UNIVERSITY OF NEW HAVEN • CONNECTICUT • 1920

Final Project Report
Distributed & Scalable Data
DSCI6007 – SPRING 2020

Syed Siraj Ul Hasan

Motivation

My project is about Geo-location Clustering. I was interested in real estate and in real estate, location plays a vital role in every aspect. Companies that specialize in mapping and data services customized to the commercial real estate world are taking geolocation to the next level. Geolocation helps retailers — and owners and brokers — better understand the true trade area of a shopping center. Geolocation technology now lets us see the existing traffic patterns of consumers that are shopping at the center. Geolocation data is most effectively used in conjunction with comprehensive market research, sales data, and other tools to assist brokerage firms, retailers, and landlords trying to identify what their trade area is on a daily or weekly basis.

In this project, I am using spark to implement k-means clustering algorithm on Geo-location data.

Documentation Of Approach And Big Data Implementation With Results

Let us start with brief overview and then get in detail of every aspect with results.

Brief Overview:

- Unzipped downloaded datasets, created a S3 bucket named ‘siraj6007’ and upload data onto it.
- Created AWS EMR clusters m4.xlarge, 1 master node and 2 slaves nodes**(System Configuration)**
- Created Notebooks onto AWS EMR clusters and started working on given assignments. Let us see details in the below section.

Detail Working Implementation with Results:

Let’s dive as instructed in the Instructions step by step

PROBLEM 2 →STEP 1

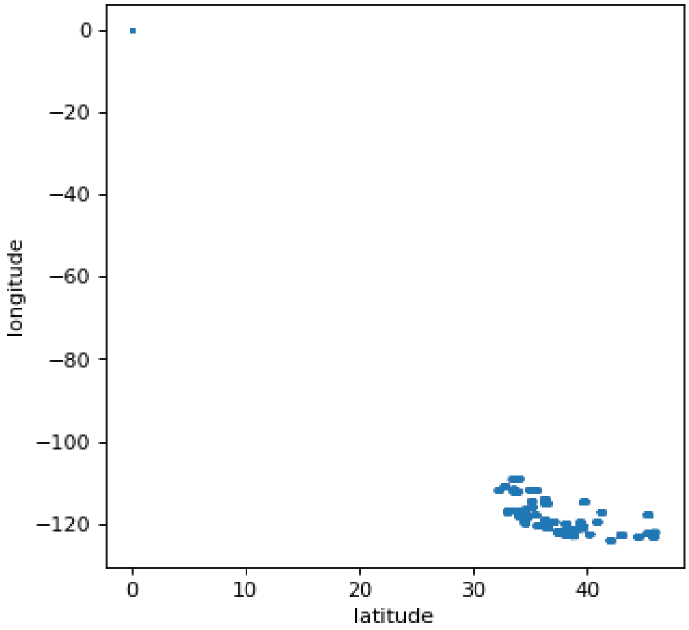
In a notebook on EMR cluster, after importing basic libraries such as pyspark etc. loaded ‘devicestatus.txt’ from the S3 bucket. This datafile required data scrubbing as multiple delimiters were used. Split() was used to pre-process initial data.

Extracted important features such as longitude and latitude from the data. Type casting issues were resolved while extracting features.

Placed that to a panda’s data frame. Now this pre-processed data was stored to bucket for further use.

```
df.rdd.coalesce(1).saveAsTextFile(output_bucket)
```

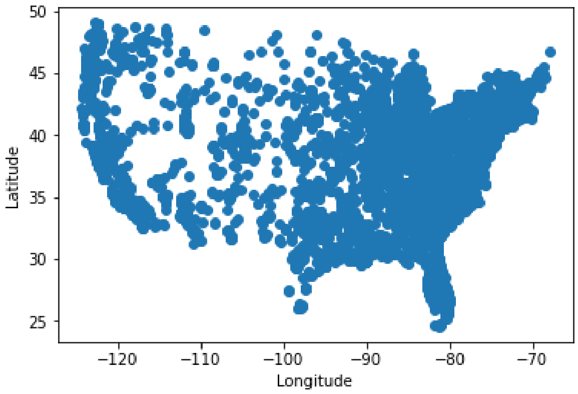
Finally plotted Longitude vs latitude using Matplotlib



PROBLEM 2 →STEP 2

The process was same as PROBLEM 2 →STEP 1 but here 'sample_geo.txt' was loaded from the S3 bucket. Initial preprocessing was required such as strings to numbers, Split() was used to do rest pre-processing. Rest was same.

