Using K-Means and GMM for Clustering in Revenue Management

Daniel Chocano (dchocano3@gatech.edu)

April 29, 2022

1 Problem Statement

Clustering is a canonical unsupervised machine learning task. With clustering we aim to partition a collection of items into subsets, or "clusters", so that the items within one cluster are more similar to one another than to items in a different cluster. The items we seek to cluster are represented by a set of measurements and can be thought of as a set of unlabelled data points in the feature space (ESL, FML).

Clustering has a wide range of applications in many fields, both in academia and business. Revenue management is no exception. At a high level, revenue management (RM) is the discipline of maximizing the revenue of a company that sells a perishable good or service. The process of maximizing revenue involves two aspects essentially: (1) finding the optimal amount of product to sell each day, or, in cases where production quantities are more or less fixed, minimizing daily spoilage; and (2) finding the optimal price in order to extract the most out of the customer demand, or "volume", and consumer surplus, or "willingness-to-pay". RM is widely used in the lodging and airline industry.

In the day-to-day RM operations of an airline, especially one with a hub-and-spoke network, the itineraries offered to customers are managed in so-called Origin-and-Destinations (ODs). An OD is essentially where a journey originates and ends, e.g. Zurich to New York or Cape Town to Hong Kong. Regardless of how a city pair is flown, either nonstop or with transfers, the OD is the same. Very often in practice, when pricing and steering several thousand ODs, airlines bundle up their ODs geographically. Say, all ODs from different origins in Germany to different destinations in Spain and Portugal are grouped together because they are geographically close. While such a subset stands to reason, it does so on the grounds of geography alone, and disregards completely important OD measurements such as booking behavior, travel patterns, compartment mix, products purchased etc.

The purpose of this analysis is to show that machine learning clustering methods that take into account different OD measurements can reach far better results to grouping ODs than plain geographical proximity. In other words, this analysis proposes that closeness in the feature space is a better similarity metric than physical proximity. This analysis uses both k-means and mixture of Gaussians for the clustering task and compares both methods to one another and to geographical grouping.

2 Data Set

2.1 Data Source

The data set used in this analysis is from the Lufthansa Group Network Airlines (LHG). It includes all tickets sold for origin Zurich with departure date between January 1 and December 31 2019. Only longhaul intercontinental destinations are taken into account.

The data set has $n = 132$ rows and $m = 34$ columns. Rows are observations. Columns are the measurements, or "features". (Ergo we speak of the "feature space"). The data has been aggregated on a destination city level, i.e. each row represents the aggregated bookings from Zurich (ZRH) to a particular intercontinental destination city, say Manila (MNL) or Zanzibar (ZNZ).

The features include various measurements describing OD characteristics, e.g. is the OD offered nonstop or only with a transfer, how many corporate travellers were identified etc., as well as passenger purchasing and traveling behavior such as how many weeks before departure did passengers book their tickets, how long did they stay at their destination, what products they purchased etc. The complete list of original features and their description are provided in Appendix A.

The data fits well to the unsupervised learning task as there are no labels for OD clusters. The only feature that could be construed as an a priori label in the case of geographical grouping is the area where a destination is located, e.g. Africa, Asia Pacific, Near Orient and so on.

2.2 Feature Engineering

There is little wrangling needed for this data set. Most numerical features are proportions, expressed as fractions ranging from 0 to 1; though many features have many values lower than 0.5. The only numerical features provided as counts are *ms_larger* and *ms_min*, ranging from 0-16 and 0-6 respectively. Similarly, the metric *comp_score* has values greater than one. In order to make all these metrics more readily comparable, we normalize the features, i.e. we scale them to a range between 0 and 1. That way we don't run the risk of under or overestimating their magnitude.

Even after scaling, we see several features with potential outliers; some of them quite extreme – see Figure 1. We chose not to remove these observations from the data as we wish to retain as much generalization as possible for the models we will train with this data. The data used here is relatively limited in scope. It includes only ODs originating in Zurich. If we apply the models trained with it to larger data sets with a wider scope, there might be many ODs with these apparent "more extreme" values. Take the metric *ms_larger* for instance which measures how many competitors on that OD have a market share higher than LHG. Since origin Zurich is a market where LHG enjoys a relatively strong position, it is rather uncommon to have competitors with much higher market shares. However, if we expand our analysis to, say, all markets in Europe, there will be many ODs at this range of *ms_larger* as LHG is prone not to have a dominant position in each and every OD. We want our models be able to handle such cases correctly.

