MOTIVATION:

In data mining, Clustering is considered as an important technique to differentiate various groups of datasets, classes, or clusters within a set of objects. The data objects those are like each other are grouped together to be placed in the same cluster, while the objects different to one another are placed in different groups or clusters. Clustering gives us a clearer picture of the data allowing us to observe each cluster features and then focus on a specific cluster for further analysis.

STEPS:

1. Create an s3 bucket and upload the data (devicestatus.txt, sample_geo.txt, lat_longs.txt) to the bucket.
2. Create an EMR cluster and launch a Jupyter notebook and prepare the notebook to run the latest version of Apache PySpark.
3. Import data individually, clean and pre-process the data and plot a basic graph showing the (latitude, longtitude) pairs.
4. Implement k-means clustering algorithm for the data to get the cluster center points.
5. Calculate the Euclidean Distance and the Greater Circle Distance use the Cluster center coordinates and the actual coordinates from data.
6. Plot the Cluster center coordinates and Actual coordinates.

STEP 1:

We will be using the s3 service provided by Amazon Web Services to create a storage system to upload the files. Upload the data (devicestatus.txt, sample_geo.txt, lat_longs.txt) into the s3 bucket and use the directory of these files to read, write and open the data. This bucket will be used to upload the parsed data i.e., the output cleaned data of devicestatus.txt to the folder “parsed_data”.

STEP 2:

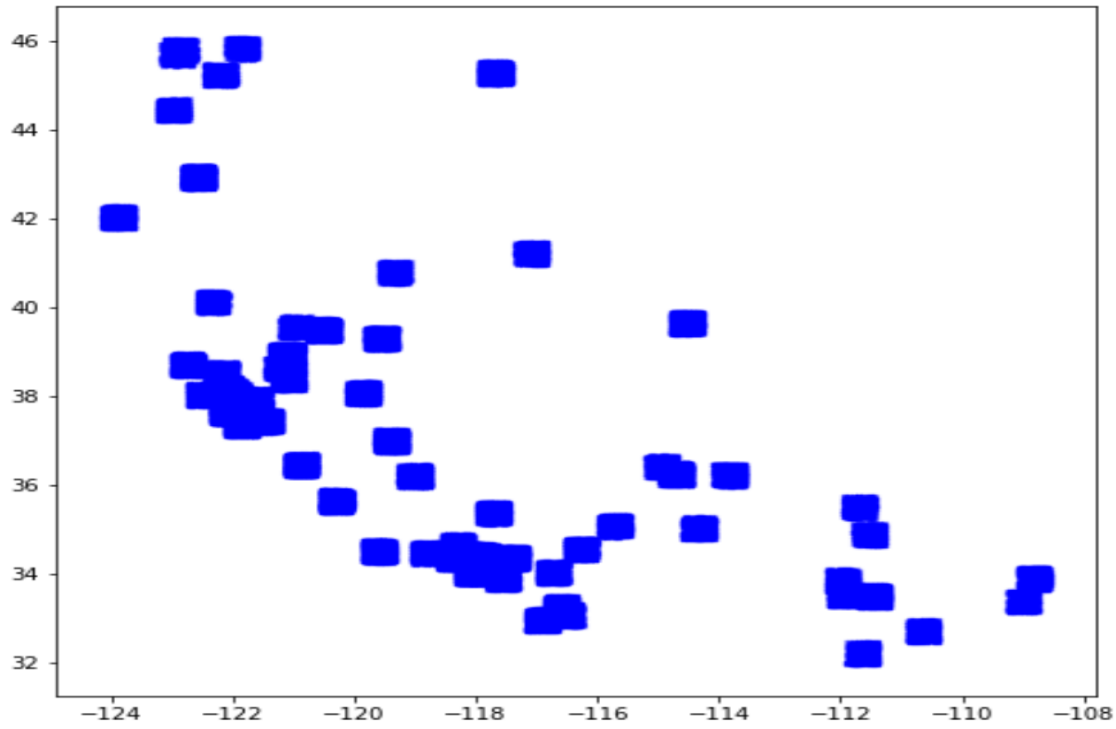
We are creating an EMR cluster service using Amazon Web Services to launch a Jupyter notebook and preparing the notebook to run the latest version of PySpark.

