Naser Salas

Professor Kontothanassis

CDS DS210

05/04/2023

DS210 Final Project Report

Topic and explanation:

Hypothesis: Airports in the United States increase their overall centrality measures over time.

For this project, I chose to use a publicly available dataset of US airports and their connections. I thought this would be a good dataset to illustrate networks and their connections. I chose three different years, 1998, 2003, and 2008 to analyze six different airports while comparing their centrality over time. The first airport is Champaign's Willard Airport (CMI), my home airport that only flies from Champaign to Dallas Fort Worth and Champaign to Chicago. The second airport was Chicago (ORD), a major airport in Illinois and the United States. I also studied an additional four airports that are spread out across different sections of the United States. Atlanta (ATL), the busiest airport in the United States. Portland (PDX) because it is along the west coast, Fort Lauderdale (FLL) because it is in the south, and then Washington D.C. (DCA) because it is a midpoint between the North and South parts of the United States. The original data set consisted of the origin airport, the destination airport, the origin city, destination city, the number of times that the specific flight took place, the number of passengers, the number of flights, distance, fly date, origin population, destination population, origin airport latitude, destination airport latitude, origin airport longitude, and destination airport longitude. I then decided to trim this data set down to only the essential features to find centrality. This included the origin airport code, destination airport code, number of flights, number of passengers, number of seats, number of times flown, distance, origin population, and destination population. I then made six different CSV files based on the Illinois and United States airports overall, with the Illinois CSV file acting as a subgraph to the United States CSV file. I started by reading the CSV file in Rust, importing each feature as a float. I also created a hashmap of the unique airport codes and population. If an element was not in the hash, it would not have been added to the hashmap of the unique airport (codes).

Furthermore, I used distance for the weight of each node. However, it made no difference when comparing the weight we used in calculating centrality. I also added the airport code as the edges, meaning that the number of edges was the total number of airports used. In order to compare the centrality between the data sets, I implemented four different algorithms in Rust to compute the centrality of each airport over time, testing my initial hypothesis.

Data set you picked

File name: USA Airport Dataset

Website:
https://www.kaggle.com/datasets/flashgordon/usa-airport-dataset

Data set:

This was modified into 6 different datasets based on the year and type of data.

1. 1998IL.csv
2. 1998USA.csv
3. 2003IL.csv
4. 2003USA.csv
5. 2008IL.csv
6. 2008USA.csv

I Extracted data from it to make the point for the state of Illinois and the entire USA but only took into account where the flight had passengers greater than 4.

Data
1998
-> USA
-> IL

USA 1998

```
create table USA1998 as
 select
   Origin_airport
 , Destination_airport
 , count(*) as Numtimes
 , sum(Passengers) as Passengers
 , sum(Seats) as Seats
 , sum(Flights) as Flights
 , max(Distance) as Distance
 , max(Origin_population) as Origin_population
 , max(Destination_population) as Destination_population
from `airports` where year(fly_date) = 1998 and Passengers > 4 group by origin_airport,
destination_airport;
```

Illinois 1998

```
create table IL1998 as
 select
   Origin_airport
 , Destination_airport
 , count(*) as Numtimes
 , sum(Passengers) as Passengers
 , sum(Seats) as Seats
 , sum(Flights) as Flights
 , max(Distance) as Distance
 , max(Origin_population) as Origin_population
 , max(Destination_population) as Destination_population
from `airports` where year(fly_date) = 1998 and trim(substring_index(Origin_city,',',-1))
in ('IL') and Passengers > 4 group by origin_airport, destination_airport;
```

2003

-> USA
-> IL

USA 2003

```
create table USA2003 as
 select
   Origin_airport
 , Destination_airport
 , count(*) as Numtimes
 , sum(Passengers) as Passengers
 , sum(Seats) as Seats
 , sum(Flights) as Flights
 , max(Distance) as Distance
 , max(Origin_population) as Origin_population
 , max(Destination_population) as Destination_population
from `airports` where year(fly_date) = 2003 and Passengers > 4 group by origin_airport,
destination_airport;
```