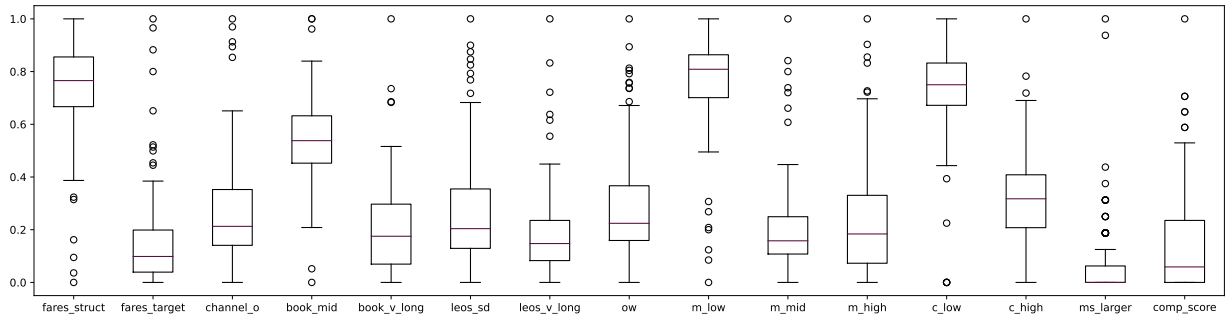

Figure 1: Boxplots of scaled features with (potential) outliers.

Finally, in order to construct the feature matrix, X, the three categorical variables (*dest_cty*, *dest_area* and *dest_ctry*) and the two indicator variables (*online* and *lx_online*) are removed to concentrate the feature space on measurements concerning customer characteristics as well as purchasing and travelling patterns. These are the more interesting metrics and; thus, should play a more central role in the OD partitioning.

2.3 Exploratory Data Analysis

Before moving on to the next section, it is worth having a look at the feature space in terms of the first two principal components. This strengthens our intuition as we can observe (1) which features contribute the most in explaining the variability in the data (and are; thus, candidates for features driving similarity or dissimilarity among partitions), and (2) how the observations are distributed over the first two principal components.

By applying PCA over the observations, i.e. reducing along the n observations, we see the magnitude of the final features scaled on each of the first two principal components – see Figure 2. The features *leos_long*, *leos_short*, *book_short* and *corp_ind* have the biggest influence on the first principal component, whereas *ms*, *ms_min*, *channel_o*, *fares_struct*, *ms_larger* have the biggest impact on the second.

On the other hand, by applying PCA over the features, i.e. reducing along the m features, we plot the data distribution in a lower dimensional representation. Figure 3 shows the two-dimensional kernel density estimation

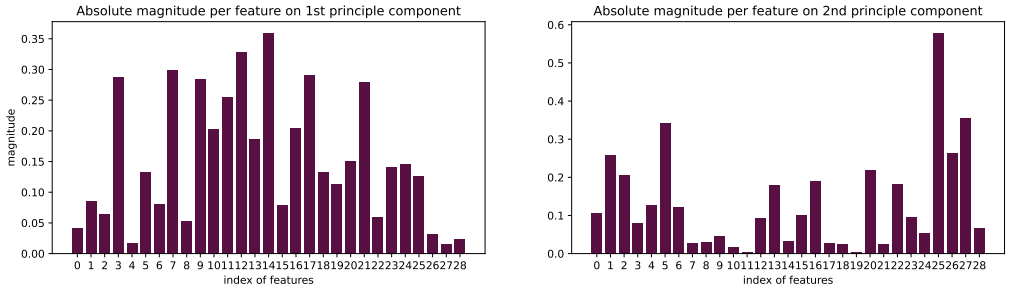

Figure 2: Magnitude of each of the features, in absolute terms, on the first two principal components.