STEP 3:

We will be importing the files using the s3 storage services using the directory which can be found from your bucket. The files contain lot of data which are not relevant to our project so, we need to clean the file and prepare it to create our data frame. The data that we will be needing are primarily the (latitude, longitude) pairs. We need to make sure that the data is clear of delimiters (comma, pipes (|) and so on).

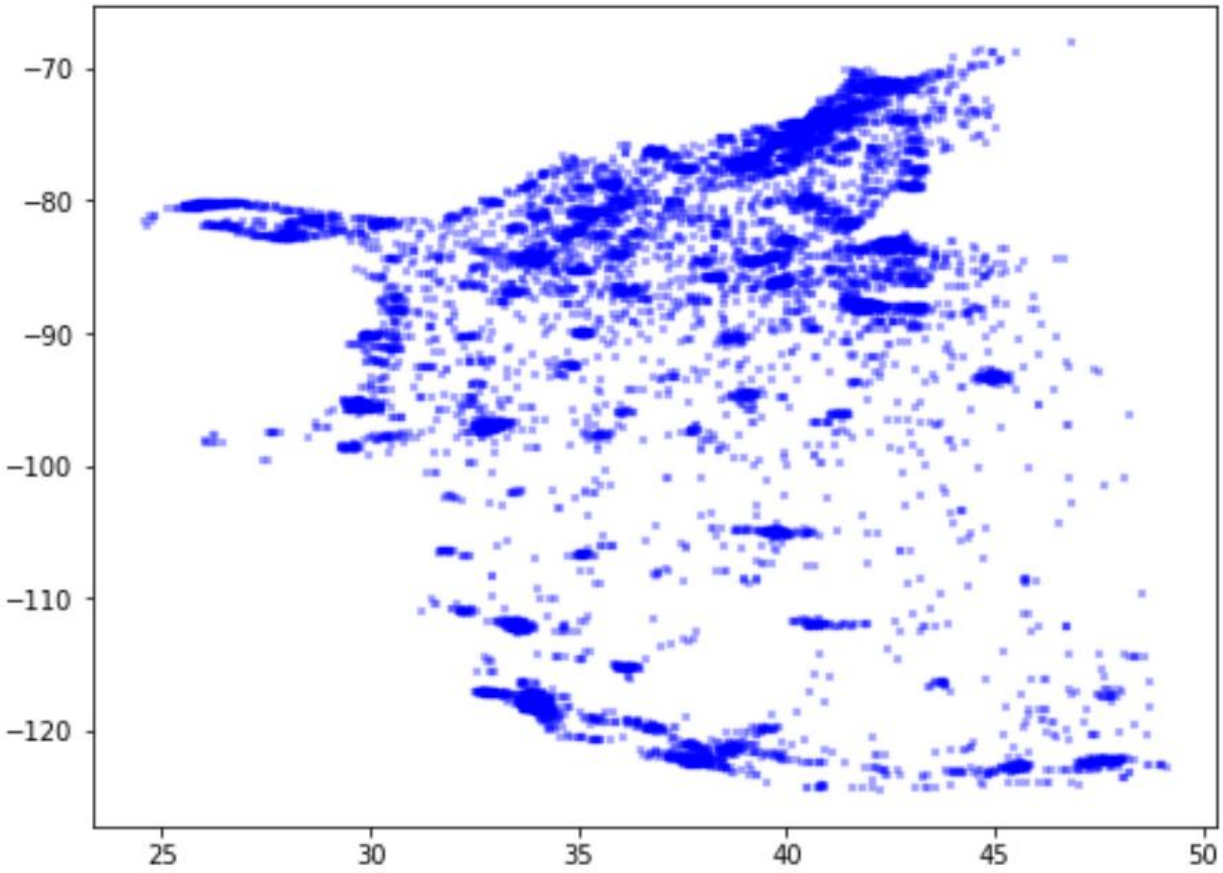
FILE 1:

1. We clean the devicestatus.txt file by eliminating comma (,), pipes(|), text (except the Manufacturer and Model) having the (latitude, longitude) pairs, date, manufacturer and Model ID.
2. Then we create a pyspark data frame forming new columns accordingly.
3. We plot the (latitude, longitude) pairs using data visualization library.



