

Association for Information Systems

AIS Electronic Library (AISeL)

MENACIS2021

MENA

11-14-2021

Identification of centroids of Mohammed V airport arrivals.

Imane Belqasmi

Akram El Korchi

Abdellah Menou

Follow this and additional works at: <https://aisel.aisnet.org/menacis2021>

This material is brought to you by the MENA at AIS Electronic Library (AISeL). It has been accepted for inclusion in MENACIS2021 by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Identification of centroids of Mohammed V airport arrivals.

Introduction

The Air Traffic Management (ATM) market is projected to grow from an estimated USD 14.1 billion in 2018 to 18.8 billion by 2025. This growth can be attributed to increasing airport investments and modernization of ATM infrastructure.

In 2018, Moroccan airports received more than 22,534,000 passengers, according to the National Office of Airports (ONDA) which represents 10 percent more than in 2017. The Agadir airport recorded an increase of 25 percent in passengers; Marrakech 21 percent; Fez 17 percent; Ouarzazate 37 percent; Essaouira 25 percent; and Dakhla 20 percent and even though Mohammed V International Airport in Casablanca has known only an increase by 4 percent from 2017 but it handled 43 percent of Moroccan international air traffic in 2018, receiving 9,732,044 passengers which makes it the busiest airport of Morocco. Increasing air travel has led to an increase in commercial air traffic and since commercial air traffic has been higher and has greater frequency than tactical air traffic. This in turn leads to the increasing need for ATM systems to cope with the increasing commercial air traffic and one of its strategies is aircraft trajectory data by using analysis models.

Models for predicting aircraft motion are an important component of modern aeronautical systems. In this given paper, we developed a method to analyze aircraft's motion and evaluate its efficiency in terminal airspace, the controlled airspace surrounding a given airport. The method fits the model based on a historical dataset of radar-based position measurements of aircraft landings and takeoffs at that airport. We find that the model generates realistic trajectories, provides accurate predictions allowing to control the performance of the system.

This paper presents a characterization of air traffic performance based on aircraft tracking data recorded by surveillance systems. We use unsupervised learning and apply a flight trajectory clustering framework to identify traffic patterns at the terminal for destination Mohammed V International Airport.

Methodology

The data-driven approach for air traffic performance characterization is based on three steps:

First, data is exposed to a data pre-processing which allows to clean, filter and structure the flight tracking dataset. Moreover, a trajectory clustering framework is applied to this data, given as result identified air traffic patterns at the terminal airspace as clusters. For each cluster, a nominal route is calculated. A detailed description of each step is provided next.

A. Data Description :

The raw dataset contains flight tracks from 7 days of August of 2018 obtained through the FlightRadar24 flight tracking service (FlightRadar24, 2019). FlightRadar24 is one of the various online flight tracking services made available after the introduction of new surveillance technologies in ATM, such as Automatic Dependent Surveillance - Broadcast (ADS-B). FlightRadar24 relies on a huge network of crowdsourced ADS-B receivers around the world that pick-up flight information (flight ID, aircraft position etc) broadcasted by the aircraft's ADS-B transponder and send this information to their servers to provide opensource live flight tracking. The raw datasets report one-minute updates of aircraft state, including flight ID, latitude, longitude, altitude, speed, origin airport, destination airport and aircraft type. Flight trajectories were then segmented according to the different flight phases. To extract the

terminal area departure phase, we considered the trajectory information between the destination airport and the terminal area boundary, which was modeled as a circle of 40 nm radius with its center at the destination airport aka MohammedV.

B. Resampling

Data resampling is then performed to transform each time-series into a high dimensional feature vector of fixed dimension and enable the assessment of similarity between flight trajectories using standard Euclidean distance. The resampling approach normalizes the time stamps for each trajectory, divides it into a fixed number of equally sized time blocks and linearly interpolates the spatial position for the fixed number of normalized time stamps. The result is a feature vector of 2D spatial position evenly spaced in time.

C. Clustering at spatial scale:

The idea behind this framework is to identify spatial patterns of aircraft movement. Actually, clustering is an unsupervised learning method that aims to identify groups of similar observations in a dataset without prior knowledge about the existence of these groups or about how the observations are distributed among them.

In this context, the goal is to find groups of similar trajectories in the spatial dimension. We define a group of spatially similar trajectories as a trajectory pattern.

This process requires then a data representation, a similarity/distance function and a clustering method. First of all, the portion of the trajectory associated with the airspace region of interest is extracted. In this case, to extract the terminal area phase, it was defined as the trajectory information between the airport runway threshold and the terminal area boundary, which was modeled as a circle of 40-mile radius with its center at the airport. The filtered flight trajectories are characterized by time-series of different lengths, depending on the time spent in the airspace volume.

