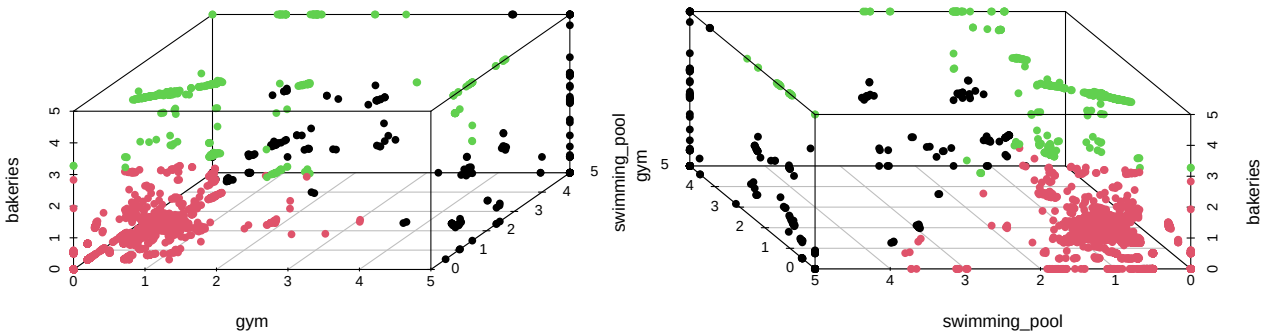
Table 2: Quality of Clustering Combinations with Index and clusters

Clusters	Number of clusters	Silhouette Index
View Points,Park,Beach	6	.5620834
Theater,Art Galleries,Cafe	4	.5994621
Restaurant, Bar, Local Service	7	.5431321
Gardens,Gym,Spa	4	.7145031
Dance Clubs, Theater, Spa	4	.6046022
Resort, Bakeries, Zoo	3	.5156392
Art Galleries, Bar, Bakeries	5	.551172
View Points, Museum, Burger-Pizza	4	.5344893
Zoo, Resort, Bar	2	.4211836
Bar, Burger-Pizza,Juice Bars	5	.5932957
Zoo, Hotel, Dance Clubs	3	.5758509
Bakeries, Cafe, Park	4	.6029162
Gardens, Resort, Swimming Pool	4	.6468726
Restaurant, Theater, Bakeries	4	.522686
Museum, Spa, Art Galleries	6	.5726653
Park, Burger-Pizza, Church	3	.5004534
Dance Clubs, Monuments, Museum	4	.5812412
Art Galleries, Gym, Bar	5	.5632228
Gym, Park, Mall	5	.5490831
Gym, Swimming Pool, Bakeries	3	.7618442
Resort, Beach, Spa	4	.5719415
Church, Art Galleries, Monuments	3	.6400298
Gardens, Hotel, Monuments	5	.6351123
Bar, Gym, Local Service	4	.5752452
Theater, View Points, Museum	7	.4979704
Theater, Museum, Restaurant	5	.4809223
Hotel, Art Galleries, Gardens	6	.6596394
Art Galleries, Monuments, Gardens	5	.6537716

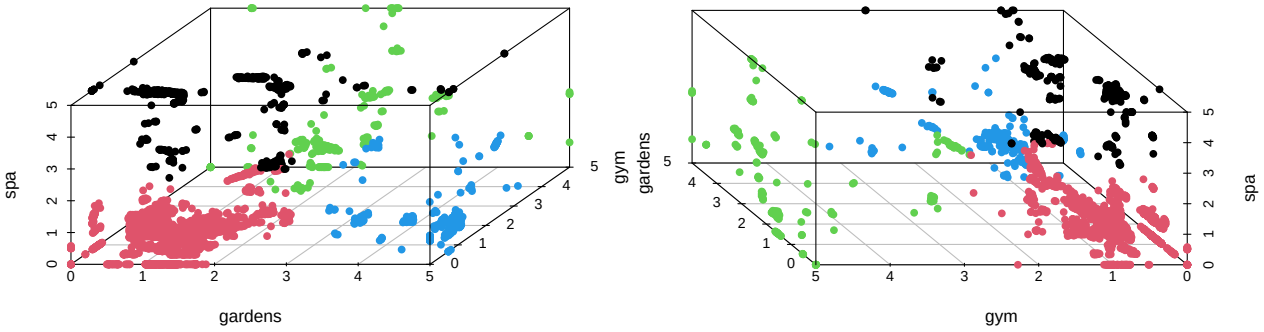
The top 3 clusters are colored in Blue, the lowest 3 in Red