Finally plotted Longitude vs latitude using Matplotlib.pyplot



PROBLEM 2 →STEP 3

The process was same as above except here we used 'lat_longs.txt'

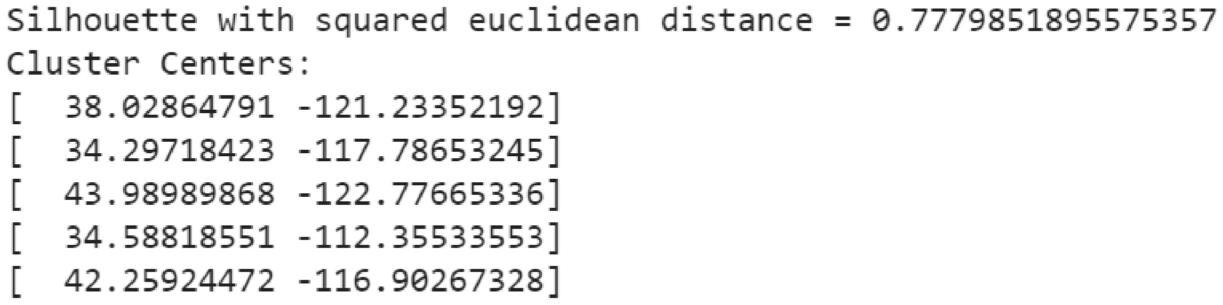
PROBLEM 3

With the resources given in PROBLEM 3 →STEP 1&2, I have built a K-means clustering model with 5 clusters on device location data and used 2 and 4 clusters on synthetic location data. 4 and 6 clusters on lat_longs.txt

Calculated Euclidean Distance and Greater Circular distance in the devicestatus.txt. Predictions done. Snaps are attached below respectively for each data frames.

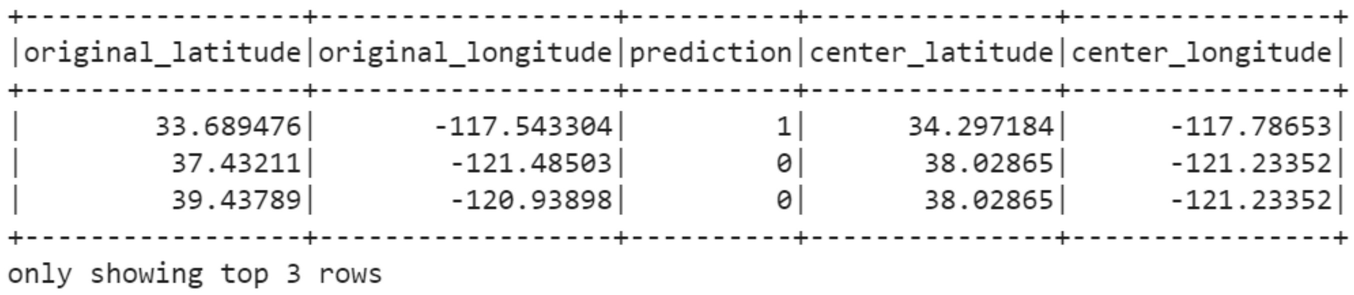
Representation of 5 clusters on devicestatus.txt

Euclidean Distance



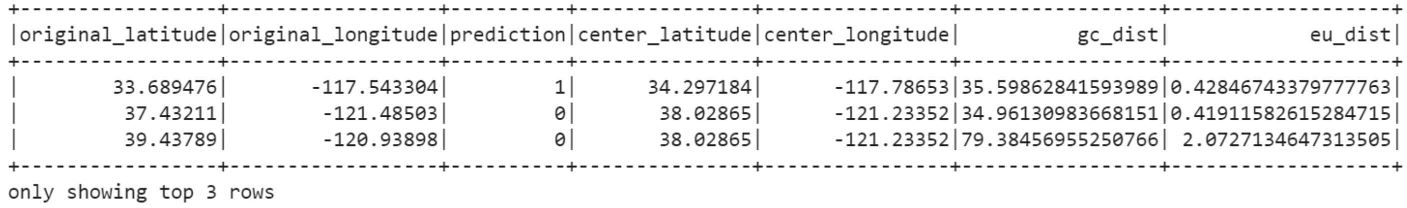
```
Silhouette with squared euclidean distance = 0.7779851895575357
Cluster Centers:
[  38.02864791 -121.23352192]
[  34.29718423 -117.78653245]
[  43.98989868 -122.77665336]
[  34.58818551 -112.35533553]
[  42.25924472 -116.90267328]
```