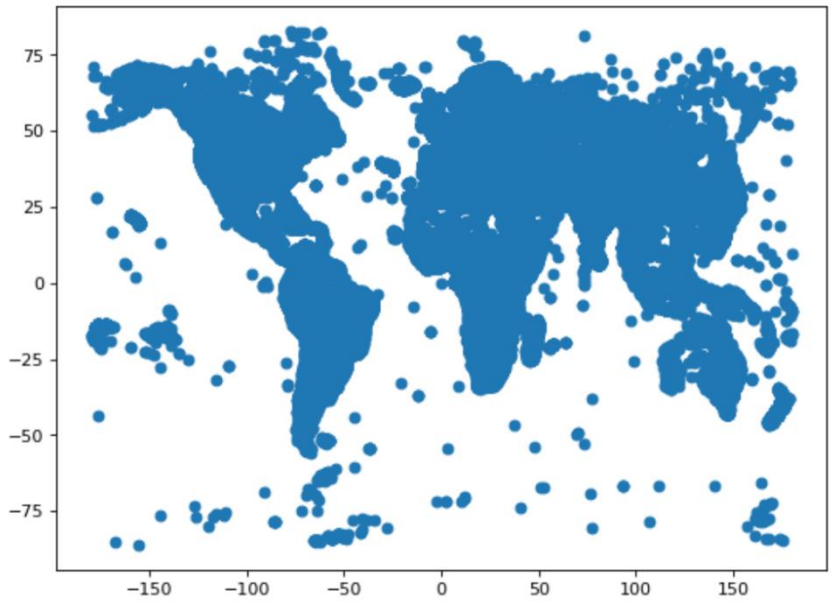
FILE 2:

1. We follow the same cleaning process for the sample_geo.txt file to create a data containing the (latitude, longitude) pairs and LocationID.
2. We create a pyspark data frame and plot the (latitude, longitude) pairs.



FILE 3:

1. We follow similar cleaning process for the lat_longs.txt file to create data containing (latitude, longitude) pairs and Site.
2. We create a pyspark data frame and plot the (latitude, longitude) pairs.



STEP 4:

FILE1:

We will be implementing the k-means clustering algorithm on the cleaned data. Before we move on to the algorithm, we need to create a PySpark data frame. Below would be an example of a PySpark DataFrame.

```
+---------+-----------+-------------------+------------+-----+
| Latitude|  Longitude|               Date|Manufacturer|Model|
+---------+-----------+-------------------+------------+-----+
|33.689476|-117.543304|2014-03-15 10:10:20|    Sorrento| F41L|
+---------+-----------+-------------------+------------+-----+
only showing top 1 row
```

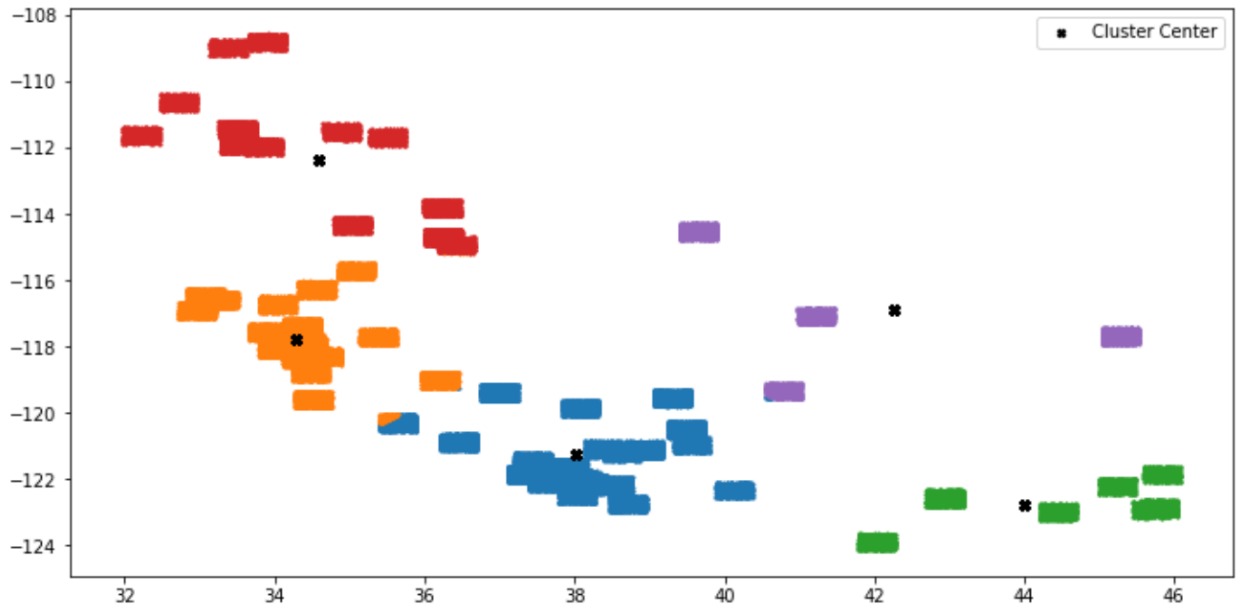
Then, we implement the k-means clustering algorithm using k = 5 which would give the Cluster center coordinates. Again, we create a DataFrame with the new Cluster center coordinates.

```
+---------+-----------+-----------+----------+------------+
| Latitude| Longtitude|Predictions|CCLatitude|CCLongtitude|
+---------+-----------+-----------+----------+------------+
|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
```