Illinois 2003

```
create table IL2003 as
 select
   Origin_airport
 , Destination_airport
 , count(*) as Numtimes
 , sum(Passengers) as Passengers
 , sum(Seats) as Seats
 , sum(Flights) as Flights
 , max(Distance) as Distance
 , max(Origin_population) as Origin_population
 , max(Destination_population) as Destination_population
from `airports` where year(fly_date) = 2003 and trim(substring_index(Origin_city,',',-1))
in ('IL') and Passengers > 4 group by origin_airport, destination_airport;
```

2008

-> USA
-> IL

USA 2008

```
create table USA2008 as
 select
   Origin_airport
 , Destination_airport
 , count(*) as Numtimes
 , sum(Passengers) as Passengers
 , sum(Seats) as Seats
 , sum(Flights) as Flights
 , max(Distance) as Distance
 , max(Origin_population) as Origin_population
 , max(Destination_population) as Destination_population
from `airports` where year(fly_date) = 2008 and Passengers > 4 group by origin_airport,
destination_airport;
```

Illinois 2008

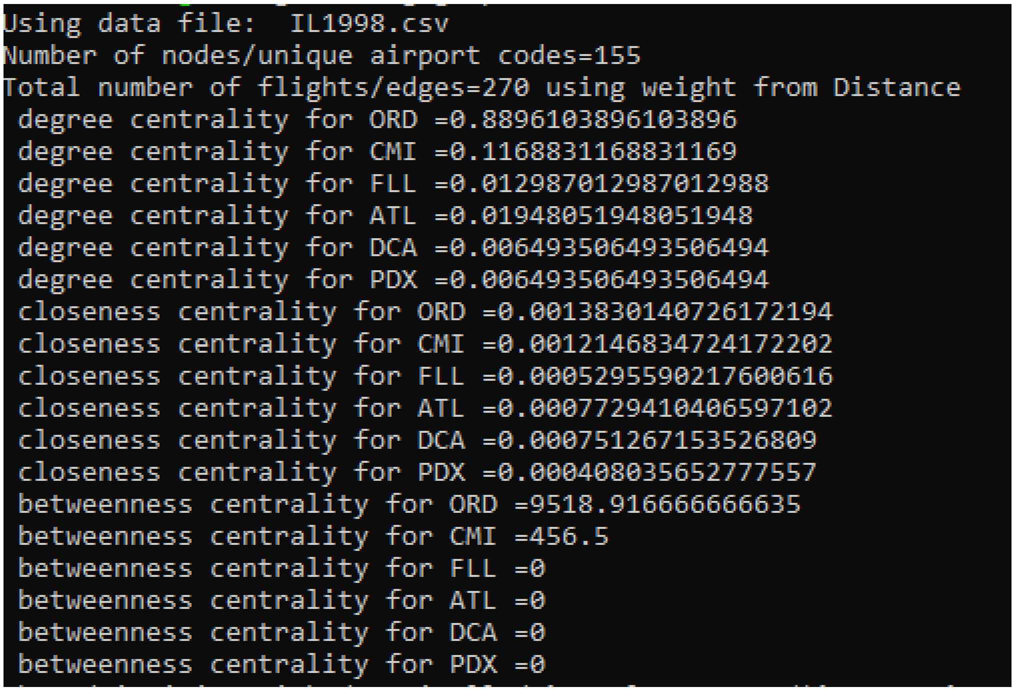
```
create table IL2008 as
 select
   Origin_airport
 , Destination_airport
 , count(*) as Numtimes
 , sum(Passengers) as Passengers
 , sum(Seats) as Seats
 , sum(Flights) as Flights
 , max(Distance) as Distance
 , max(Origin_population) as Origin_population
 , max(Destination_population) as Destination_population
from `airports` where year(fly_date) = 2008 and trim(substring_index(Origin_city,',',-1))
in ('IL') and Passengers > 4 group by origin_airport, destination_airport;
```

What algorithms did you implement?