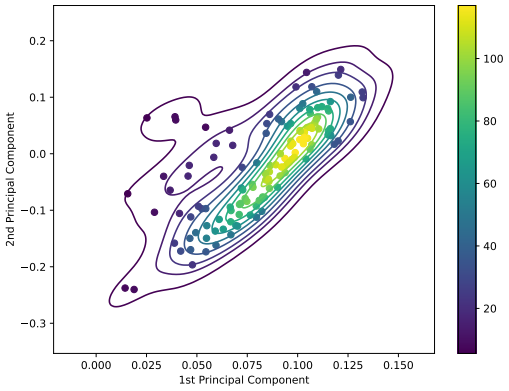

Figure 3: KDE-scatter plot of all observations over the first two principal components.

(KDE) scatter plot over the first two principal components. By visual inspection it can be inferred that assuming a unimodal distribution is rather noisy for this data; at least in relation to the first two principal components. This hints to the existence of different clusters or a mix of distributions generating the data.

3 Methodology

This analysis makes use of two so-called prototype methods: k-means clustering and Gaussian mixture model (GMM). Both methods are relatively simple, essentially nonparametric, and highly effective for prediction (ESL).

In case of k-means, the prototypes are the k centroids, or cluster centers. After an initial guess, the centroids are computed iteratively to minimize the within-cluster variance. So when the algorithm converges, each data point is assigned to the centroid closest to it. "Closeness" is usually defined by Eucledian distance in the feature space. The only hyperparameter, if you will, to be defined in the learning process is the number of centroids, k. We finally use silhouette analysis to determine an optimal value of k centroids.

In the case of GMM, the prototypes are individual uni-modal Gaussian densities, each with their own mean (centroid) and covariance matrix (roundness and decay). GMM is a so-called generative model that produces smooth posterior probabilities describing the underlying densities that generated the data we observe. These probability distributions can be used to cluster the data points. Thus, GMM is often referred to as a "soft" clustering method, while k-means is "hard" (ESL). We use the Akaike Information Criterion (AIC) to determine the number of components in GMM.

The motivation for trying out both methods and comparing their results is the fact that k-means has a clear drawback which stems from the simple distance-from-cluster-center point allocation it performs. This allocation runs under the implicit assumption that the clusters are circular; an assumption which won't always hold. In addition, the standard k-means algorithm doesn't provide a way to account for other shapes. This drawback is addressed by GMM, which can produce ellipsoid clusters (PDSH).

4 Evaluation and Final Results

4.1 Hard Clusters with K-Means

Figure 4 plots an elbow diagram of the within-cluster sum of squares (WCSS) over the number of centroids k. Naturally, as we keep adding clusters, the WCSS decreases. The biggest drop comes from adding the second cluster. Every subsequent cluster added yields diminishing returns in terms of marginal WCSS decrease. Even though there is no clear "kink", one can conclude from Figure 4 that the marginal return per additional centroid is considerably lower as of the ninth cluster. These findings can be further fine-tuned via silhouette analysis.

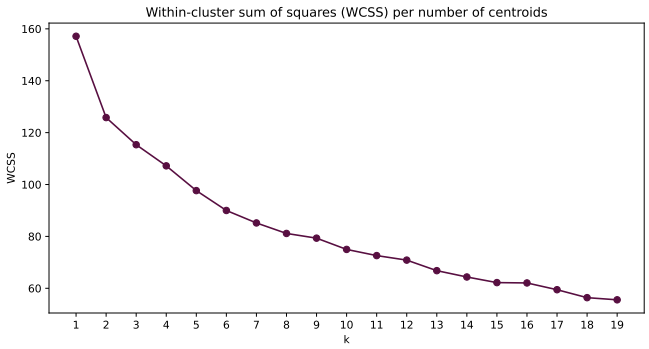

Figure 4: Elbow diagram for $k = 1, \ldots, 20$.

