Discovering Communities in Global Flight Route Graph

Team 19

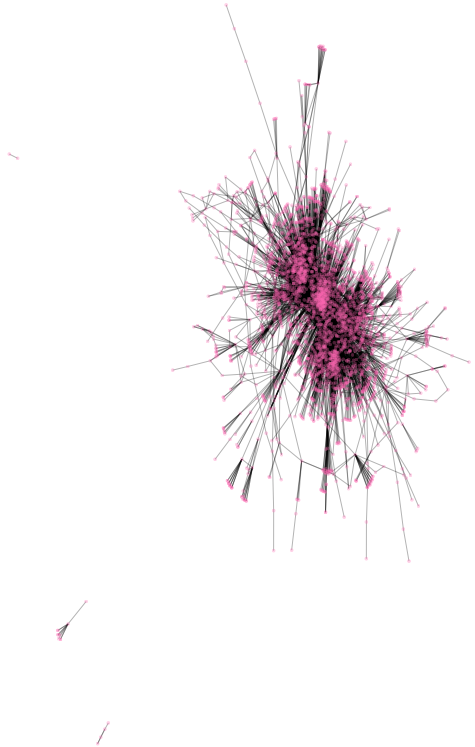
Fengyu Cai

Liangwei Chen

Junze Li

Wanhao Zhou

Introduction



- Open Flights route information
- Community
  - flights concentrated *inside* and relatively sparse towards *outside*
- Goal
  - Discover communities
  - Weakly supervised: clustering with assigned initial centers
  - Unsupervised: community detection

I. Data Acquisition and Preprocessing

II. Graph Analysis and Visualization

III. Learning on Graph

IV. Conclusion

I. Data Acquisition and Preprocessing

I. Data Acquisition and Preprocessing

- Open Flights route database
  - 67663 edges (flights) and 3321 nodes (airports)
- Open Flights airport database
  - Geological information of airport
    - Airport ID, Latitude / Longitude, Time zone information, etc.
- Data cleaning
  - Remove records with unspecified source or destination airport
  - Remove records with unrecorded or wrong airport information
  - 66771 flight routes and 3188 airports

I. Data Acquisition and Preprocessing

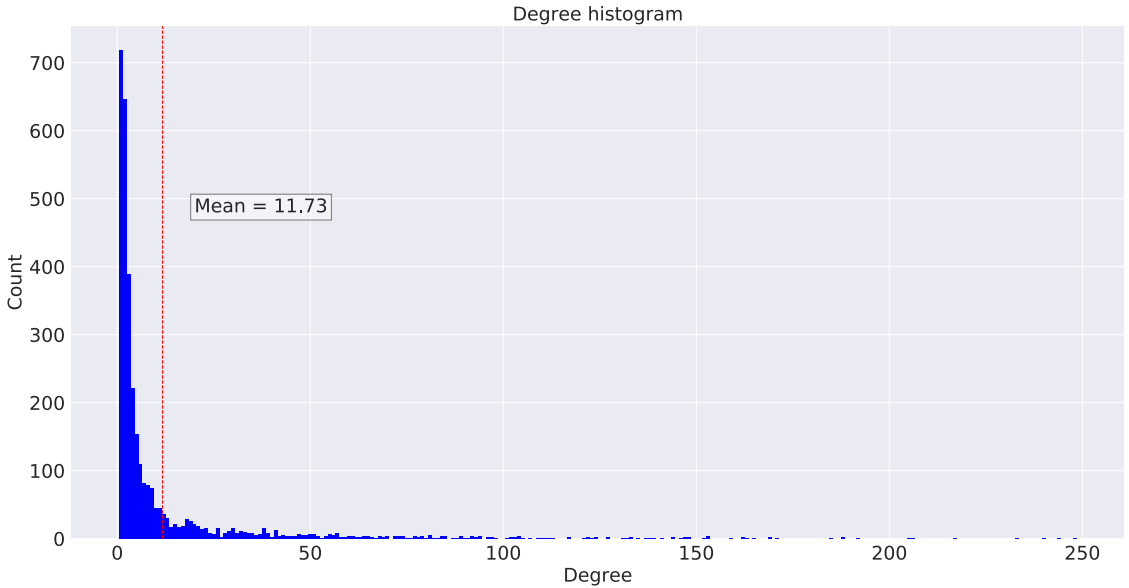
- Adjacency matrix construction
  - Unweighted matrix U
  - Weighted matrix W
    - $W_{i,j} = \exp\left(-\frac{\text{GeodesicDist}_{(i,j)}}{\sigma}\right) \cdot (1 - \delta_{i,j})$
- Adjacency matrices are finally processed to be symmetric