Graphs of Best Clusterings:

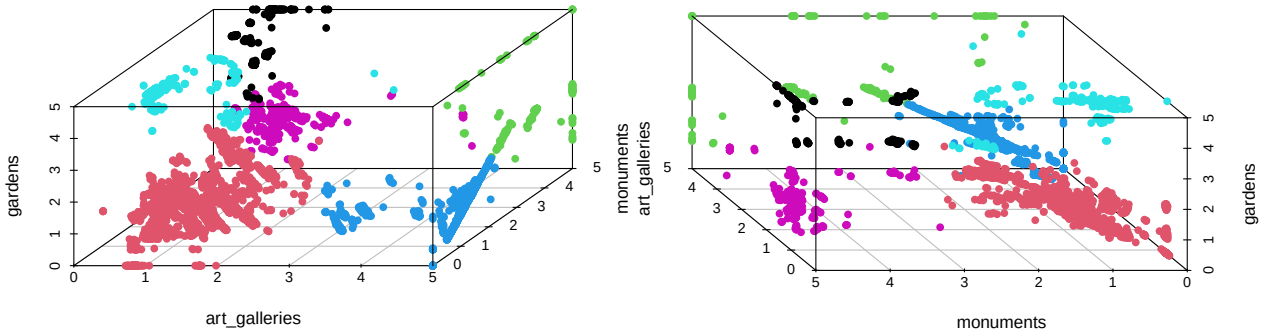
Following, we have graphed the 3 best clustering out of all 27 combinations.



We have three clusters for this combination. The lime cluster represents reviewers who somewhat highly rated bakeries between three to five stars, whereas most of their ratings on gyms and bakeries were poor. These raters may be big eaters since their appetite could positively influence their perception of food places, including bakeries. Conversely, they would not enjoy exercising as much as discovering new cuisines, hence rating fitness facilities like gyms and swimming pools with low stars.



There are four distinct clusters present. The black cluster indicates a group of people reviewed gardens relatively highly with three stars or higher, but their ratings on spas were mixed and the gyms were poorly rated. These raters could be nature lovers because gardens consist of multiple flora of potential interest, while their views on indoor spaces greatly differ. One possible reason why gyms were rated lower than spas could be due to their inclusion of manmade materials such as barbells and treadmills.