Predictions



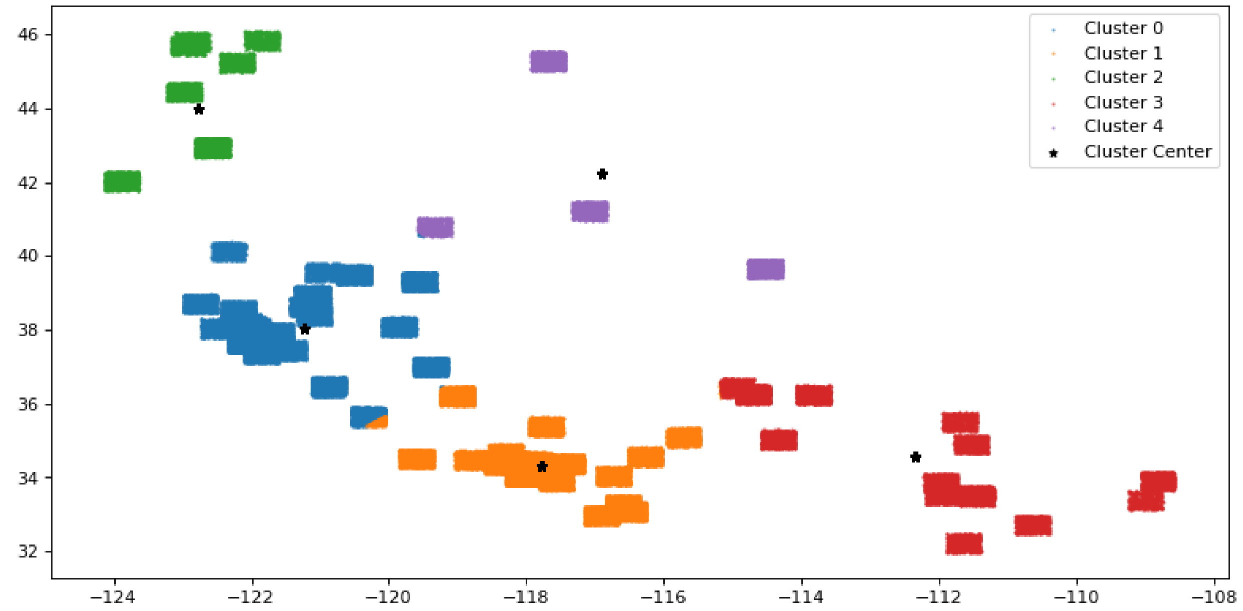
```
+-----------------+------------------+----------+---------------+----------------+
|original_latitude|original_longitude|prediction|center_latitude|center_longitude|
+-----------------+------------------+----------+---------------+----------------+
|        33.689476|       -117.543304|         1|      34.297184|      -117.78653|
|         37.43211|        -121.48503|         0|       38.02865|      -121.23352|
|         39.43789|        -120.93898|         0|       38.02865|      -121.23352|
+-----------------+------------------+----------+---------------+----------------+
only showing top 3 rows
```