When considering $k = 2, \ldots, 8$, using two or five centroids achieves the best results both in terms of overall silhouette scores and consistency in cluster size and performance – see Figure 5. The left plot shows the average silhouette score (dotted red vertical line) and the individual score per observation in each cluster. The silhouette score ranges between $[-1, +1]$. A value near $+1$ indicates the observation is far away from the neighboring clusters. A value of 0 indicates the sample is on or very close to the decision boundary between two neighboring clusters. A negative value indicates those observations might have been assigned to the wrong cluster. This gives a perspective into the density and separation of the formed clusters (SKL). The respective silhouettes also visualize the size of each cluster. The right plot shows the KDE scatter plot over the first two principle components, which reinforces our intuition on how the clusters are built over the feature space.

With $k = 2$ one cluster performs better than the other. Most of the observations in cluster 0 have above average silhouette scores. This first cluster is also larger than the second. At the same time, roughly 2/3 of the observations in cluster 1 have lower-than-average scores with a few very close to the decision boundary of the blue cluster. However, a two-cluster partition seems overly simplified considering the variety in the data. This speaks strongly for more than just two clusters.

With $k = 5$ one cluster, now number 1, continues to outperform the others in both scores and size. Cluster 2 though small has mainly above-average scores. Cluster 3 captures almost as many observations as cluster 1 though it has some observations with slightly negative values. Individually, clusters 0 and 4 don't perform very well; however overall, they do have few observations with above-average silhouette scores. These two clusters seem to have some overlaps with the others and are, thus, not perfectly separable. The strong performance of clusters 1, 2 and 3 when setting $k = 5$ warrants the presence of five clusters.

Considering these findings, the final k-means model is trained with $k = 5$. Figure 6 visualizes the box plots for the different features, again unscaled, for each of the five clusters. These box plots collectively describe the typical (mean and spread) customer purchasing and traveling behavior as well as the competitive situation in the ODs in each respective cluster. At the same time, Table 1 provides the detailed list of destinations per cluster.

K-means partitions the ODs geography-agnostically. This is evident when looking at clusters 1 and 4. Yet it is interesting to note that some clusters do follow somewhat of a geographic pattern. Cluster 2 for instance is comprised solely of destinations in Sub-Saharan Africa while cluster 0 includes destinations in Africa and Middle East.

Further inspection of the clusters uncovers some interesting patterns. For example, customers traveling from Zurich to cluster 2 predominantly purchase leisure target group products, short before departure and on third party

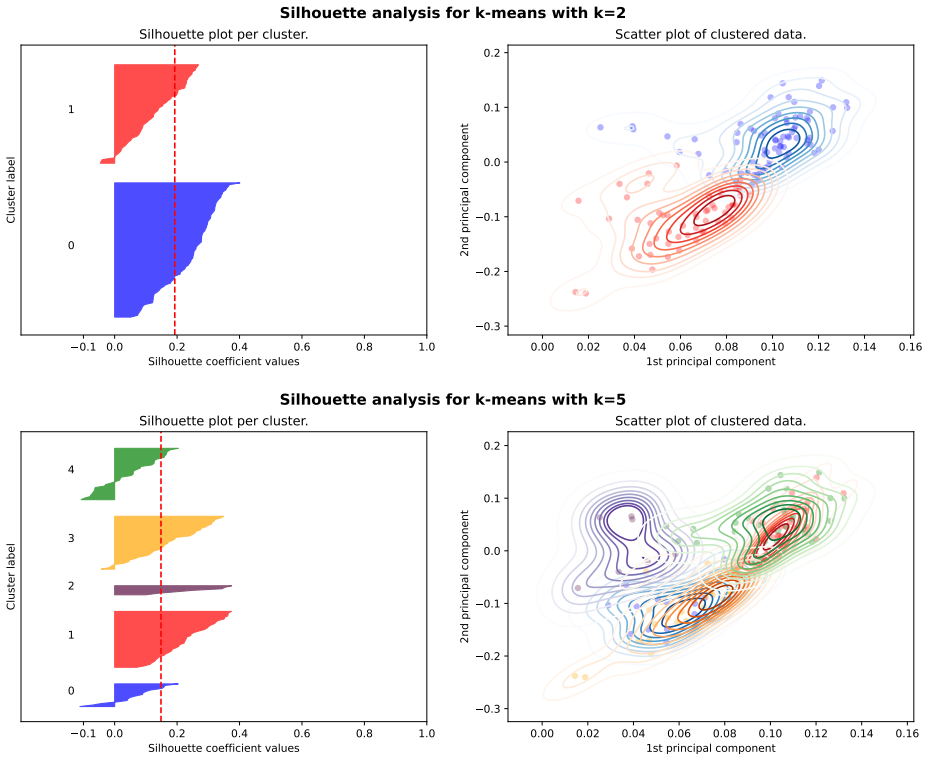

Figure 5: Per cluster silhouette coefficients and KDE scatter plot for k=2 and k=5.