I implemented four different algorithms on each csv file. The first two algorithms were closeness centrality, and betweenness centrality, the third was eigenvector centrality and the last one was degree centrality. Closeness is a measure of distance towards others in a data set. A low closeness score means that nodes do not have to travel as far in order to network or connect with other nodes, these are usually more central nodes. A larger score means that nodes are less central and usually have to travel much more to connect with each other. Betweenness measures how often a node is in the shortest path between two other nodes in a network. Those nodes with high betweenness centrality score are the most central of the nodes or hubs. Eigenvector centrality measures how much a node influences other nodes. A high eigenvector centrality score explains that that node has a high direct connection to other nodes. By contrast, a low scoring node doesn't affect other nodes nearly as much as a higher one would. Lastly, degree centrality measures the number of connections that each node has in the data set. More central nodes have high degree of centrality while less central nodes have low degree centrality. Only those with a USA csv file were run with the eigenvector algorithm, probably because of the fewer number of nodes in the IL file I could not run the eigenvector algorithm there. This program in Rust was run by a "Cargo run," and followed by a selected CSV file and feature used for the weight in the command prompt. An example of this would read, "cargo run -- USA2008.csv Distance," the instructions were also commented in the Rust file as well.

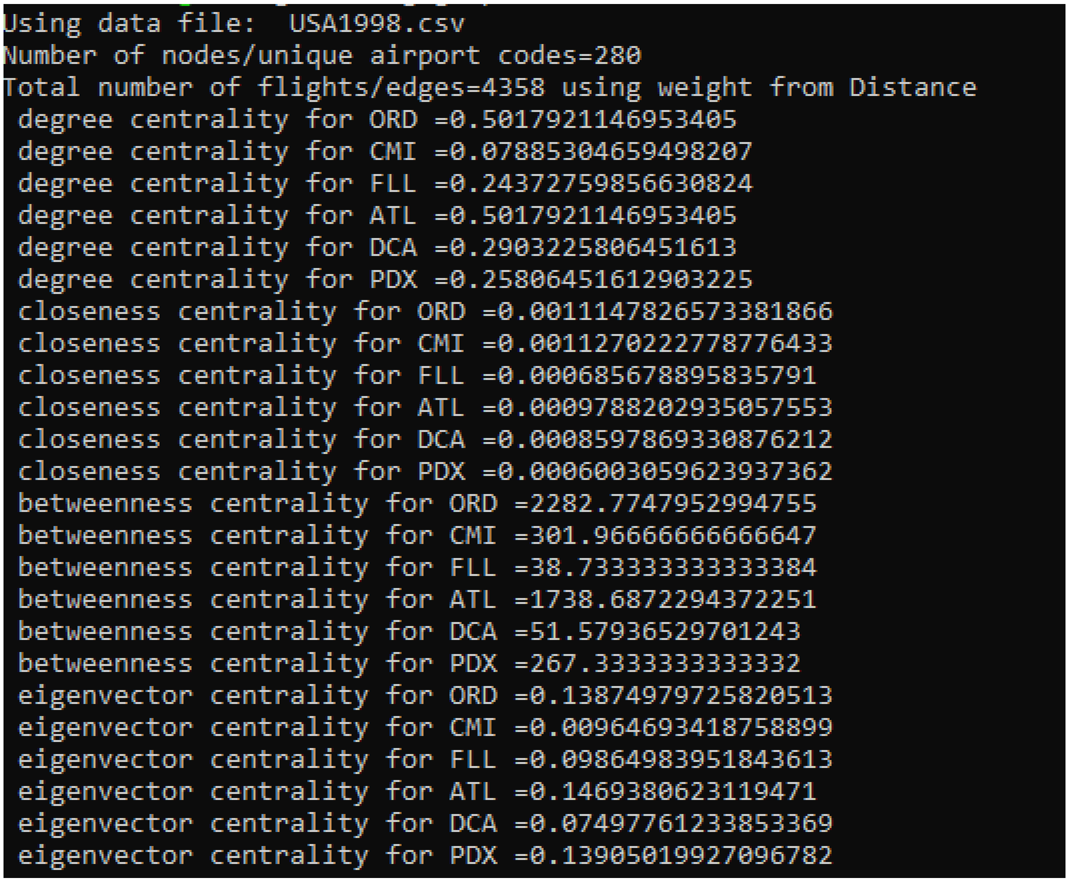
What interesting things did you discover?

