**Final Project Report: K-Means for Geo-Location Clustering in Spark**

**1. Project Motivation**

K-mean as one of unsupervised learning algorithms is very powerful in data visualization and data interpretation. It accomplishes the idea of auto-clustering or auto-categorizing. We can use k-mean in marketing data, document data and geographic data. It occupies different meanings in classification, behavior clustering and location evaluation. In this project, we will apply k-mean algorithm in SPARK environment where we will extract different sizes of big datasets and fit them into k-mean algorithm. We will visualize results from clustering and show our experimental results with our comprehensions.

**2. Clustering Approach**

Choices: k-mean, k-median, EM algorithm.

Model: k-mean

Training Parameters: k centroids, distance measurement.

Visualization: matplot

**2.1 K-means Stages**

We break down our approach into three stages. We name first stage as data initialization, second stage as training model and last stage as visualization.

Stage1: we choose proper number of k clusters as using random choice where we restrict the range of randomized picking to achieve proper values of centroids. A local dataset will be chosen for experimental result for initial practice.

Stage2: k-mean will be fit by geographic data and will make k’ iterations to form k clusters

Stage3: we will visualize different clusters with different colors by using matplot package in python

**2.2 K-Means Implementation and Program Design**

Work Flow:

closest\_point(): given a (latitude/longitude) point and an array of current center points returns the index in the array of the center closest to the given point

add\_points(): given two points, return a point which is the sum of the two points.

euclidean\_distance(): calculate the eculidean distance between two points.

great\_circle\_distance(): the shortest distance between two points on the surface of a sphere, measured along the surface of the sphere

WCSS(): within-clusters sum-of-squares measure

kmeans\_cluster(): aggregating functions above and make k clusters

visualize(): make color for k groups of data points and make label ‘x’ on k centroids

read\_data(): extract data from given datasets

**3. Small Dataset on Pseudo Cluster**

3.1: Dataset Information

3.2: Local Execution Results and Analysis

**4. Big Data Application and Cloud Computing**

4.1 Dataset Source and Description

4.2 Cloud Computing on Amazon EMR Cluster

4.2.1 Modification and Implementation for Cloud Execution

4.2.2 Execution on Amazon EMR Cluster

4.3 Cloud Execution Results and Analysis

5 Conclusion

6 Reflection after the Final Project

6.1 Lessen learned

6.2 Future Work

The result on 4 clusters indicates that two different distance measurement results in different distances between clusters. The clusters in great circle measurement are closer to each other than in Euclidean distance measurement, however, great circle measurement is more complex than Euclidean distance measurement from perspective of algorithm analysis. Our big dataset derives from some area of European. The computation advantages could not be evaluated from the datasets we applied.