Figure 6: Per cluster feature box plots for final k-means model with $k = 5$.

channels. These customers book in mid Economy Class and low booking classes in the higher compartments. In terms of travel patterns, customers tend to stay long term at their destination and almost never short term. Looking at the competitive landscape, LHG tends to be dominant in these markets.

A contrasting example to cluster 2 is cluster 4. Here, customers purchase mainly structural fares and almost no target group products. Though they still book primarily via third party channels, they do so much further in advance. These markets are also more fragmented and LHG tends to have lower-than-average market shares.

Cluster	Destinations
0	ACC, AGA, ALG, AMM, BEY, CAI, CAS, DJE, ETH, JRO, MCT, RAK, SID, SSH, TLV, TUN
1	BGI, BUE, CPT, CUN, CWB, DAR, DEL, DEN, DUR, EBL, HAV, HRG, JNB, LAS, LAX, MIA, MLE, MRU, NBO, ORL, PLZ, PUJ, RIO, RMF, SAN, SAO, SEZ, SFO, SIN, SJO, TPA, TYO, USM, VFA, VRA, YMQ, YVR, YYC
2	ABJ, DKR, DLA, EBB, FIH, KGL, YAO
3	ABV, ALA, ASB, ATL, AUS, BAH, BJS, BLR, BOM, BOS, CHI, CLT, DFW, DMM, DTT, DXB, HKG, HOU, KWI, LOS, MAA, NDR, NKG, NYC, PHC, PHL, PHX, RUH, SEA, SHA, SHE, SLC, SSG, TAO, WAS, YOW
4	AKL, ANC, BKK, BNE, BOG, CHC, CMB, DPS, FMY, HKT, IGU, JKT, KUL, LAD, LIM, MBA, MBJ, MEL, MEX, MNL, NGO, OSA, PER, POP, PTY, SCL, SEL, SGN, SYD, THR, TPE, WDH, YHZ, YTO, ZNZ

Table 1: ODs per cluster for final k-means model with $k = 5$.

4.2 Soft Clusters with GMM

Figure 7 plots the Akaike Information Criterion (AIC) for an increasing number of Gaussian densities, or components. AIC is particularly useful in this setting as it allows us to estimate the prediction error, which is something we can hardly test with unlabelled data. The lowest AIC score is reached with $n = 13$ components. This means that up until the 13th component, each additional one adds predictive value to the model. As of the 14th component, each additional one rather overfits the data. So, the final GMM is trained assuming this many individual Gaussian probability distributions.

Figure 8 visualizes the box plots for the different features for each GMM component. Again, we can observe the differences in customer booking and travelling patterns as well as the differences in competitive landscape. The complete list of ODs in each component is provided in Table 2.

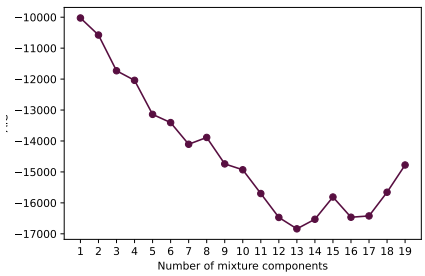

Figure 7: Akaike information criterion (AIC) for $n = 1, \ldots, 20$.