1998 Illinois Data set:



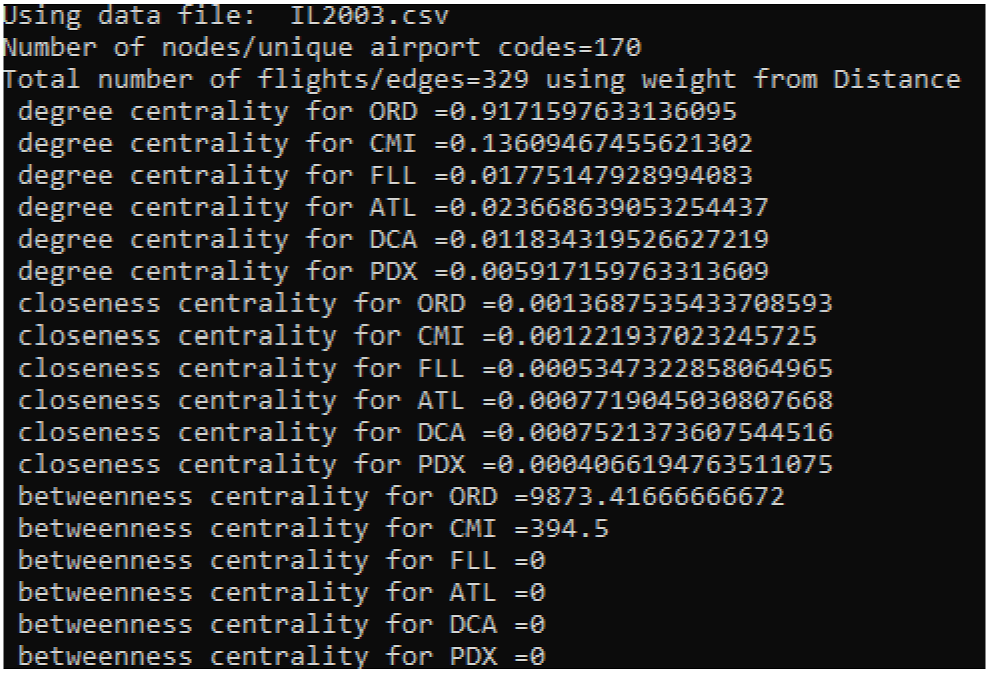
```
Using data file:  IL1998.csv
Number of nodes/unique airport codes=155
Total number of flights/edges=270 using weight from Distance
 degree centrality for ORD =0.8896103896103896
 degree centrality for CMI =0.1168831168831169
 degree centrality for FLL =0.012987012987012988
 degree centrality for ATL =0.01948051948051948
 degree centrality for DCA =0.006493506493506494
 degree centrality for PDX =0.006493506493506494
 closeness centrality for ORD =0.0013830140726172194
 closeness centrality for CMI =0.0012146834724172202
 closeness centrality for FLL =0.0005295590217600616
 closeness centrality for ATL =0.0007729410406597102
 closeness centrality for DCA =0.000751267153526809
 closeness centrality for PDX =0.000408035652777557
 betweenness centrality for ORD =9518.916666666635
 betweenness centrality for CMI =456.5
 betweenness centrality for FLL =0
 betweenness centrality for ATL =0
 betweenness centrality for DCA =0
 betweenness centrality for PDX =0
```