II. Graph Analysis and Visualization

II. Graph Analysis and Visualization

- Basic properties
  - Degree distribution, clustering coefficient, # connected components, etc.
- Centrality
  - Betweenness, PageRank, eigenvector
- Visualization

II. Graph Analysis and Visualization



The distribution typically follows the power law.
It is the so-called scale-free network.
Average degree is about 11.73.

II. Graph Analysis and Visualization

Attribute	Value
Clustering Coefficient	0.49
Giant Component Diameter	12
# Connected Components	7
Average Degree	11.73

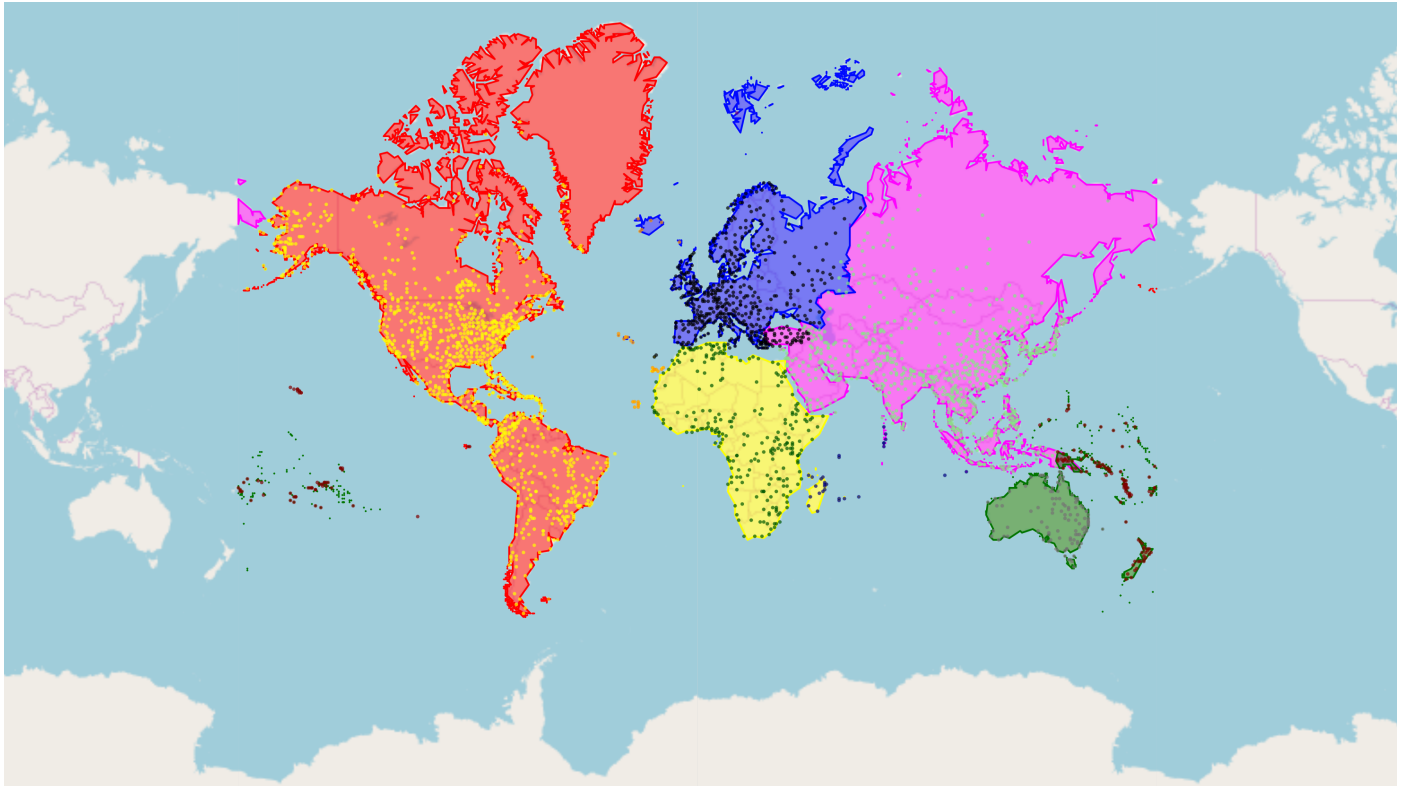
Key properties of the graph.
We proceed with the largest connected component for future analysis.
The largest connected component contains 99.19% of total nodes.