Then, we move ahead to find the Euclidean distance – the straight-line distance between two points in space and the Great circle Distance – the distance between two points on the surface of sphere measured along the surface of sphere (in our case Earth).

```
+---------+-----------+-----------+----------+------------+-----------------+------------------+
| Latitude| Longtitude|Predictions|CCLatitude|CCLongtitude|          GCDist|            EucDis|
+---------+-----------+-----------+----------+------------+-----------------+------------------+
|33.689476|-117.543304|          1| 34.297184|  -117.78653|35.598628415939890|0.42846743379777763|
+---------+-----------+-----------+----------+------------+-----------------+------------------+
only showing top 1 row
```

Finally, we plot the Cluster centers for **K = 5**



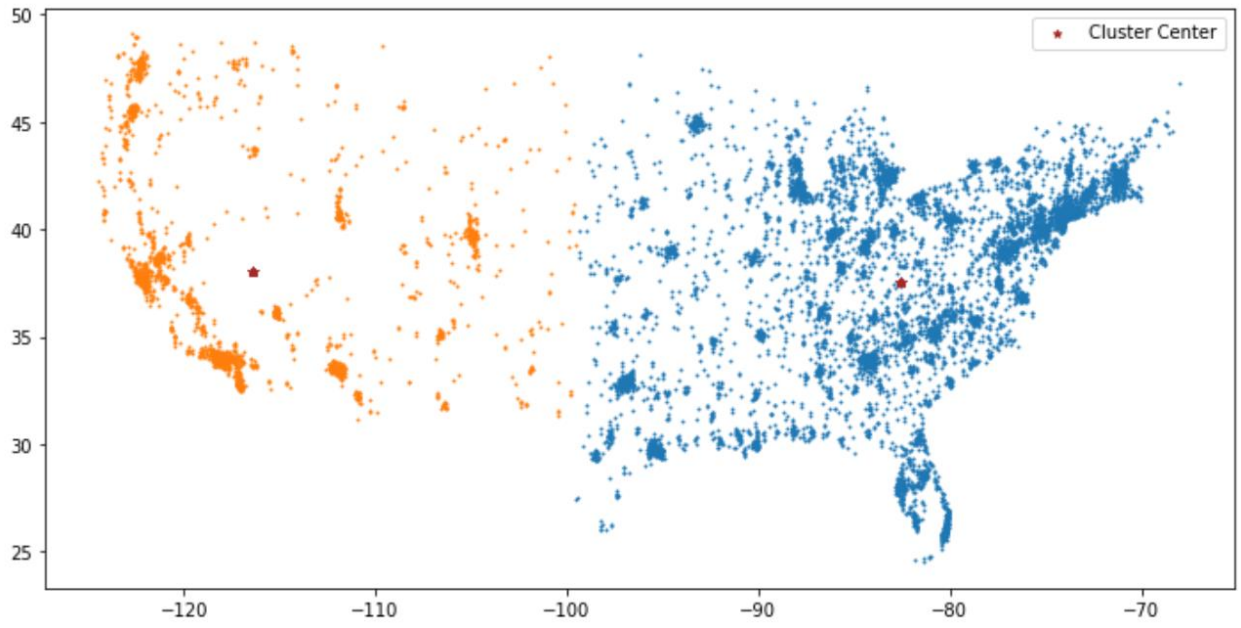
FILE2:

We again clean the data, create PySpark DataFrame, and implement the k-means algorithm using k =2 and k =4 which shows that we will be getting 2 and 4 cluster center coordinates respectively.

Then we calculate the EuclideanDistance and the GreatCircleDistance.

```
+--------+----------+-----------+----------+------------+------------------+------------------+
|Latitude|Longtitude|Predictions|CCLatitude|CCLongtitude|           GCDist|           EucDist|
+--------+----------+-----------+----------+------------+------------------+------------------+
|37.77254| -77.49955|          0|  40.14836|   -76.96599|268175.3106034865|5.929209526933846|
+--------+----------+-----------+----------+------------+------------------+------------------+
only showing top 1 row
```

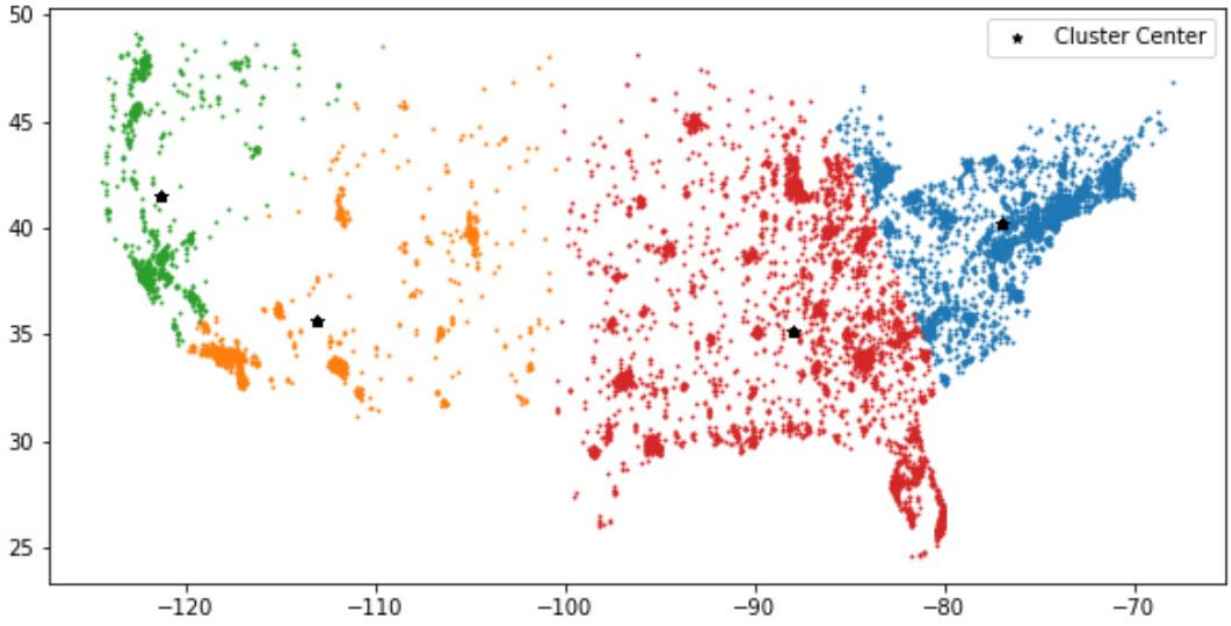
Plot for k = 2



We can see that cluster centers formed in the North East and Western region of the US.

Plot for k = 4

We can see cluster centers in the North East, Central, North West and South West regions of the US.



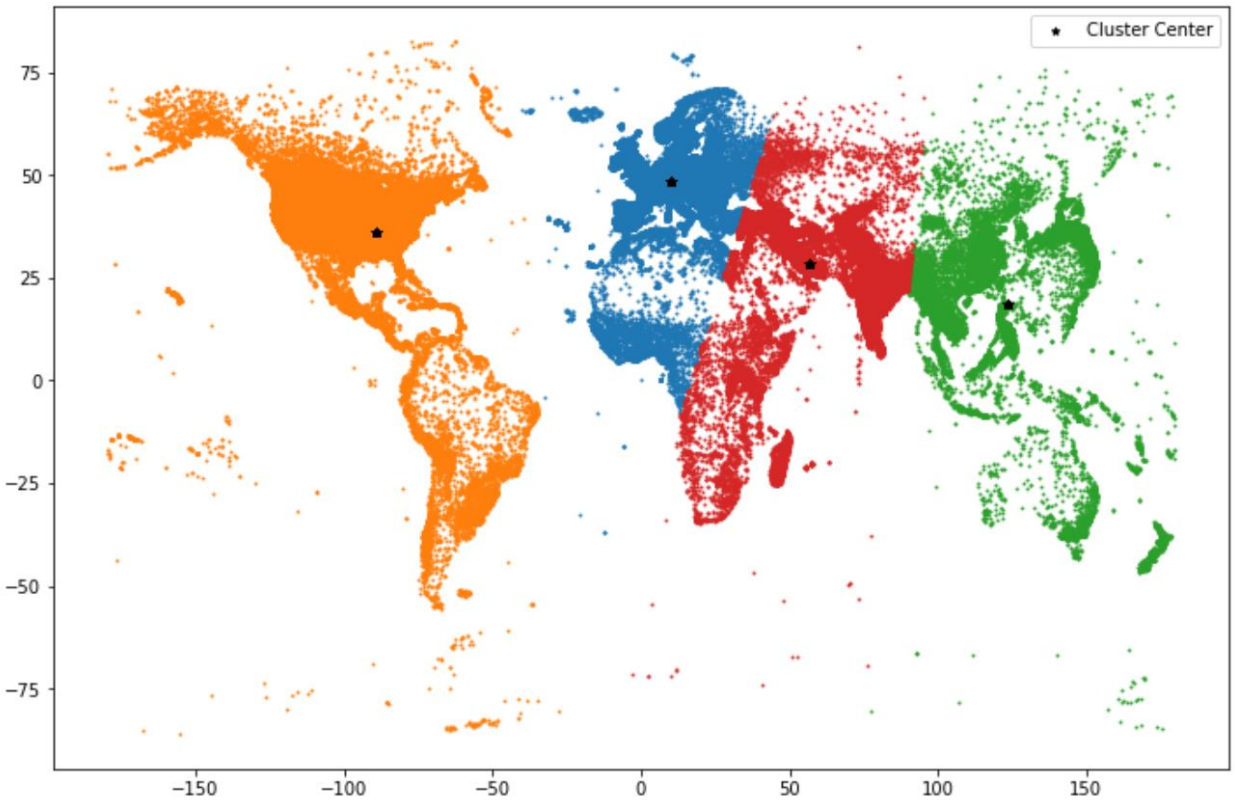
FILE3:

We follow the same cleaning process to standardize the data, create a PySpark DataFrame and implement the k-means clustering algorithm using k = 2 and k = 4.

We then calculate the EuclideanDistance and GreatCircleDistance manually and add them as columns to the existing PySpark DataFrame.

```
+--------+----------+-----------+----------+------------+-----------------+------------------+
|Latitude|Longtitude|Predictions|CCLatitude|CCLongtitude|           GCDist|            EucDis|
+--------+----------+-----------+----------+------------+-----------------+------------------+
|    36.7| 3.2166667|          0| 48.436523|    9.999369|708.4618533443007|183.75101049274076|
+--------+----------+-----------+----------+------------+-----------------+------------------+
only showing top 1 row
```