1998 USA Data set:



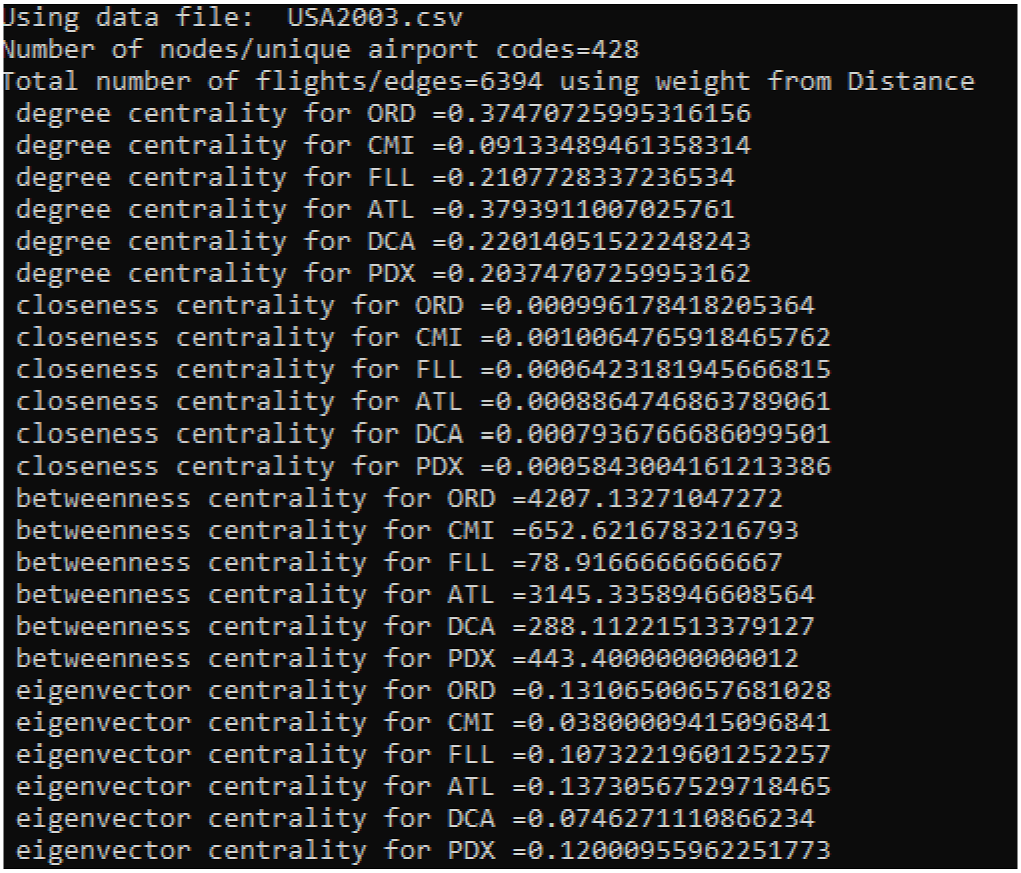
```
Using data file:  USA1998.csv
Number of nodes/unique airport codes=280
Total number of flights/edges=4358 using weight from Distance
 degree centrality for ORD =0.5017921146953405
 degree centrality for CMI =0.07885304659498207
 degree centrality for FLL =0.24372759856630824
 degree centrality for ATL =0.5017921146953405
 degree centrality for DCA =0.2903225806451613
 degree centrality for PDX =0.25806451612903225
 closeness centrality for ORD =0.0011147826573381866
 closeness centrality for CMI =0.0011270222778776433
 closeness centrality for FLL =0.000685678895835791
 closeness centrality for ATL =0.0009788202935057553
 closeness centrality for DCA =0.0008597869330876212
 closeness centrality for PDX =0.0006003059623937362
 betweenness centrality for ORD =2282.7747952994755
 betweenness centrality for CMI =301.96666666666647
 betweenness centrality for FLL =38.733333333333384
 betweenness centrality for ATL =1738.6872294372251
 betweenness centrality for DCA =51.57936529701243
 betweenness centrality for PDX =267.3333333333332
 eigenvector centrality for ORD =0.13874979725820513
 eigenvector centrality for CMI =0.00964693418758899
 eigenvector centrality for FLL =0.09864983951843613
 eigenvector centrality for ATL =0.1469380623119471
 eigenvector centrality for DCA =0.07497761233853369
 eigenvector centrality for PDX =0.13905019927096782
```

