Gentry Atkinson
CS7331: Spring 2019
Semester Project Proposal

1. Introduction:

There are 100s of accepted techniques for clustering sets of unlabeled data into broad groups, of which dozens are in common usage. This can make it difficult for an analyst to take a clustering produced by some algorithm and asses it as being valid or not. Commonly, the clusters are compared to the expectations that an expert in the data's field would have in order to determine if the clusters are a good fit. But that relies on the existence of some fore-knowledge and so not every researcher can rely on that sort of approach. Furthermore, such external validations of data clusters are fairly ad-hoc and subjective, so two experts in a field might disagree about how well fitted the same set of clusters are on the same data. Finally, external validation does not say anything about the strength of the algorithm or contribute to the standing of the science.

To compensate for these weaknesses many indices have been developed which numerically indicate the strength of the fitting between some clusters and some data based only on the similarities between points in the data. This allows for two algorithms, or the same algorithm with different parameters, to be evaluated on some data so that the best clustering can be chosen. These indices are calculated from the data and are collectively referred to as Internal Validity Measures. Two of the most common indices for measuring cluster validity are the Dunn Index proposed in 1974, and the Silhouette Index proposed in 1987.

Both the Dunn and Silhouette indices are reliable and commonly accepted but both rely on calculations which compare each point in a data set to each other and so can be as expensive as the task of clustering the data originally was. Using parallelism to improve the speed with which these indices can be calculated would make them much more useful. This project will create sequential and parallel implementations of both the Dunn and Silhouette indices using OpenMP and then compare their speed on some testing data.

2. Tools:

2.1 The Dunn Index was first proposed by JC Dunn in the Journal of Cybernetics in 1974. It indicates the ratio between the minimum distance between any two clusters and the the maximum of the mean distance between points in any cluster. Because it divides by the max average distance,one poorly fit cluster can badly skew the index, so it is treated as the 'worst case' fittedness of any cluster in a set.

2.2 The Silhouette Index was first proposed by Peter J. Rouseeuw in the Journal of Computational and Applied Mathematics in 1987. It generates a graphical representation of the fittedness of some clusters by comparing the closeness of points within a cluster to their closeness to points in other clusters (cohesion vs. separation). In this way it is similar to the Dunn Index but rather than returning a single value, it produces a graph representing the fittedness of the clusters for easy interpretation by the analyst.
2.3 K-Means is a clustering algorithm which has been in common usage since 1967. It functions by guessing and then iteratively adjusting k cluster centers within some data. Although K-Means is not being

tested or evaluated in this project, it will be used to generate the clusters that will then be evaluated by means of Dunn and Silhouettes.

2.4 OpenMP is an API which is used to add multithreading to C++ programs using a series of compiler directives. It was first released for Fortran in 1997 and was expanded to C++ in 2002. The use of #pragma directives within the code means that the parallelism can easily be turned on and off at compile time, essentially allowing for sequential and parallel executables to be produced from the same code.

3. Data:

Synthetic data will be generated for this project in order to ensure that easily detectable clusters exist within the code. Two data sets will be generated each with 100,000 points. The first data set will 2-dimensional and points will be uniformly distributed around three centers. The second data set will be 10-dimensional and points will be uniformly distributed around two centers.

These data sets will each be clustered using K-Means with k equal to 2 and 3 using a Euclidean distance measure. This will produce four clustered datasets of which two should be well fitted and two should be poorly fitted. The experiments described in Section 4 will be conducted on these four sets.

4. Procedure:

An implementation of Dunn and Silhouette will be written in C++. Both algorithms rely on large summations so the code will have expensive for loops which can be parallelized using OpenMP #pragmas. Each algorithm will have three executables produced for it, one sequential and two with varied degrees of parallelism. The degree of parallelism might be determined by the number of threads specified or by the level of optimization specified at compile time. At this time, I am not certain which will be the most appropriate for this project.

Each of the six total implementations will be run on each of the four datasets giving us 24 total runs. The runs will be timed using the perf tool on a Linux system. Of these runs 6 will be well fit on dataset 1, 6 will be poorly fit on set 1, 6 will be well fit on set 2, and six will be poorly fit on set 2. I will tabulate and chart these results to determine the amount of speed-up which is realized with multithreaded implementations of Dunn and Silhouette.

5. Expected Results :

I am confident that there will be a measurable speedup in both Dunn and Silhouette and that the speed-up will increase at the higher degree of parallelism. I do not expect for there to be an appreciable difference in the speed-up that is realized in Dunn vs. Silhouette since both algorithms rely on similar calculations but it will be interesting to see if this holds true.