Predictions with Greater Circular Distance



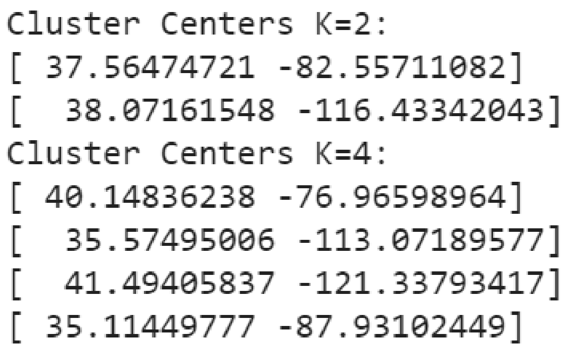
```
+-----------------+------------------+----------+---------------+----------------+-----------------+-------------------+
|original_latitude|original_longitude|prediction|center_latitude|center_longitude|          gc_dist|            eu_dist|
+-----------------+------------------+----------+---------------+----------------+-----------------+-------------------+
|        33.689476|       -117.543304|         1|      34.297184|      -117.78653|35.59862841593989|0.42846743379777763|
|         37.43211|        -121.48503|         0|       38.02865|      -121.23352|34.96130983668151|0.41911582615284715|
|         39.43789|        -120.93898|         0|       38.02865|      -121.23352|79.38456955250766| 2.0727134647313505|
+-----------------+------------------+----------+---------------+----------------+-----------------+-------------------+
only showing top 3 rows
```

5 Cluster Representation



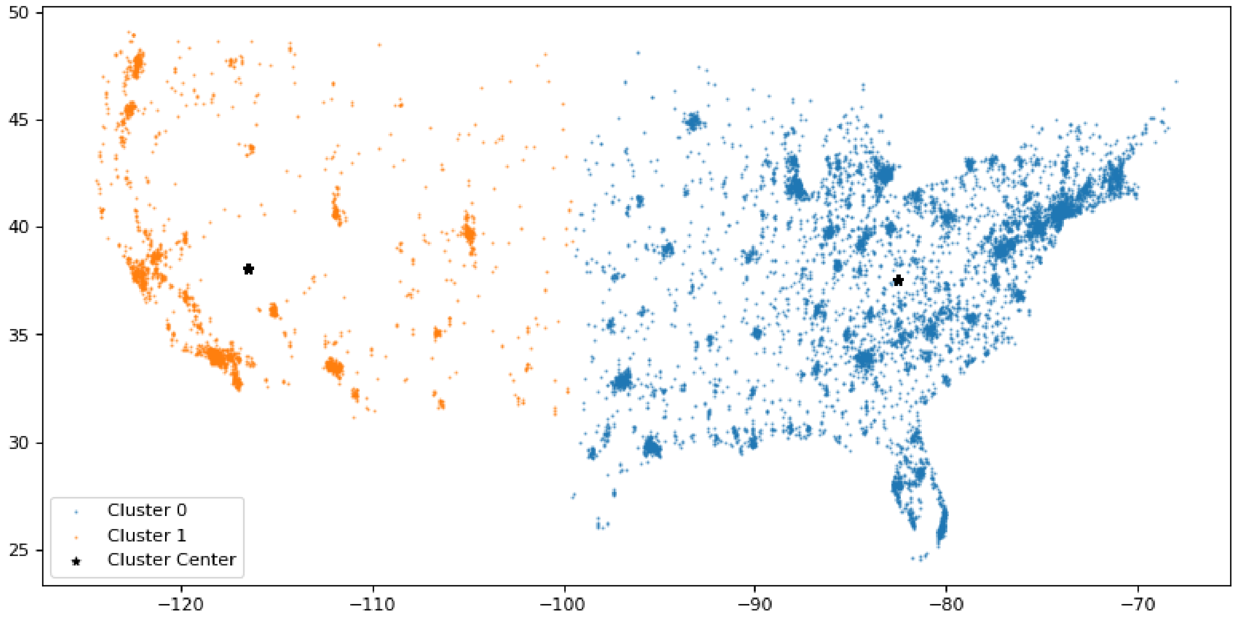
<u>Representation Of 2 &4 Clusters On Sample_Geo.Txt</u>