2003 Illinois Data set:



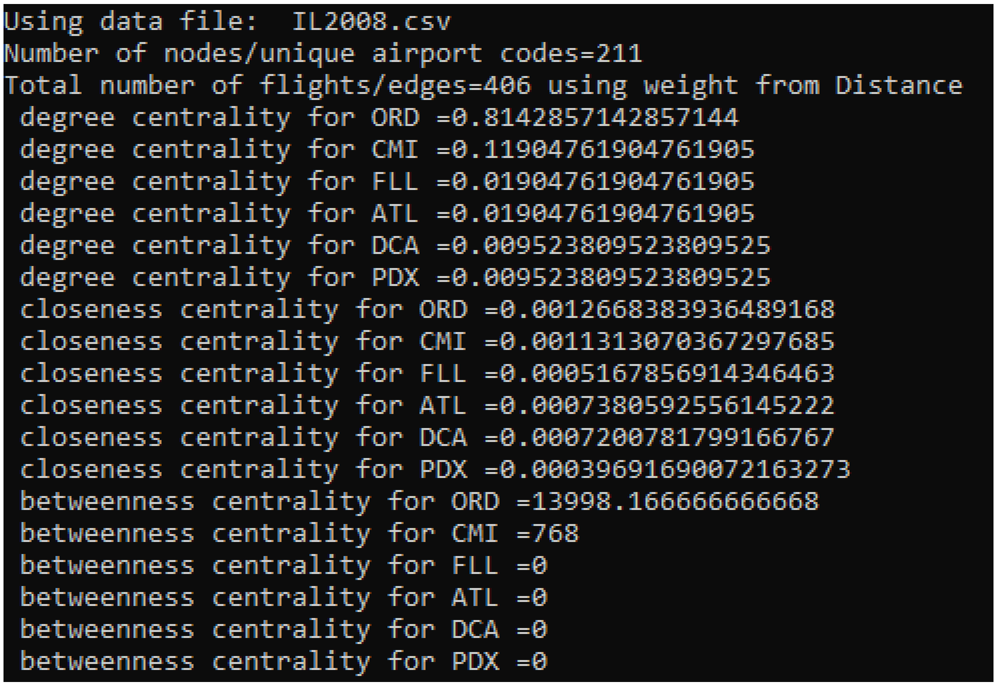
```
Using data file:  IL2003.csv
Number of nodes/unique airport codes=170
Total number of flights/edges=329 using weight from Distance
 degree centrality for ORD =0.9171597633136095
 degree centrality for CMI =0.13609467455621302
 degree centrality for FLL =0.01775147928994083
 degree centrality for ATL =0.023668639053254437
 degree centrality for DCA =0.011834319526627219
 degree centrality for PDX =0.005917159763313609
 closeness centrality for ORD =0.0013687535433708593
 closeness centrality for CMI =0.001221937023245725
 closeness centrality for FLL =0.0005347322858064965
 closeness centrality for ATL =0.0007719045030807668
 closeness centrality for DCA =0.0007521373607544516
 closeness centrality for PDX =0.0004066194763511075
 betweenness centrality for ORD =9873.41666666672
 betweenness centrality for CMI =394.5
 betweenness centrality for FLL =0
 betweenness centrality for ATL =0
 betweenness centrality for DCA =0
 betweenness centrality for PDX =0
```