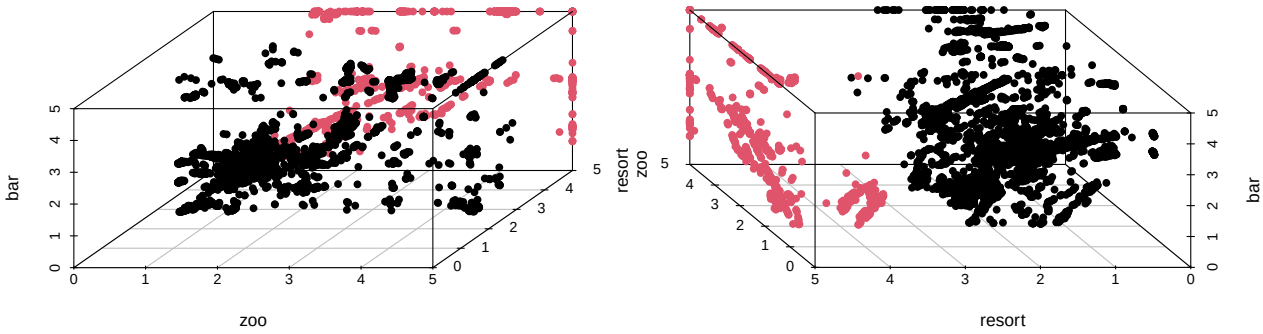


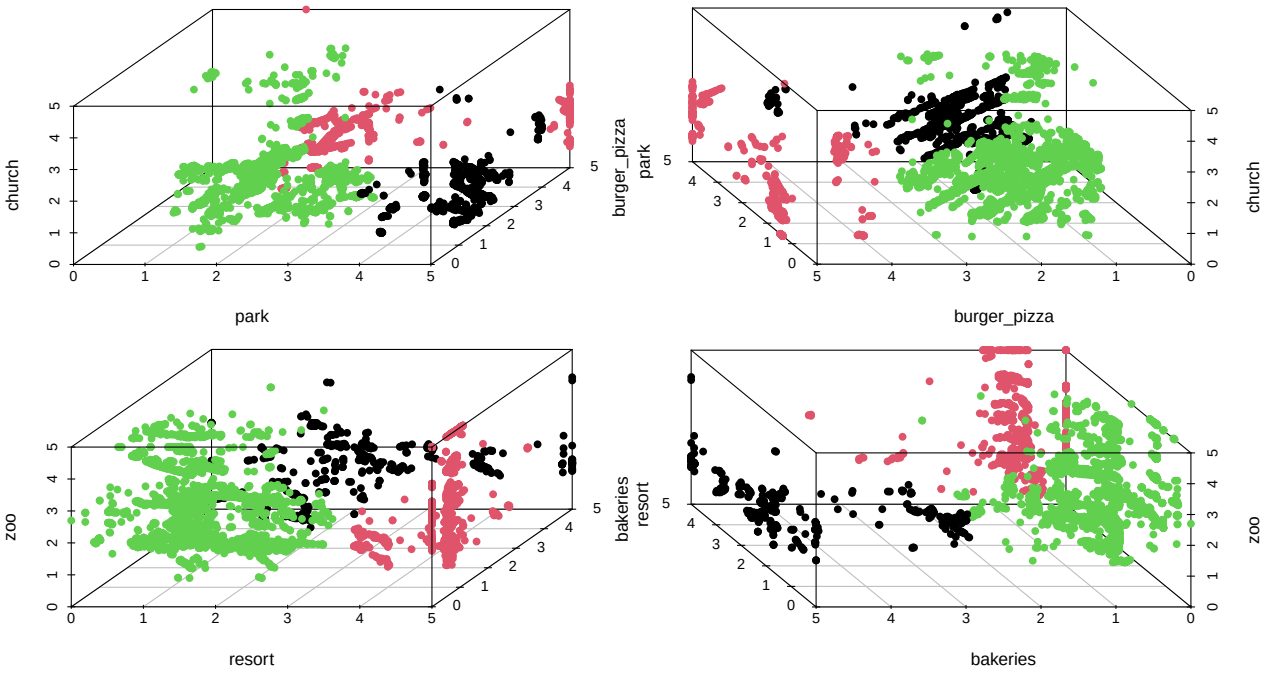
Five clusters are generated from this combo. The people in the black cluster have highly rated art galleries but their reviews on gardens and monuments were mixed. They might reflect older audiences because the elderly tend to prefer indoor spaces such as art galleries but their inclination towards outdoors varies greatly depending on their underlying conditions.

To our surprise, all three graphs have similar clusters; some reviewers tended to rate every category of attractions unfavorably, usually at two stars or lower on average. We conclude they are Internet trolls because such a uniform pattern cannot be observed if their ratings reflect their actual perception, especially since combinations one and three do not share any category of attractions.

Graphs of Worst Clusterings:

Following, we have graphed the 3 worst clustering out of all 27 combinations. We include them to depict that the quality of clustering partially depends on the categories considered.





Extra Credit:

Introduction:

Background:

In this portion of the project, we attempt to implement matrix completion in the google travel reviews dataset. The data includes average ratings from approximately 5,000 people on 24 different categories of attractions(spa, resort, etc).

In reality, one may obtain only partial review data and would like to infer the rest to provide accurate recommendations. To simulate this situation, we randomly select a portion of the data to treat as missing and then use matrix completion to predict the missing ratings. Because we have complete data, we can then compare the predictions to the true ratings. This line of analysis is inspired primarily from "Exact Matrix Completion via Convex Optimization" by Candes and Retchet. To do the actual computation, we use the soft impute algorithm(Mazumder, 2010).

Statistical questions of Interest:

Our goal is to complete the matrix. In doing so, we assume that the data has some low rank structure, there is some information about ratings in each row and column(ensured by random sampling), and that the matrix is incoherent. These conditions are requirements for the completion problem to be well posed(Candes 2008). To evaluate our predictions, we compare our completed matrix to the whole data.

Analysis Plan:

Matrix Completion:

Let X be the data matrix which is 24 by 5,426 (5,426 users and 24 categories reviewed). X_{ij} is the average review rating of the jth person in the ith category. Let Ω be the set of observed entries. This set is created by randomly sampling points to treat as observed from the full matrix X. Let

$$P_\Omega(X) = \begin{cases} X_{ij} & (i,j) \in \Omega \\ 0 & otherwise \end{cases}$$

and let Z be the matrix that is completed using the soft impute method.