DBSCAN algorithm :

The final requirement in the clustering methodology is the method for grouping similar observations. A density-based clustering algorithm – Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al., 1996) – is used for flight trajectory clustering. As the name of the algorithm suggests, this method is suitable for datasets with noise. In flight trajectory datasets, the standard routes and adaptations produce the core underlying patterns, yet abnormal trajectories can also occur for a variety of reasons and can be considered as noise. DBSCAN enables the identification of the core trajectory patterns in the presence of abnormal trajectory profiles. Other advantages of DBSCAN include the ability to discover non-convex clusters and no need to set the number of clusters a priori. DBSCAN relies on two input parameters in order to cluster the data space:

MinPts: a minimum number of points (observations);

ϵ : a distance threshold.

D. Results:

In the final step, the nominal routes are identified as a reference that could be used to measure both temporal or spacial efficiency.

Case study : Mohammed V Arrivals

In the last years, there is more talks about whatso-called “Big Data Analytics” since there is an increasing availability of data with new technologies (ADS-B: open-source flight tracking data “Flight Aware, FlightRadar24”) and as a consequence to it, opportunities to leverage the available system data for improved decision-making.

In our case, since the international airport of Mohammed V is known as the busiest airport of Morocco, we thought of a solution that will lighten the workload of its air traffic controllers by detecting the optimal arrival routes that pilots should follow. By this, we have provided a data that represents 7 days

of collecting from FLIGHTRADAR24, from 23/07/2019 to 30/07/2019 focused on Mohammed V Arrivals. Those information were regrouped within an R program file called "track_data" where it was separated by indicators. We can see in figure 1 that it was the first flight collected starting from 2pm. It includes each position (longitude and latitude) it took during his flight inside the TMA until his landing. Aside from that, you find the heading, speed, equipment, registration, origin, destination, identification (IATA/ICAO), Squawk code.

index	time	lat	lon	alt	heading	speed	eqpt	eqpt	eqpt	alt	alt	alt	alt	alt
1	2019-07-23 14:00:00	33.5854	10.1338	27430	135	28700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:01:00	33.5851	10.1337	27320	135	28300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:02:00	33.5847	10.1335	27200	135	27900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:03:00	33.5843	10.1333	27080	135	27500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:04:00	33.5839	10.1331	26960	135	27100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:05:00	33.5835	10.1329	26840	135	26700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:06:00	33.5831	10.1327	26720	135	26300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:07:00	33.5827	10.1325	26600	135	25900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:08:00	33.5823	10.1323	26480	135	25500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:09:00	33.5819	10.1321	26360	135	25100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:10:00	33.5815	10.1319	26240	135	24700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:11:00	33.5811	10.1317	26120	135	24300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:12:00	33.5807	10.1315	26000	135	23900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:13:00	33.5803	10.1313	25880	135	23500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:14:00	33.5799	10.1311	25760	135	23100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:15:00	33.5795	10.1309	25640	135	22700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:16:00	33.5791	10.1307	25520	135	22300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:17:00	33.5787	10.1305	25400	135	21900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:18:00	33.5783	10.1303	25280	135	21500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:19:00	33.5779	10.1301	25160	135	21100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:20:00	33.5775	10.1299	25040	135	20700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:21:00	33.5771	10.1297	24920	135	20300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:22:00	33.5767	10.1295	24800	135	19900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:23:00	33.5763	10.1293	24680	135	19500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:24:00	33.5759	10.1291	24560	135	19100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:25:00	33.5755	10.1289	24440	135	18700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:26:00	33.5751	10.1287	24320	135	18300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:27:00	33.5747	10.1285	24200	135	17900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:28:00	33.5743	10.1283	24080	135	17500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:29:00	33.5739	10.1281	23960	135	17100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:30:00	33.5735	10.1279	23840	135	16700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:31:00	33.5731	10.1277	23720	135	16300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:32:00	33.5727	10.1275	23600	135	15900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:33:00	33.5723	10.1273	23480	135	15500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:34:00	33.5719	10.1271	23360	135	15100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:35:00	33.5715	10.1269	23240	135	14700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:36:00	33.5711	10.1267	23120	135	14300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:37:00	33.5707	10.1265	23000	135	13900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:38:00	33.5703	10.1263	22880	135	13500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:39:00	33.5699	10.1261	22760	135	13100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:40:00	33.5695	10.1259	22640	135	12700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:41:00	33.5691	10.1257	22520	135	12300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:42:00	33.5687	10.1255	22400	135	11900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:43:00	33.5683	10.1253	22280	135	11500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:44:00	33.5679	10.1251	22160	135	11100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:45:00	33.5675	10.1249	22040	135	10700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:46:00	33.5671	10.1247	21920	135	10300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:47:00	33.5667	10.1245	21800	135	9900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:48:00	33.5663	10.1243	21680	135	9500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:49:00	33.5659	10.1241	21560	135	9100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:50:00	33.5655	10.1239	21440	135	8700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:51:00	33.5651	10.1237	21320	135	8300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:52:00	33.5647	10.1235	21200	135	7900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:53:00	33.5643	10.1233	21080	135	7500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:54:00	33.5639	10.1231	20960	135	7100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:55:00	33.5635	10.1229	20840	135	6700	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:56:00	33.5631	10.1227	20720	135	6300	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:57:00	33.5627	10.1225	20600	135	5900	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:58:00	33.5623	10.1223	20480	135	5500	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 14:59:00	33.5619	10.1221	20360	135	5100	A78	0102	7A	0A	450	AA01	0210	90
W 1	2019-07-23 15:00:00	33.5615	10.1219	20240	135	4700	A78	0102	7A	0A	450	AA01	0210	90