II. Graph Analysis and Visualization

Rank	Betweenness	PageRank	Eigenvector
1	CDG, FR	ATL, US	AMS, NL
2	LAX, US	ORD, US	FRA, DE
3	ANC, US	ISL, TR	CDG, FR
4	DXB, AE	DFW, US	MUC, DE
5	FRA, DE	DEN, US	FCO, IT

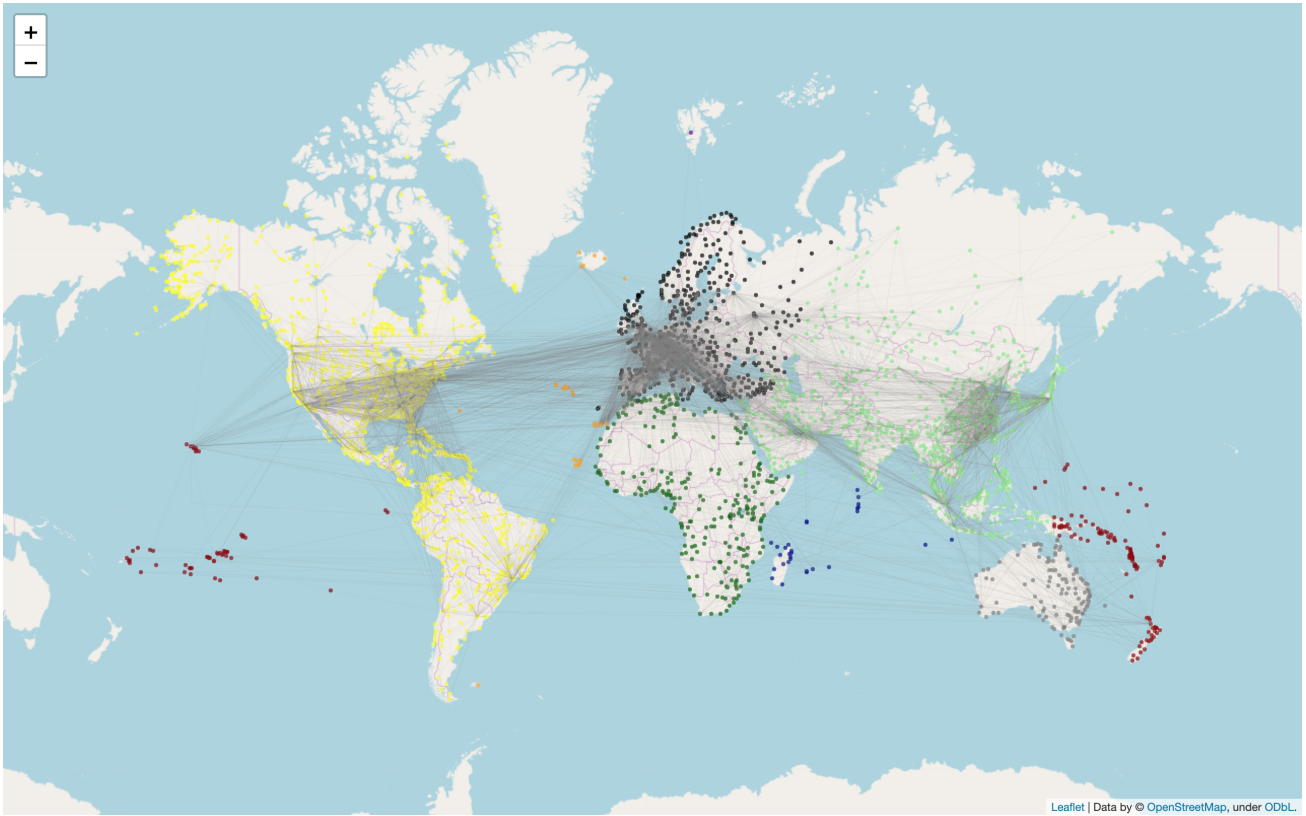
Rank of airports based on different centrality metrics.