The objective of the matrix completion is to find the matrix Z such that it has the smallest possible rank and the observed entries are as close to the original as possible. This can be written as follows, where δ is some small real number:

$$\|P_\Omega(X) - P_\Omega(Z)\|_F < \delta min \ \|Z\|_*$$

Here, $\|Z\|_*$ is the nuclear norm defined as follows. $\sigma(Z)$ are the singular values of Z. The nuclear norm is used as an estimator of the rank because it is easier to minimize.

$$\|Z\|_* = \sum_{k=1}^{n} \sigma_k(Z)$$

To solve this, we decided to use the soft impute algorithm (Mazumderet al.,2010) to minimize the following function. It essentially uses soft thresholding to solve the objective function specified below:

$$\min_{Z} f_\lambda(Z) = \frac{1}{2} \|P_\Omega(X) - P_\Omega(Z)\|_F^2 + \lambda \|Z\|_*$$

To choose the hyperparameter, we run soft impute on the following λ values (1,6,11…,101). Then we calculate the frobenius norm of the residual as shown below. We choose the λ values which results in the smallest value for the residual because it gives the matrix that is closest to the original.

$$total\, residual = \|Z - X\|_F$$

In reality, in a matrix completion problem, one wouldn't know the ground truth. Then, to pick a λ, one should treat a subset of values of the incomplete matrix as missing (creating a train/test set). This would allow one to validate the λ that one picks to see ensure the matrix completion generalizes well.

To look more closely at our completed matrix, we calculate partitions of the residuals in the following ways:

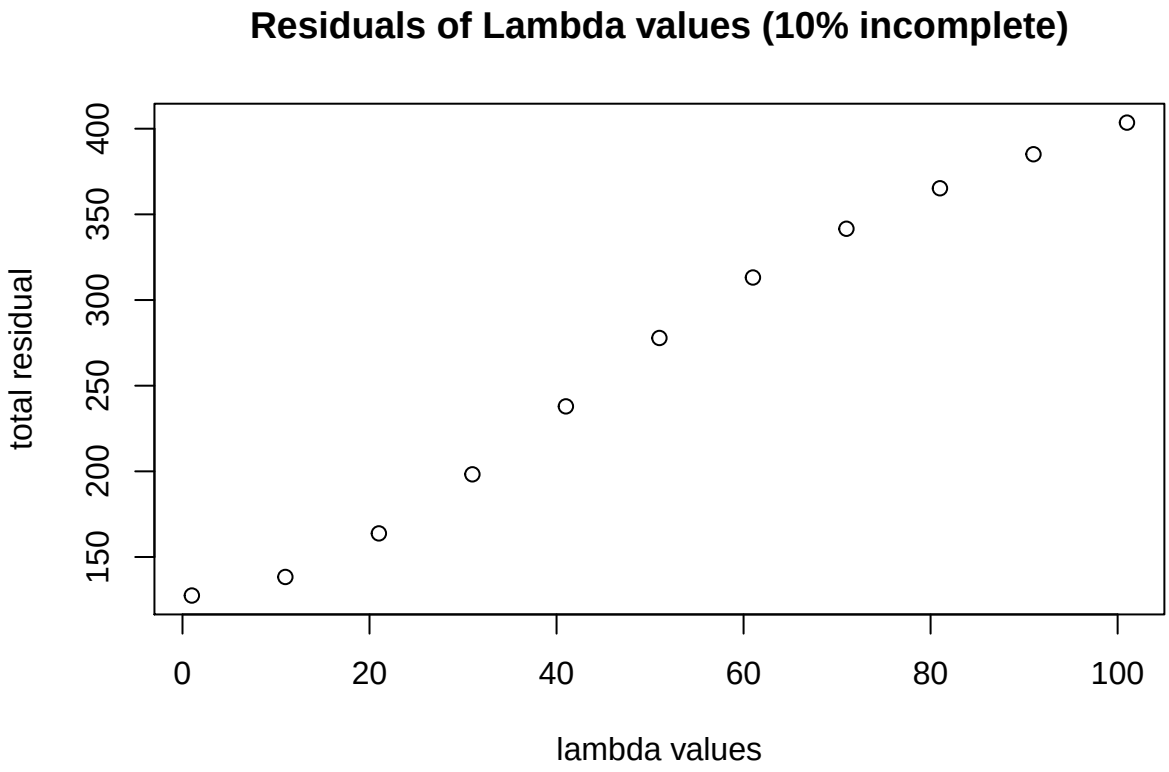
$$observed\, residual = \|P_\Omega(Z) - P_\Omega(X)\|_F$$

$$unobserved\, residual = \left\|P_{\notin\Omega}(Z) - P_{\notin\Omega}(X)\right\|_F$$