2003 USA Data set:



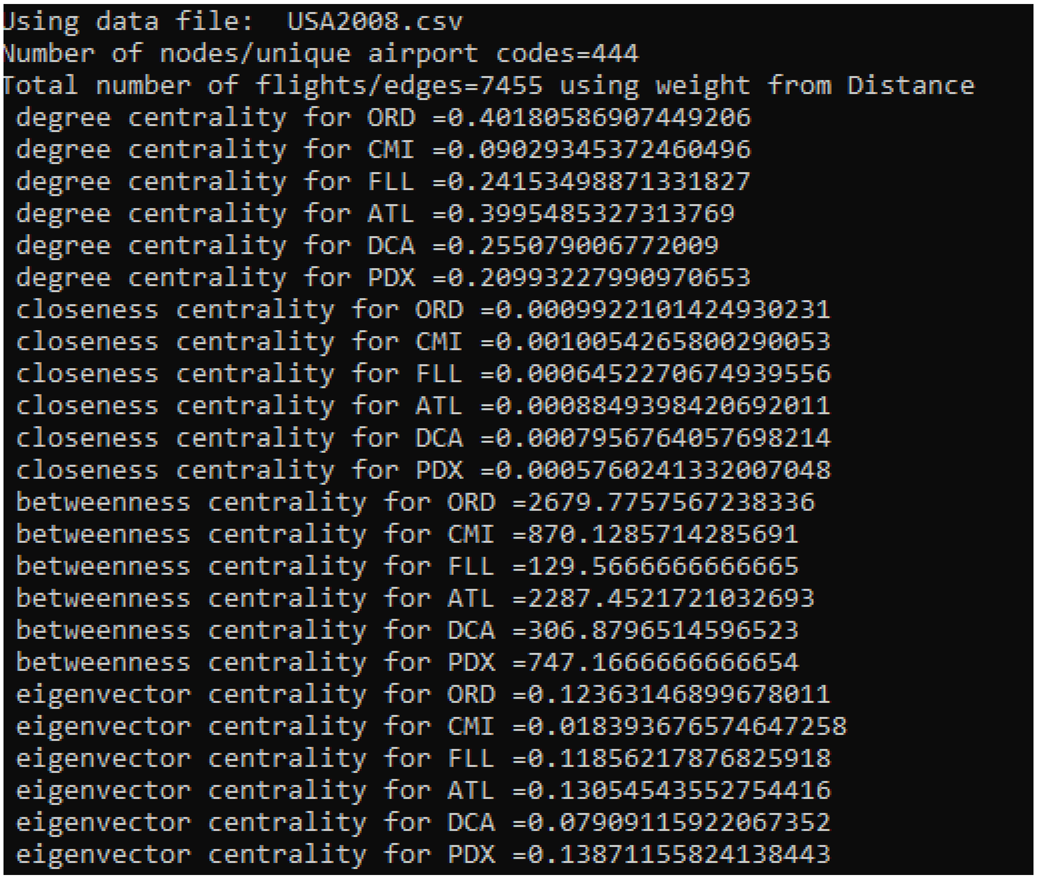
```
Using data file:  USA2003.csv
Number of nodes/unique airport codes=428
Total number of flights/edges=6394 using weight from Distance
 degree centrality for ORD =0.37470725995316156
 degree centrality for CMI =0.09133489461358314
 degree centrality for FLL =0.2107728337236534
 degree centrality for ATL =0.3793911007025761
 degree centrality for DCA =0.22014051522248243
 degree centrality for PDX =0.20374707259953162
 closeness centrality for ORD =0.000996178418205364
 closeness centrality for CMI =0.0010064765918465762
 closeness centrality for FLL =0.0006423181945666815
 closeness centrality for ATL =0.0008864746863789061
 closeness centrality for DCA =0.0007936766686099501
 closeness centrality for PDX =0.0005843004161213386
 betweenness centrality for ORD =4207.13271047272
 betweenness centrality for CMI =652.6216783216793
 betweenness centrality for FLL =78.9166666666667
 betweenness centrality for ATL =3145.3358946608564
 betweenness centrality for DCA =288.11221513379127
 betweenness centrality for PDX =443.4000000000012
 eigenvector centrality for ORD =0.13106500657681028
 eigenvector centrality for CMI =0.03800009415096841
 eigenvector centrality for FLL =0.10732219601252257
 eigenvector centrality for ATL =0.13730567529718465
 eigenvector centrality for DCA =0.0746271110866234
 eigenvector centrality for PDX =0.12000955962251773
```

2008 Illinois Data set:



```
Using data file:  IL2008.csv
Number of nodes/unique airport codes=211
Total number of flights/edges=406 using weight from Distance
 degree centrality for ORD =0.8142857142857144
 degree centrality for CMI =0.11904761904761905
 degree centrality for FLL =0.01904761904761905
 degree centrality for ATL =0.01904761904761905
 degree centrality for DCA =0.009523809523809525
 degree centrality for PDX =0.009523809523809525
 closeness centrality for ORD =0.0012668383936489168
 closeness centrality for CMI =0.0011313070367297685
 closeness centrality for FLL =0.0005167856914346463
 closeness centrality for ATL =0.0007380592556145222
 closeness centrality for DCA =0.0007200781799166767
 closeness centrality for PDX =0.00039691690072163273
 betweenness centrality for ORD =13998.166666666668
 betweenness centrality for CMI =768
 betweenness centrality for FLL =0
 betweenness centrality for ATL =0
 betweenness centrality for DCA =0
 betweenness centrality for PDX =0
```