II. Graph Analysis and Visualization



Visualization of global airports with borders set by continent.
Airports are marked with different colors based on *time zone.*

II. Graph Analysis and Visualization



Visualization of all flights in the dataset (see Appendix).
Airports are marked with different colors based on *time zone.*

III. Learning on Graph

III. Learning on Graph

- Clustering methods
  - Feature extraction
  - Initial assignment
- Community detection methods
  - Greedy modularity maximization
  - Label propagation
- Case study

III. Learning on Graph

- Feature extraction by Laplacian eigenmap
- Laplacian matrix $L = D - A$.
- D represents the degree of each node
- In the normalized cut problem, we solve the vector u such that $Lu = \lambda Du$.
- Perform the eigenvector decomposition on the normalized Laplacian matrix $L_n = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$.
- Suppose $L_n v_k = \lambda_k v_k$, we obtain $u_k = D^{-\frac{1}{2}} v_k$, associated with λ_k.

III. Learning on Graph

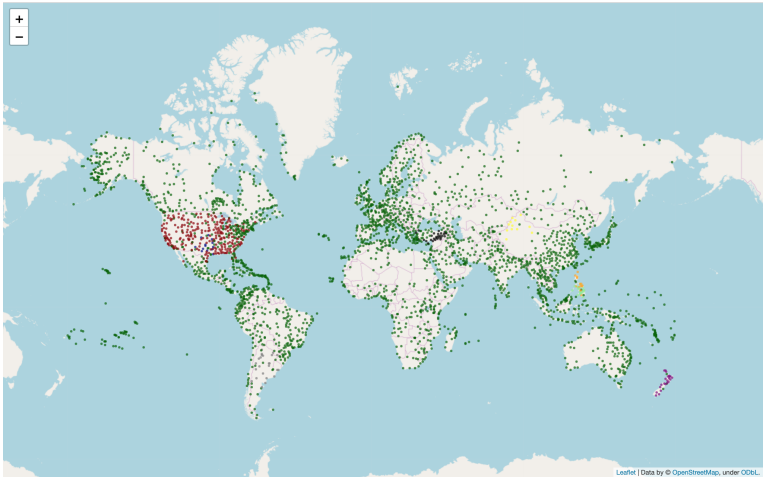
- Results are heavily susceptible to the initial assignment.
- Solution: utilize the time zone information.
- Assumption: 9 clusters, belonging to each continent.
- Assignment: initial centers chosen as the topmost ranked airport in each continent by centrality metrics.