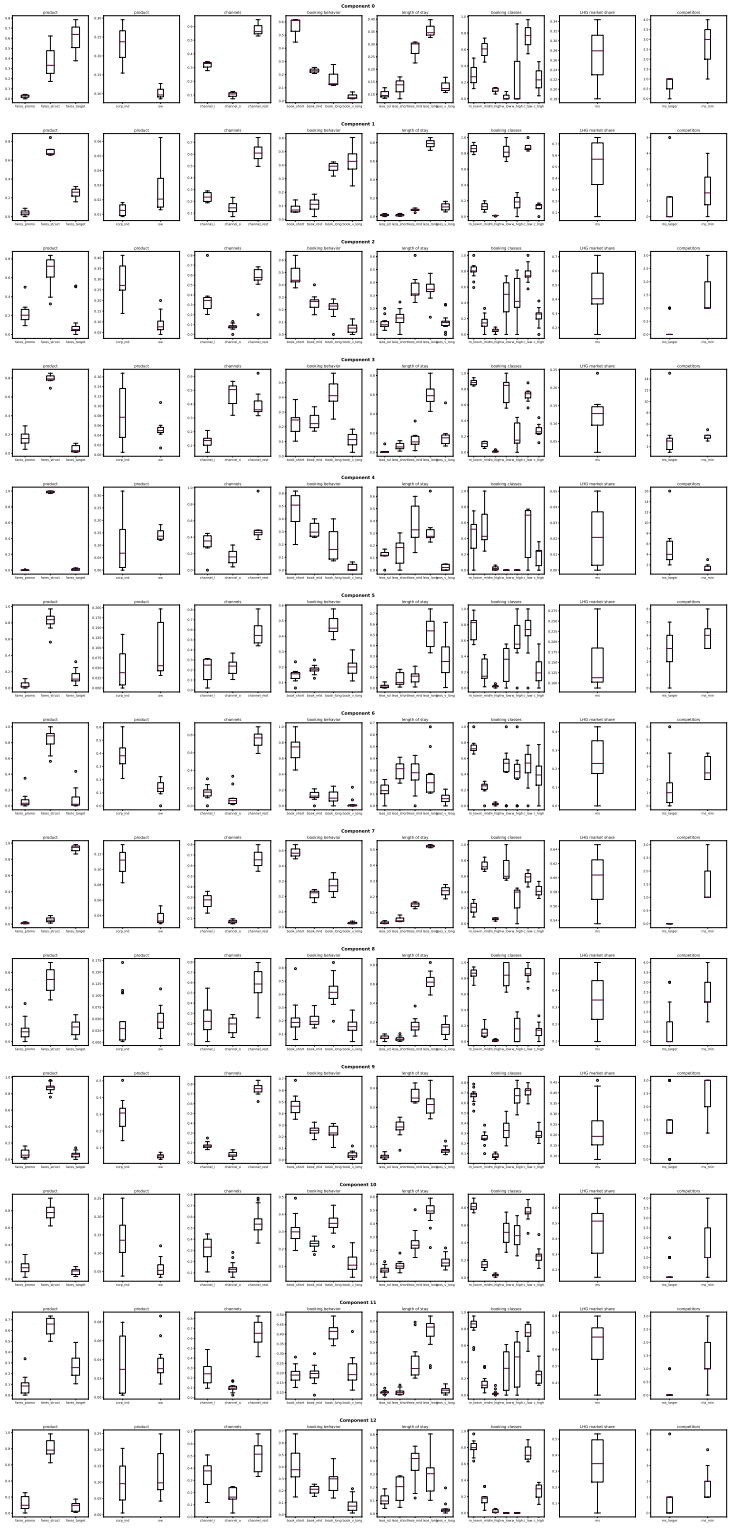

Figure 8: Per component feature box plots for final GMM model with $n = 13$.