Clusters Creations

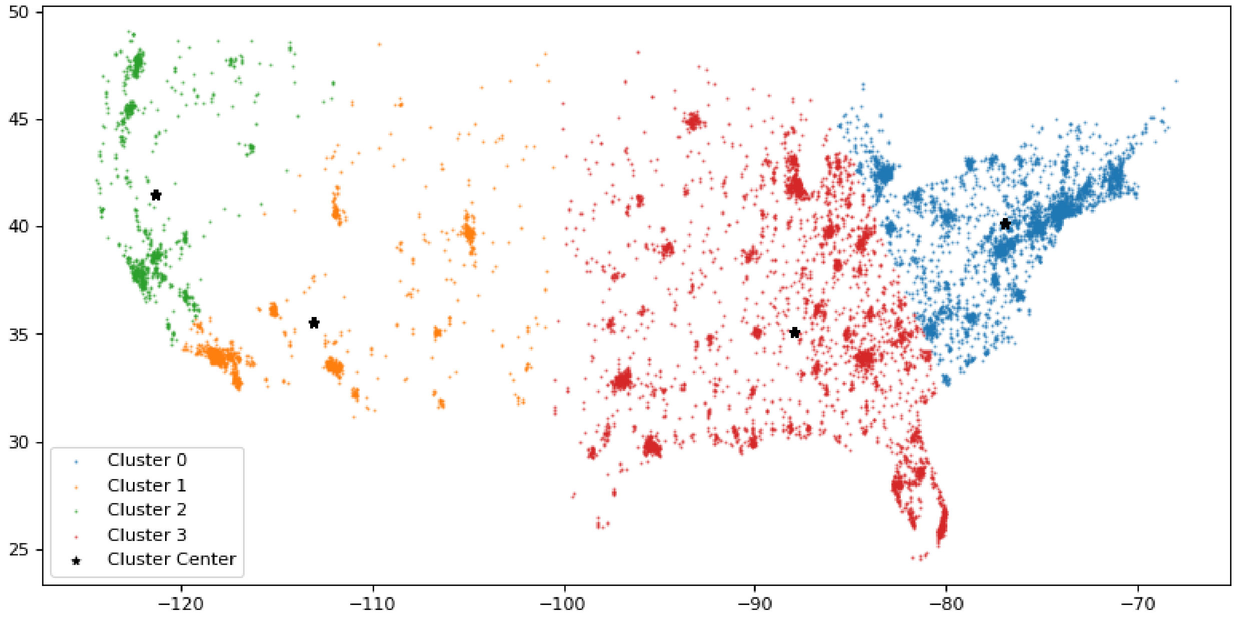


```
Cluster Centers K=2:
[ 37.56474721 -82.55711082]
[  38.07161548 -116.43342043]
Cluster Centers K=4:
[ 40.14836238 -76.96598964]
[  35.57495006 -113.07189577]
[  41.49405837 -121.33793417]
[ 35.11449777 -87.93102449]
```