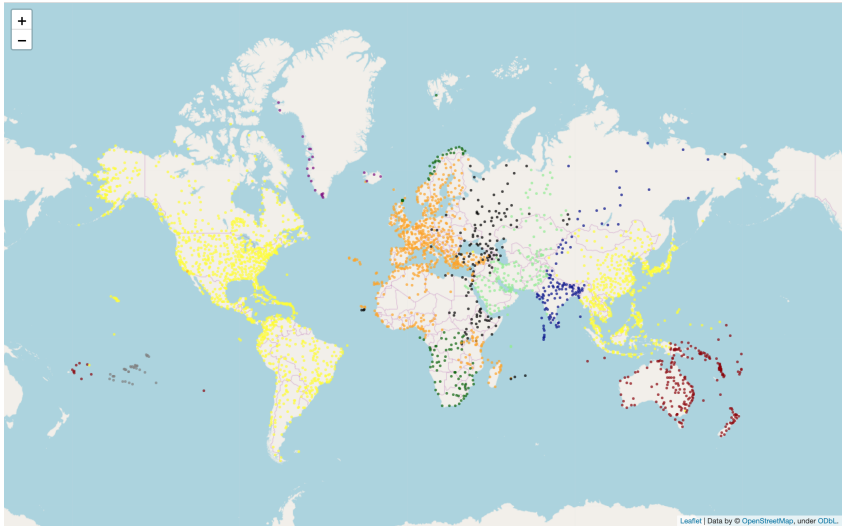
III. Learning on Graph

CONTINENT	CITY
AFRICA	JOHANNESBURG
AMERICA	CHICAGO
ARCTIC	SVALBARD
ASIA	DUBAI
ATLANTIC	GRAN CANARIA
AUSTRALIA	SYDNEY
EUROPE	PARIS
INDIAN	MALE
PACIFIC	HONOLULU

Centers chosen for each continent by aggregating different centrality rankings.

III. Learning on Graph





Left: random initial assignment.
Right: initial assignment with centers chosen by centrality metrics.

III. Learning on Graph

- Clustering needs to some extent supervision.

- Proceed with community detection methods, which depends purely on the network structure.
  - Greedy modularity maximization
  - Label propagation

III. Learning on Graph

- Modularity :

$$Q = \frac{1}{2|E|} \sum_{ij} \left[A_{ij} - \frac{d_i d_j}{2|E|} \right] \delta_{c_i, c_j}$$

$$= \sum_{c_i \in C} \left[\frac{\left| E_{c_i}^{in} \right|}{|E|} - \left(\frac{2\left| E_{c_i}^{in} \right| + \left| E_{c_i}^{out} \right|}{2|E|} \right)^2 \right]$$

- Modularity favors a community that has more edges within the same community than expected number of edges while preserving the degrees of the vertices.
- Use Greedy modularity maximization (Chen et al., 2014) to solve.