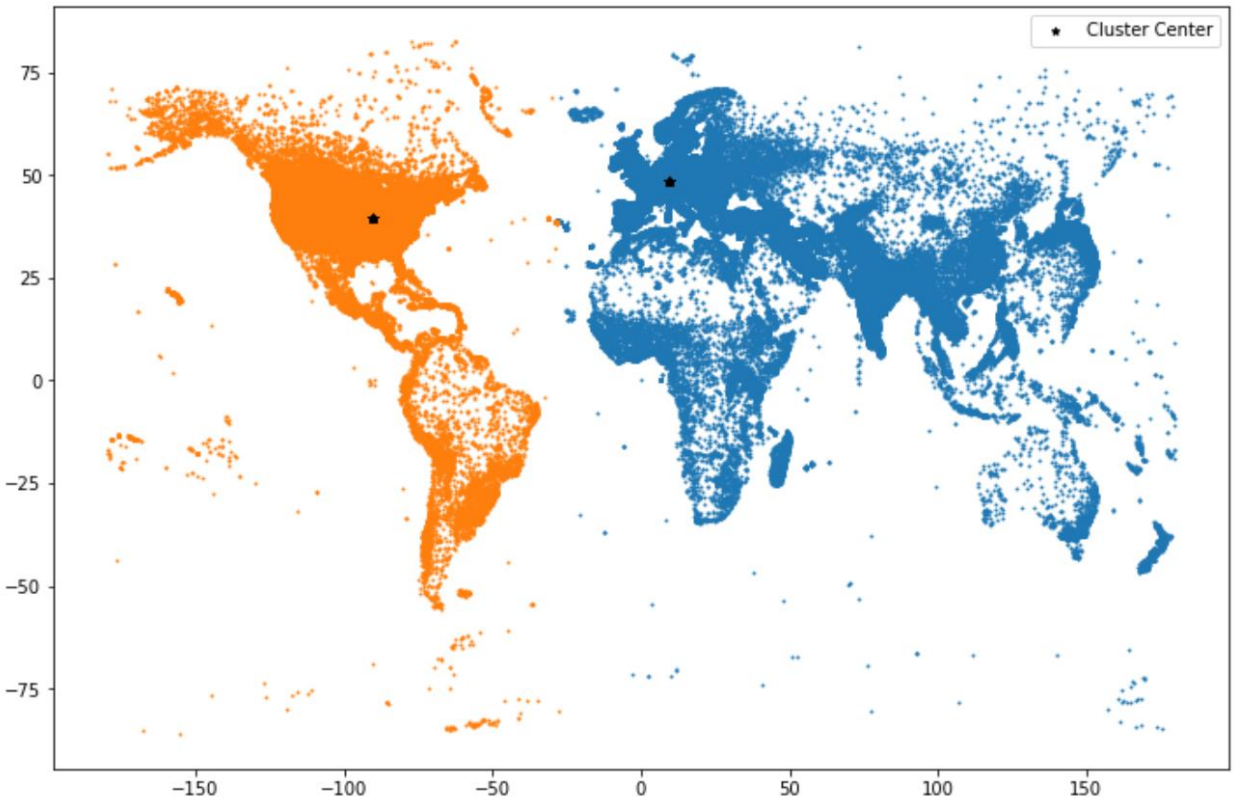
As the data file has the coordinates of the entire world, we could see clusters outside the US. Below is the plot for (latitude, longitude) pairs for lat_longs.txt for k =4.



We calculated the EuclideanDistance and GreatCircleDistance for k = 2

```
+--------+----------+-----------+----------+------------+-----------------+------------------+
|Latitude|Longtitude|Predictions|CCLatitude|CCLongtitude|           GCDist|            EucDis|
+--------+----------+-----------+----------+------------+-----------------+------------------+
|    36.7| 3.2166667|          0|  48.59884|      9.7733|713.1443294349903|184.57178927056543|
+--------+----------+-----------+----------+------------+-----------------+------------------+
only showing top 1 row
```

Below is the plot for (latitude, longitude) pairs for k = 2.



Future Steps:

We could try implementing a similar approach by increasing the number of clusters which would show us more user activity in areas where the network would generally be scarcer. From this, we can derivate more decisions based on the customer demand and user activity.

Conclusion

We implemented the k-means clustering for the data provided by a wireless carrier. Based on the analysis and visualizations, we can see that the network is focusing it's service on highly populated areas and less in areas with difficult-to-reach geographic conditions.

We could improve by installing more transponders to expand the service for the people in less inhabitant regions.