Component	Destinations
0	ABJ, ACC, DKR
1	ANC, PLZ, SJO, WDH
2	BJS, BOM, BOS, CHI, DXB, EBB, ETH, HKG, KGL, NKG, PHC, SHA, SHE
3	AKL, DPS, JKT, KUL, LIM, MNL, TPE
4	AGA, ALG, CAS, SID, SSH, TUN
5	BNE, CHC, HKT, MEL, OSA, PER, POP, SCL, SGN, SYD, ZNZ
6	ABV, ALA, ASB, BAH, DMM, KWI, LOS, NDR, RUH, SSG
7	DLA, FIH, YAO
8	BGI, BKK, BOG, CMB, FMY, IGU, LAD, MBA, MBJ, MRU, NGO, PTY, SEZ, USM, YHZ, YVR, YYC
9	ATL, AUS, BLR, CLT, DFW, DTT, HOU, PHL, PHX, SLC, TAO, WAS
10	BUE, CWB, DEL, DEN, JNB, LAX, MAA, MEX, NBO, NYC, ORL, RIO, SAN, SAO, SEA, SEL, SFO, SIN, THR, TYO, YMQ, YOW, YTO
11	CPT, CUN, DAR, DUR, HAV, HRG, LAS, MIA, MLE, PUJ, RMF, TPA, VFA, VRA
12	AMM, BEY, CAI, DJE, EBL, JRO, MCT, RAK, TLV

Table 2: ODs per component for final GMM model with $n = 13$.

4.3 Conclusions

Compared to plain geographical grouping, both k-means and GMM bundle up ODs considerably differently. The OD grouping in these unsupervised learning models is based on proximity in the feature space and not geographical proximity. Interestingly enough, in both models some groups have ODs that are geographically close to one another. However, this pertains solely to the fact that the customer behavior and market situation in these ODs are similar.

The final k-means and GMM models trained here both seem plausible in their own right. They fare relatively well measured by their individual performance metrics and their resulting partitions display important similarities in OD characteristics. The only drawback from a GMM with $n = 13$ components in this particular data set is that we end up with some relatively very small subsets, for example component 0, 1 and 7.

In general terms, regardless of the model used, further inspection of the resulting OD subsets can serve well to validate the plausibility of the model's data partition as well as when trying to further optimize pricing and steering decisions in RM in order to exploit patterns in customer behavior and the competitive situation in the different markets.

References

- (ESL) Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Predictions* (2nd ed.). Springer Series in Statistics.
- (FML) Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of Machine Learning* (2nd ed.). The MIT Press.
- (PDSH) VanderPlas, J. (2016). *Python Data Science Handbook*. O'Reilly Media.
- (SKL) Selecting the number of clusters with silhouette analysis on KMeans clustering, accessed April 29 2022, https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Appendix A: List of Original Features

Here the complete list of features in the data set.

Name	Description
dest_cty	destination city in IATA three-letter code
dest_area	destination area code
dest_ctry	destination country in IATA two-letter code
online	binary indicator: '1' if destination is operated solely by LHG, '0' otherwise
lx_online	binary indicator: '1' if destination is operated by carrier LX, '0' otherwise
fares_promo	proportion of promotional fares sold
fares_struct	proportion of structural fares sold
fares_target	proportion of target group fares sold
corp_ind	proportion of bookings made with an identified corporate credit card
channel_i	proportion of bookings made on own dotcoms
channel_o	proportion of bookings made on third party online platforms
channel_rest	proportion of bookings made on other channels, e.g. traditional travel agencies
book_short	proportion of bookings made 0-4 weeks prior departure
book_mid	proportion of bookings made 5-10 weeks prior departure
book_long	proportion of bookings made 11-31 weeks prior departure
book_v_long	proportion of bookings made >31 weeks prior departure
leos_sd	proportion of passengers with same day return
leos_short	proportion of passengers staying 1-4 days at their destination
leos_mid	proportion of passengers staying 5-10 days at their destination
leos_long	proportion of passengers staying 11-28 days at their destination
leos_v_long	proportion of passengers staying >28 days at their destination
ow	proportion of bookings purchasing a one-way ticket
sa_ind	proportion of bookings with SA indicator
m_low	proportion of economy class bookings made in the low booking classes
m_mid	proportion of economy class bookings made in the middle booking classes
m_high	proportion of economy class bookings made in the high booking classes
w_low	proportion of premium economy class bookings made in the low booking classes
w_high	proportion of premium economy class bookings made in the high booking classes
c_low	proportion of business class bookings made in the low booking classes
c_high	proportion of business class bookings made in the high booking classes
ms	LHG's market share
ms_larger	number of competitors with a higher market share than LHG
ms_min	number of carriers with >8% market share
comp_score	competitive score (internal metric)