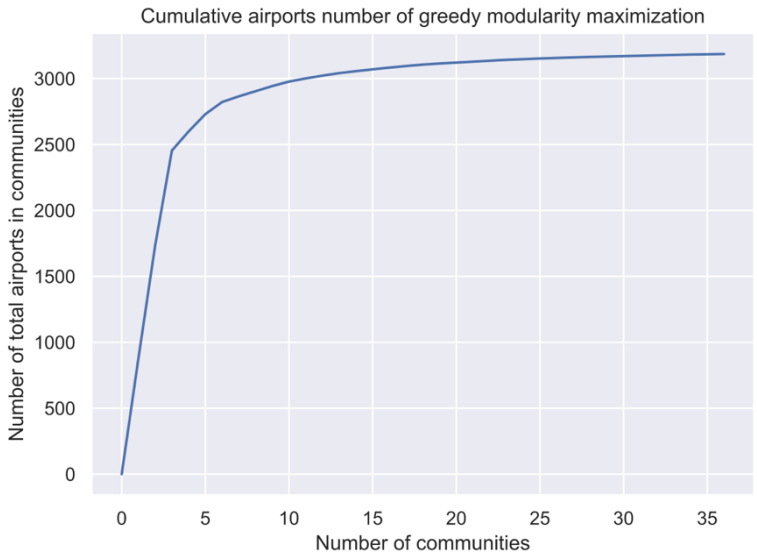
III. Learning on Graph

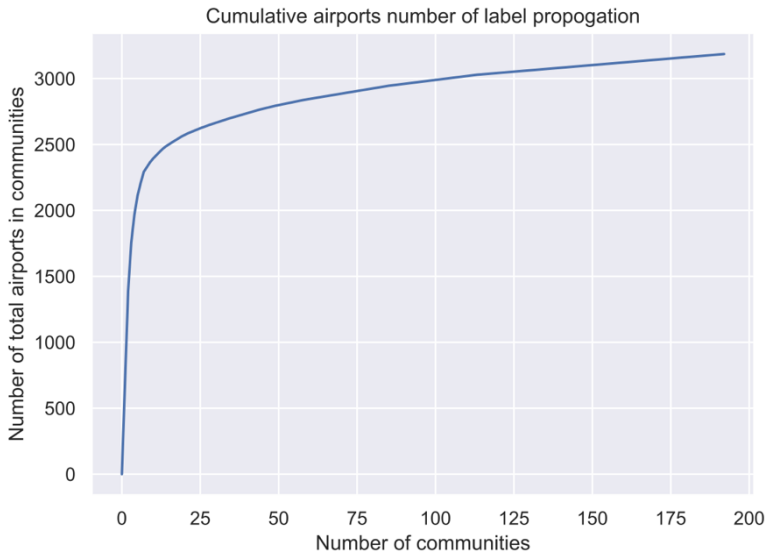
- Label propagation (Raghavan et al., 2007)
- Starts with each node representing a different label.
- Propagation step:

$$L_{v_i}(t+1) = f\left(L_{v_{i1}}(t), \ldots, L_{v_{im}}(t), \ldots, L_{v_{ik}}(t)\right)$$

- Ends when the label of each node is the majority.