a. Trajectory visualization

In order to visualize things in R and to, specifically, be able to plot same data, it is advised to use the package leaflet. You create a Leaflet map with these basic steps. Starting by creating a map widget by calling leaflet(), then adding layers (i.e., features) to the map by using layer functions (e.g. addTiles, addMarkers, addPolygons, addPolylines) to modify the map widget. Repeat step 2 as desired. Finally, print the map widget to display it.

In this case, information are taken from track_data to print the trajectory of each flight all in once.

b. Resampling:

"Path resampling" process consist of a linear interpolation of the position made so that all paths can be described by a fixed-length vector.

INPUTS:

- track_data -> data frame with FR24 data
- NPOINTS -> indicates the desired vector size. For example, if NPOINTS =30, we will have 30 observations (lat, lon) describing the trajectory from terminal area entrance to airport arrival (40 nm). Since the FR24 data collection frequency is 1 min and the duration of flights in the terminal area is on the order of 30 minutes, NPOINTS = 30 is a reasonable resampling rate.

A screenshot of an RStudio IDE showing R code for trajectory resampling. The code defines a function 'resampling' that takes 'track_data' and 'npoints' as arguments. It uses 'library("mass4geom")' and sets 'npoints' to 30. A loop iterates over each flight ID in 'track_data', extracting 'time', 'lon', 'lat', and 'duration'. It then creates 'timesteps' and 'trajectories' arrays. The code is as follows:

```
1: resampling = function(track_data, npoints)
2: {
3:
4: library("mass4geom")
5:
6: npoints = 30
7: flag = 0
8:
9:
10: for (i in 1:length(track_data$id))
11: {
12: id = which(track_data$id == i)
13:
14: if (length(id) > 0)
15: {
16: time = track_data$time[id]
17: lon = track_data$lon[id]
18: lat = track_data$lat[id]
19: duration = track_data$duration[id]
20:
21: timesteps = time
22: timesteps[1] = 0
23: for (j in 1:length(time))
24: {
25: timesteps[j] = (time[j] - time[1]) / duration
26: }
27: timesteps[length(time)] = 1
28:
29:
30: trajectories = array(0, c(1, length(timesteps), 4),
31: dimnames=list(NULL, NULL, c("timesteps", "lon", "lat", "duration")))
32:
33: trajectories[,1,] = timesteps
34: trajectories[,2,] = lon
35: trajectories[,3,] = lat
36: trajectories[,4,] = duration
37:
38: step_size = 1 / npoints
39:
40: }
41: }
42: }
```

Figure 2 : Piece of the resampling coding

c. Clustering:

The process of clustering is based on DBSCAN Algorithm. After resampling trajectories, they are now ready to be compared. We apply to them the DBSCAN algorithm, and wait to see how much clusters these trajectories has been separated from each others. In order to guess the number of the clusters, the easy way is to visualize them.

```

1- clustering <- function(resampled_track_data, nKpts, epsilon){
2
3   ##### PARTE 2: processo de clusterização usando o algoritmo DBSCAN.
4
5   # INPUTS:
6   # resampled_track_data -> matriz de trajetórias resamostradas
7   # nKpts -> parâmetro de DBSCAN
8   # epsilon -> parâmetro de DBSCAN
9   # OUTPUT:
10  # clusterId -> vetor indicando o cluster ao qual cada trajetória foi alocada, 0 indica trajetória outlier.
11
12  library("dbscan")
13
14  normalized_input <- scale(resampled_track_data)
15
16  nKpts = 3;
17  epsilon = 1.7;
18
19  cluster_output <- dbscan(normalized_input, epsilon, nKpts, weights = "ALL", borderPoints = "TRUE")
20
21  clusterId <- cluster_output$cluster
22
23  return(clusterId)
24
25
26
27
28
}

```

Figure 3 : Rcode for clustering



Figure 4: plotting and visualizing clusters.