2 Cluster Representation

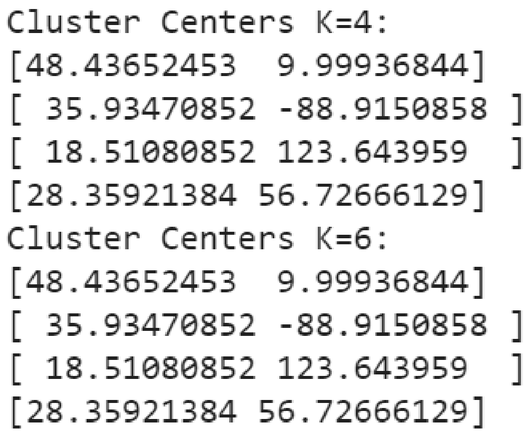


4 Cluster Representation



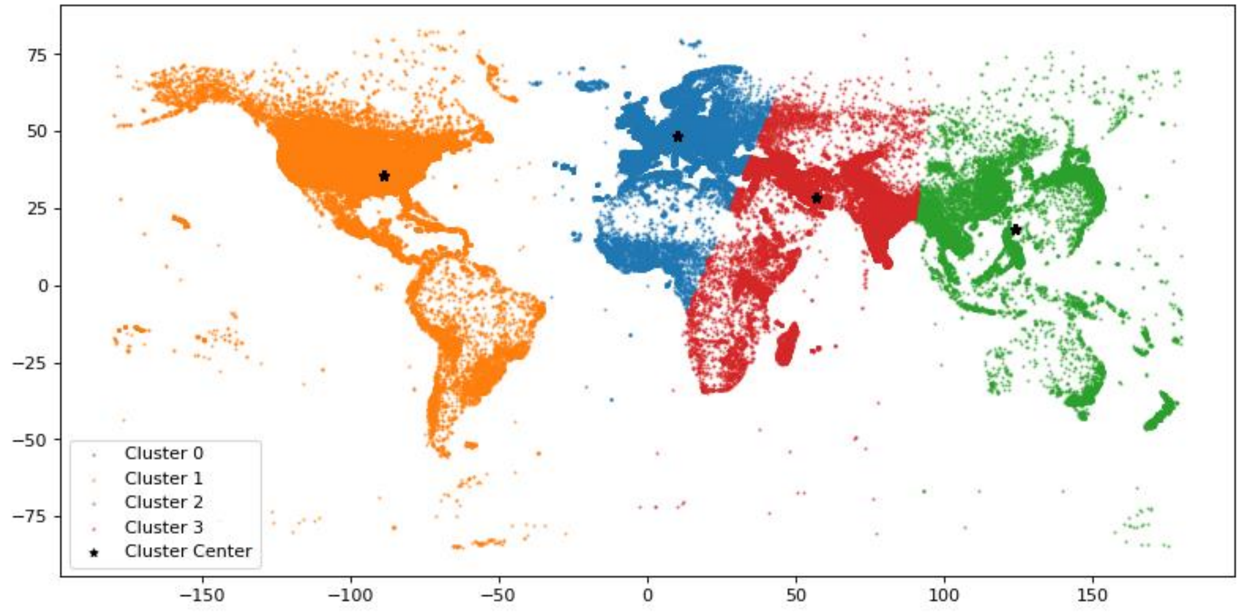
Representation Of 4 &6 lat_longs.txt

Clusters Creation

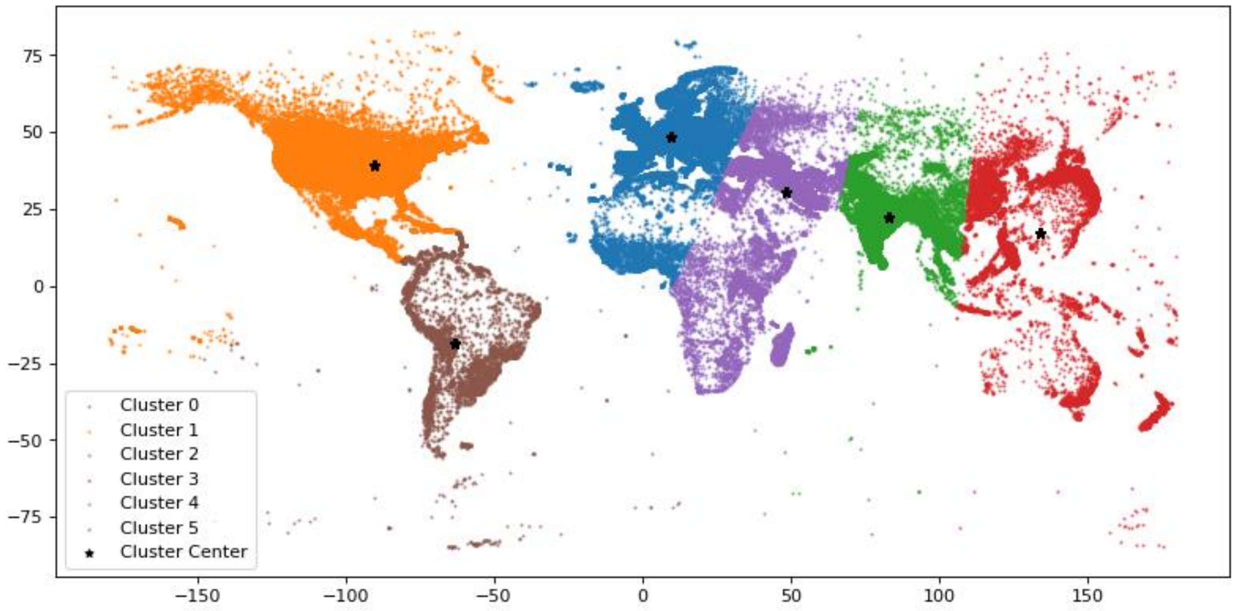


```
Cluster Centers K=4:
[48.43652453  9.99936844]
[ 35.93470852 -88.9150858 ]
[ 18.51080852 123.643959  ]
[28.35921384 56.72666129]
Cluster Centers K=6:
[48.43652453  9.99936844]
[ 35.93470852 -88.9150858 ]
[ 18.51080852 123.643959  ]
[28.35921384 56.72666129]
```