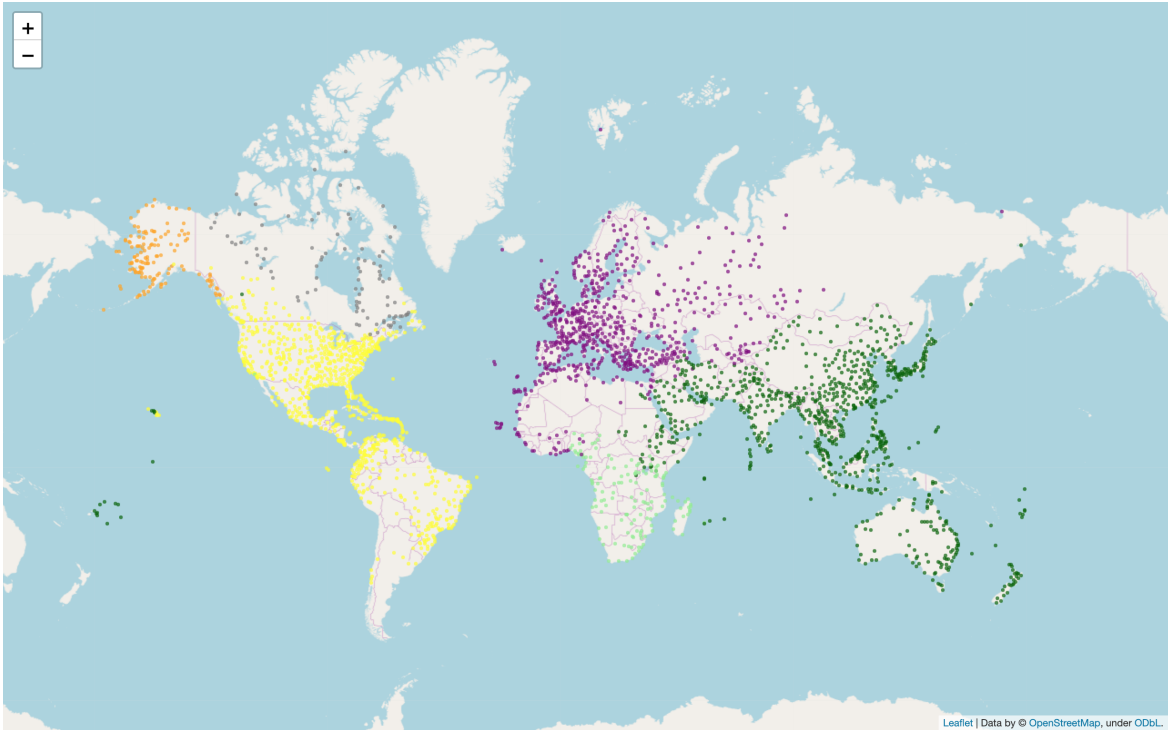
III. Learning on Graph





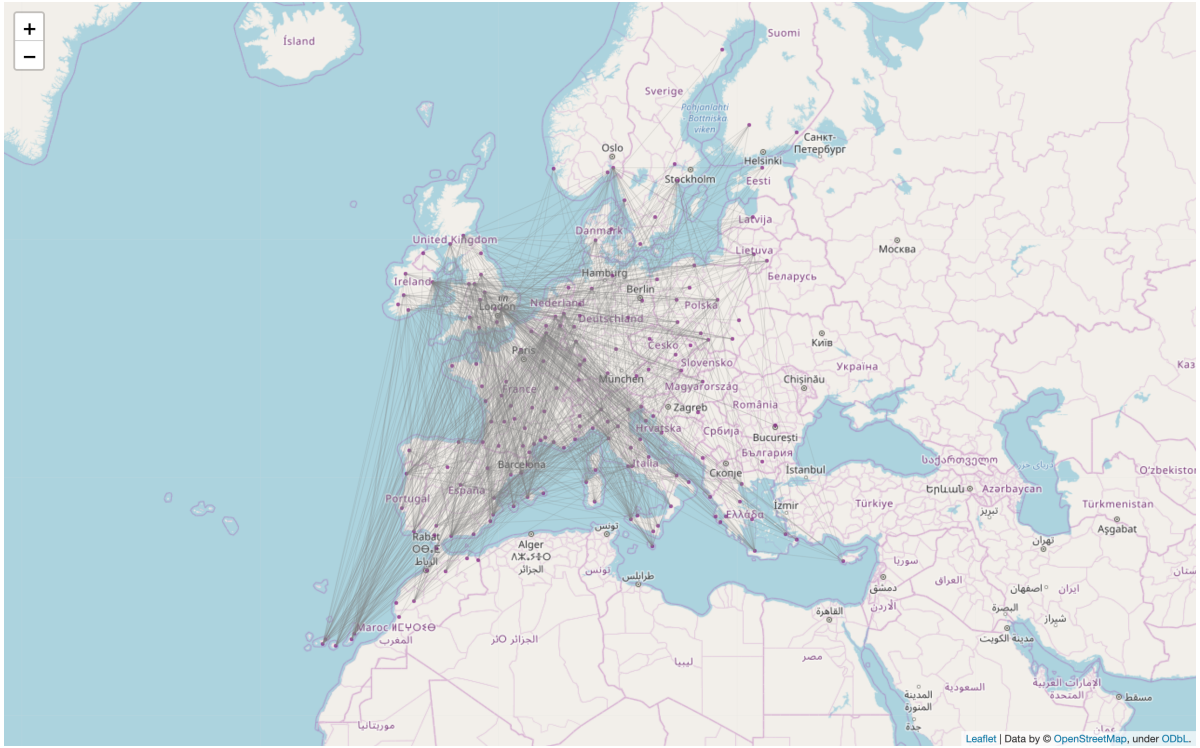
Cumulative number of airports in communities.
We need smaller number of communities to represent more airports. Greedy modularity maximization is better in this sense.