2008 USA Data set:



```
Using data file:  USA2008.csv
Number of nodes/unique airport codes=444
Total number of flights/edges=7455 using weight from Distance
 degree centrality for ORD =0.40180586907449206
 degree centrality for CMI =0.09029345372460496
 degree centrality for FLL =0.24153498871331827
 degree centrality for ATL =0.3995485327313769
 degree centrality for DCA =0.255079006772009
 degree centrality for PDX =0.20993227990970653
 closeness centrality for ORD =0.0009922101424930231
 closeness centrality for CMI =0.0010054265800290053
 closeness centrality for FLL =0.0006452270674939556
 closeness centrality for ATL =0.0008849398420692011
 closeness centrality for DCA =0.0007956764057698214
 closeness centrality for PDX =0.0005760241332007048
 betweenness centrality for ORD =2679.7757567238336
 betweenness centrality for CMI =870.1285714285691
 betweenness centrality for FLL =129.5666666666665
 betweenness centrality for ATL =2287.4521721032693
 betweenness centrality for DCA =306.8796514596523
 betweenness centrality for PDX =747.1666666666654
 eigenvector centrality for ORD =0.12363146899678011
 eigenvector centrality for CMI =0.018393676574647258
 eigenvector centrality for FLL =0.1185621787682591 8
 eigenvector centrality for ATL =0.13054543552754416
 eigenvector centrality for DCA =0.07909115922067352
 eigenvector centrality for PDX =0.13871155824138443
```

Analysis of Portland (PDX), Washington D.C. (DCA), Atlanta (ATL), and Fort Lauderdale (FLL)

When comparing these four significant airports, Fort Lauderdale decreased its degree of centrality, retained its closeness centrality, increased its betweenness centrality, and its eigenvector centrality over time. Washington D.C. decreased its degree centrality, retained its closeness centrality, increased its betweenness centrality, and retained its eigenvector centrality. Atlanta decreased its degree centrality, retained its closeness centrality, increased its betweenness centrality, and retained its eigenvector centrality score. Lastly, Portland increased its degree centrality score, retained its closeness centrality score, increased its betweenness, and retained its eigenvector centrality.

Overall, only Portland was the airport that improved its centrality, matching the hypothesis while increasing both its degree centrality score and its betweenness score over the ten years while retaining both its eigenvector centrality score and closeness centrality score. The other three airports dealt with at least one decrease in airport centrality among the four measurements. This may be because Portland is smaller than the three other airports, so it has much more room to grow. We can see, though, that Portland did not increase in eigenvector centrality or closeness centrality score; this could be because it has become a connection hub but does not influence other airports in the data set.

Analysis of Champaign and Chicago airport

When comparing the Champaign (CMI) and Chicago (ORD), both decrease their degree of centrality score, retain roughly the same closeness centrality score, and increase their betweenness centrality score, over time, almost following the same pattern. Apart from the other scores where Chicago scores much higher, Champaign and Chicago have roughly the same closeness centrality score. Chicago is indeed one of two destinations that the Champaign airport provides service having a much shorter distance on average than other flights; however, with the Chicago airport being a significant hub across the entire United States that it would still have a much higher centrality score than Champaign with only two destinations that the airport provides service to.

After comparing and analyzing these airports across three different years, we can reject the null hypothesis that the centrality of these airports increased over time.

Anything else you consider relevant

One thing that surprised me was that the eigenvector algorithm only worked for the USA CSV files. This could be because the data set could not compute the eigenvector algorithm for smaller airports or files with fewer nodes. Furthermore, even in the subgraphs, we can still see a degree and closeness centrality score meaning that the airports not located in Illinois still had some connections to Illinois airports, albeit small. However, the betweenness was 0 as they were not located in Illinois.

Resources

1. https://github.com/malcolmvr/graphrs
2. https://visiblenetworklabs.com/2021/04/16/understanding-network-centrality/

3. https://towardsdatascience.com/notes-on-graph-theory-centrality-measurements-e37d2e49550a
4. https://en.wikipedia.org/wiki/Centrality
5.