Results:

Inferential Analysis:

10% incomplete

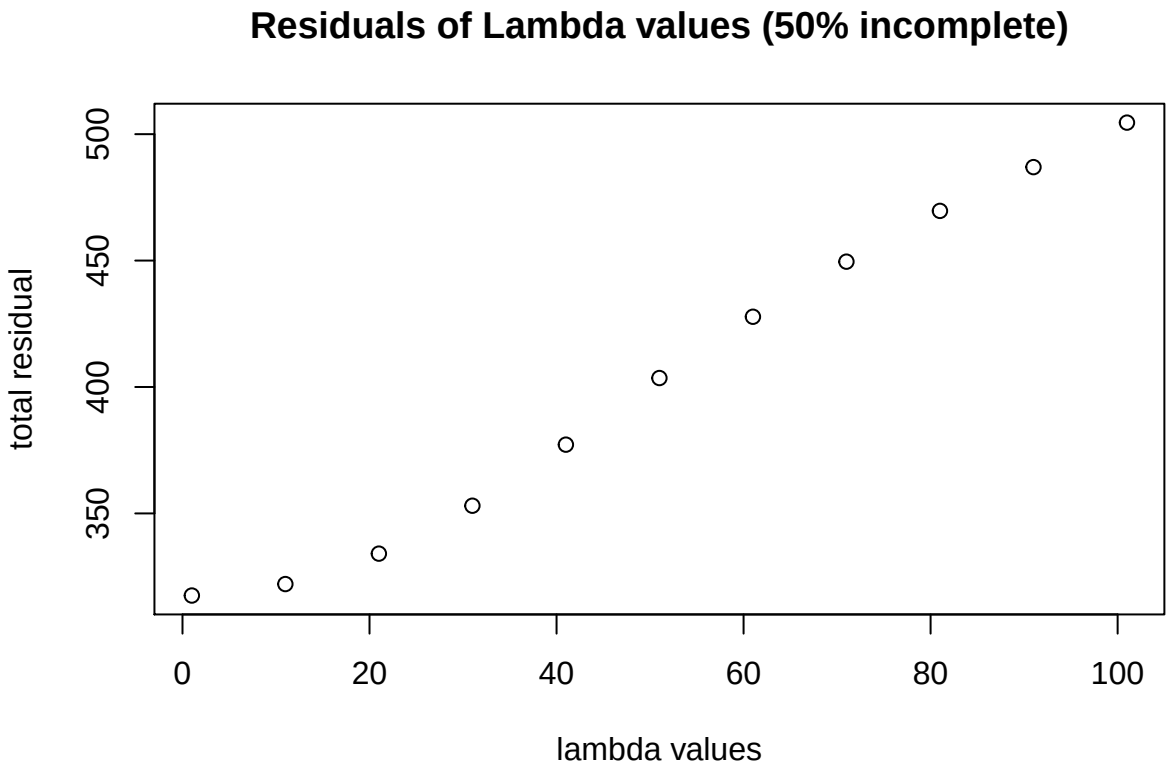


Total Residual = 127.5200442

Observed Residual = 23.999997

Unobserved Residual = 103.5200472

50% incomplete



Total Residual = 317.5298296

Observed Residual = 23.9977846

Unobserved Residual = 293.532045

Conclusion:

As expected, there is a larger residual error in completing the matrix that has more missing data. The matrix completion works quite well; there is about 0.0215199 error per completed entry. Because the observed residual is quite small, both 10% and 50% completion are able to recover the observed values with high accuracy.

Citations:

Mazumder, Rahul & Hastie, Trevor & Tibshirani, Robert. (2010). Spectral Regularization Algorithms for Learning Large Incomplete Matrices. Journal of machine learning research : JMLR. 11. 2287-2322.

Candès, E.J., Recht, B. Exact Matrix Completion via Convex Optimization. Found Comput Math 9, 717 (2009). https://doi.org/10.1007/s10208-009-9045-5