It appears that there are four different colors which means four clusters, each one oriented toward a direction. This only mean that there are basically four ways to reach the airport of Mohammed V that the aircraft could take.

d. Centroids:

After the clusters of trajectories are identified, a nominal route is determined for each cluster by solving a 1-median problem, in other words, a representative trajectory for each cluster is obtained by calculating the “center” of the cluster. For each nominal route associated with a cluster, an unimpeded flight time is calculated as the 10th percentile of the distribution of flight times observed for the members of the cluster. The nominal routes are defined as the reference ideal trajectories with which actual trajectories are compared in order to characterize performance, those nominal routes are known as centroids.

As of fact, four centroids are clearly distinct and represents the ideal trajectories.

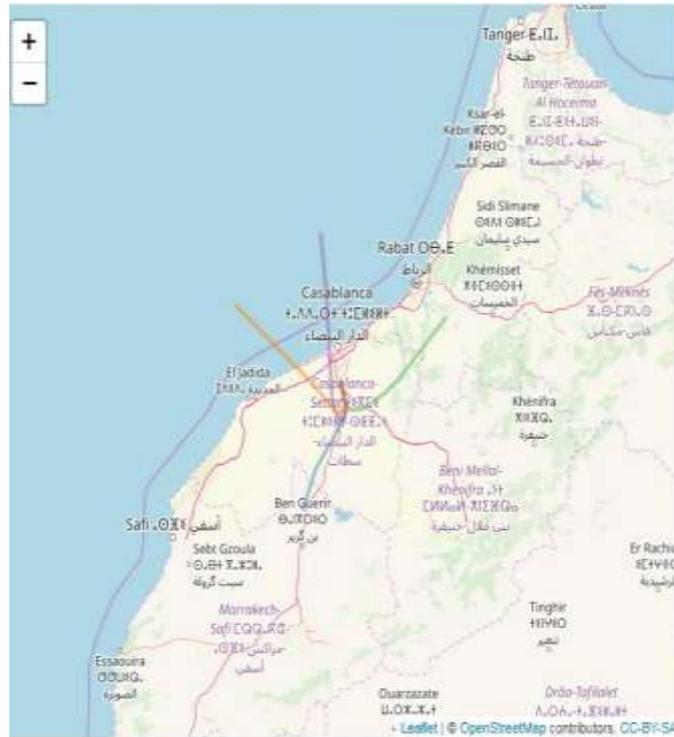


Figure 5 : visualization of centroids

Conclusion

In this article, we used a data-driven approach for characterization of the Casablanca, Morocco airspace structure and air traffic operational performance from aircraft tracking data recorded by surveillance systems. Unsupervised learning is performed associated with a flight trajectory clustering analysis to automatically identify spatial traffic patterns in terminal airspace of Mohammed V, Casablanca airspace. Based on the as-flown route structure learned, quantitative metrics could be developed to describe the structural efficiency of the airspace and the operational efficiency of the traffic flows. For this, actual flight trajectories can be projected onto reference nominal trajectories in space and time. The results allowed above can permit a cross-route comparisons of air traffic flow efficiency across the terminal phase of the flight as well as for the other phases. An interactive data analytics tool is also created to output performance statistics and air traffic visualizations.

References

1. International Air Transport Association. IATA forecasts passenger demand to double over 20 years. <http://www.iata.org/pressroom/pr/Pages/2016-10-18-02.aspx> (12-05-2019).
2. Lee, J. G.; Han, J.; Whang, K. Y.: "Trajectory clustering: a partition-and-group framework". Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 593-604, 2007.
3. Kim, J.; Mahmassani, H. S.: "Spatial and temporal characterization of travel patterns in a traffic network using vehicle trajectories". Transportation Research Part C: Emerging Technologies, Vol. 59, (2015), pp. 375-390.
4. Antonini, G.; Thiran, J. P.: "Counting pedestrians in video sequences using trajectory clustering". IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16 n° 8 (2016), pp. 1008-1020.

5. Eckstein, A.: "Automated flight track taxonomy for measuring benefits from performance-based navigation". Proceedings of the Integrated Communications, Navigation and Surveillance Conference, pp. 1-12, 2009.
6. Gariel, M.; Srivastava, A. N.; Feron, E.: "Trajectory clustering and an application to airspace monitoring". IEEE Transactions on Intelligent Transportation Systems, Vol. 12 n° 4 (2011), pp. 1511-1524.
7. Rehm, F.: "Clustering of flight tracks". Proceedings of the AIAA Infotech@Aerospace Conference, 2010.
8. Enriquez, M.: "Identifying temporally persistent flows in the terminal airspace via spectral clustering". Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar, 2013.
9. Murça, M. C. R.; Hansman, R. J.; Li, L.; Ren, P.: "Flight trajectory data analytics for characterization of air traffic flows: A comparative analysis of terminal area operations between New York, Hong Kong and Sao Paulo". Transportation Research Part C: Emerging Technologies, Vol. 97 (2018), pp. 324-347.