III. Learning on Graph



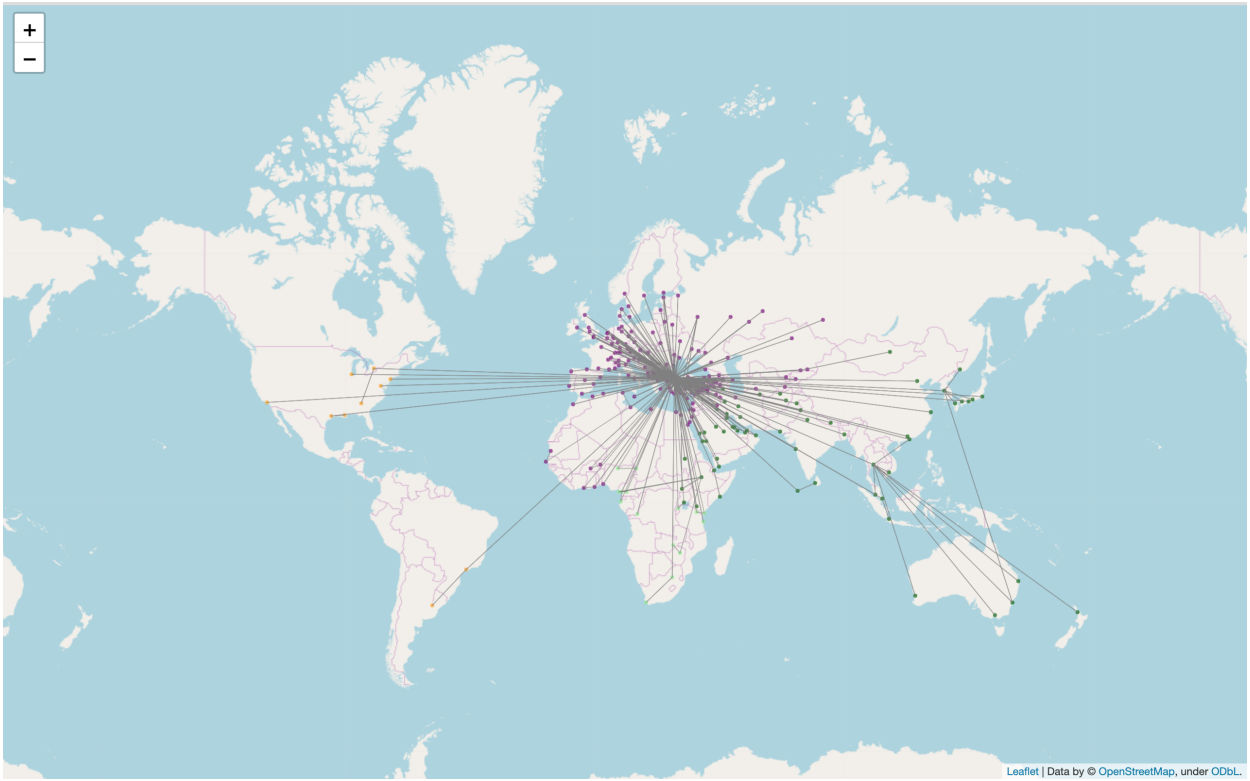
Communities discovered by greedy modularity maximization.

III. Learning on Graph



Ryan Airlines, a budget airline. Flights are all within one discovered community (labelled as purple dots)

III. Learning on Graph



Turkey Airlines interconnects many communities (denoted in different colors) as it is more international and has its center near the Istanbul airport.

IV. Conclusion

IV. Conclusion

- Explore the global flight route network to figure out potential communities among the airports.

- Steps
  - Build up the network, analyze its properties, choose the biggest component
  - Cluster with unweighted and weighted matrix
    - Use Laplacian eigenmap on weighted matrix *with careful initial assignment*
  - Finding communities automatically
    - Analyze results with cases

IV. Conclusion

- Collection of all intermediate results visualized on the global map is available here:

https://wanhaozhou.github.io/docs/display.html

References

- Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

- Chen, M., Kuzmin, K., and Szymanski, B. K. Community detection via maximization of modularity and its variants. *IEEE Transactions on Computational Social Systems*, 1 (1):46–65, 2014.

- Raghavan, U. N., Albert, R., and Kumara, S. Near linear time algorithm to detect community structures in largescale networks. *Physical review E*, 76(3):036106, 2007