4 Clusters Representation

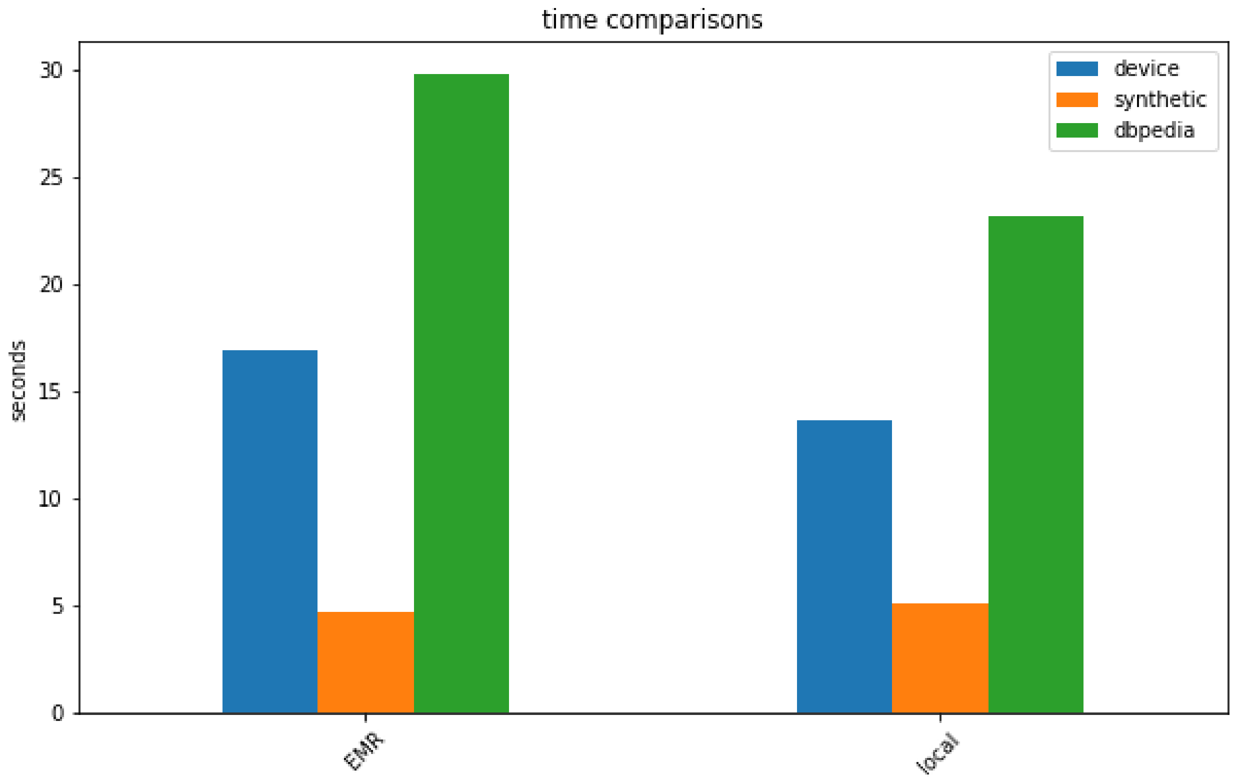


6 Clusters Representation



what could the clusters mean/represent?
Answer to that would be, here our 4 and 6 clusters are trying to represent continents to its best.

PROBLEM 3 → Step 4
Here I did runtime analysis of my K-means implementation for all datasets with local mode with 2 threads and without using persistent RDD. Comparison between those runtimes is depicted in the snap below:



Conclusion

We have Analyzed and understood different service providers and their locations of spread. Which locations are strong and weak. Where weak locations, we can have special generators which can supply at peak times. We also got intuition of clusters and its impact as how it is helping in creation of continents. Lastly Geo data is dependent on what domain the data is.