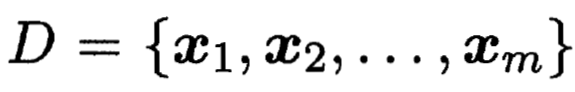
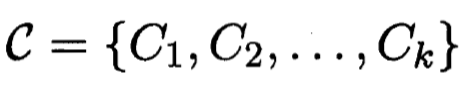
Report

**1.Theory:**

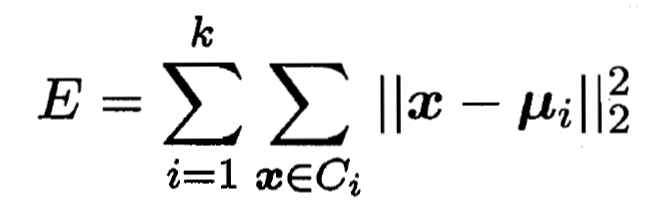
Given sample set

,

k-means clustering divides the datasets into a set of k clusters

，

where the error is minimized:

.

**2.Goal:**

This project intends to perform a standard algorithm for K-mean clustering on different datasets.

I will do the following things in both parallel and normal way:

1. Perform the algorithm on random 2d vectors and visualize the clustering.

2. Perform the algorithm on digits MNIST and visualize the mean points to give an image of "average looking" in different clusters.

3. Perform the algorithm on fashion MNIST and visualize the mean points to give an image of "average looking" in different clusters.

And, I posted the times of performing the algorithm on digits MNIST in two different ways.

All of the visualizations are in "Visualization.ipynb."

**3.Implementation:**

The c code is used to run standard algorithms for K-means clustering.

--------------------------------------

Code:

The core part is the ParallelKMC function. The normal KMC consists of two part update and assignment as explained in part 1. The algorithm then iteratively runs the two functions. The main for-loop cannot be reduced further. Since it is necessary to know the state in the last iteration to execute the next one, we cannot split the iterations into blocks with a starting point not known. The update and assignment parts can run in a parallel way.

In the function para\_assignment, I want to calculate the nearest points around mean points, I simply run the algorithm on different blocks of data points.

In the function para\_update, I need to calculate mean points based on current clustering. I just simply run the algorithm on different blocks of clusters.

--------------------------------------

Compile Command:

gcc -fopenmp Kmeans.c -o Kmeans

---------------------------------------

Run Command:

./Kmeans k iterations path

i.e.

./Kmeans 10 250 ./digits\_train.csv

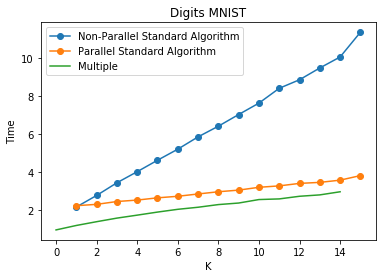
---------------------------------------

Other Files:

In the code, I have an export function to rewrite data.csv, meanPoints.csv, clustering.csv to label the data. In the Jupyter notebook Visualization.ipynb, I used python to visualize the graph for K-means clustering for random dimension 2 vectors and the meanPoints for MNIST data.

**4.Results:**

Time for Clustering Digits MNIST in Parallel and Normal Way v.s. Number of Clusters (multiple =Non-parrallel Time/ Parallel Time):



Variance of the Clustering v.s. Number of Clusters:

