CPSC-8650 Data Mining Spring 2019

Final Project

**Title:**

Lightweight Coreset Construction for K-means and K-medoids Clustering and Performance Analysis.

**Team Number: 5**

**Team Members:**

Cai, Ying [cai7@clemson.edu](mailto:cai7@clemson.edu)

Zhang, Wenxing [wenxinz@clemson.edu](mailto:wenxinz@clemson.edu)

1. **Introduction**

Unsupervised cluster analysis groups a set of objects such that the objects in the same group (called a cluster) are more similar to each other than to those in other clusters. It is a common technique for statistical data analysis, and is widely used in many fields, such as machine learning, pattern recognition, image analysis, bioinformatics, information retrieval, data compression, etc.

There are several common clustering approaches, such as partitioning approach, hierarchical approach, density-based approach, grid-based approach, etc. K-means and K-medoids both are partitioning approaches. Briefly, they partition a dataset of N objects into k clusters, to minimize some chosen evaluation metrics, such as the sum of Euclidean distances. K-means represents each cluster by using the mean center of the cluster, and assigns each point to the closest cluster center. K-medoids represents each cluster by the centroid object of this cluster, and also assigns each point to the closest cluster. They are very popular methods for clustering analysis.

Both of them are heuristic methods. K-means method, which runs in O(n) time with a large hidden constant, is more efficient than K-medoids method, which runs in O(n2) time. But K-means is more sensitive to outliers and noisy data.

The recent, unprecedented increase in the size of data sets leads to the big computational challenge for them. Coresets can be employed to speed up the inference.

“Coresets are a proven data summarization approach that can be used to scale clustering problems to massive data sets. Coresets are small, weighted subsets of the original data set such that models trained on the coreset are provably competitive with models trained on the full data set.” [1]

Coresets have been successfully constructed by some researchers and applied for a lot of clustering problems. Here, Olivier Bachem, Mario Lucic and Andreas Krause proposed a novel approach in their paper “Scalable k-Means Clustering via Lightweight Coresets” which is published by KDD 2018.

They prove that the sufficient coreset size is , which is smaller than the which is proved by previous study. A simple and embarrassingly parallel algorithm is proposed too, to construct such lightweight coresets. kmeans++ algorithm is used to solve the clustering problem on subsamples and full dataset.

We reproduce their result using 2 different datasets. We also use the k-medoids method to cluster subsamples and full datasets too.

1. **Coresets construction**

We use 3 construction methods.

1). Uniform: randomly choose m points from datasets.

2). CS

3). Light weight coresets (LWCS).

The coresets are constructed by sampling m weighted points from datasets. Each point x has weight and is sampled with probability where is the mean of the whole dataset X.

has 2 terms. The intuition behind the second term is that “the points that are far from the mean of the data have a potentially large impact on the quantization error of a clustering.” [1]. The distance used in is the Euclidean distance. Following is the construction algorithm. [1]



1. **Performance analysis.**

The performances are valued by the running time and the relative error vs. full datasets.

Running time includes the time used to construct the coresets and the time used to solve clustering problem on subsamples or full dataset using kmeans++ or kmedoids algorithms.

After we cluster the sample set, we have k clusters. Assign each object in full dataset to these k clusters and compute the quantization error . This sample quantization error is compared with the quantization error given by solving clustering problem on full set. This relative error is employed to evaluate the clustering quality.

1. **Experimental setup**

We choose samples size m {1000, 2000, 5000, 10000, 20000} and clusters k {100, 500}.

Programs are coded using Python. The multi-processing package is used to construct coresets. The sklearn/cluster package is employed to solve the kmeans++ clustering problem. Since we don’t find a kmedoids package which handles weighted points, we write the algorithm ourselves.

Require: subsample X, number of clusters k, max-iteration maxi, tolerance t

1. Randomly choose k medoids from X
2. while iterations < maxi or reduced cost > t：

for each object in X:

assign cluster

for each cluster:

for each object O in this cluster:

cost =

choose the object whose cost is minimum as the new medoid of this cluster

1. Return medoids of k clusters.
2. **Environment**

We use the Clemson University Palmetto Cluster. The lightweight coresets are constructed parallelly, we apply for several to tens of cores in order to speed up the construction.

1. **Datasets**

Two datasets are analyzed in this study.

1. KDD — 145’751 samples with 74 features measuring the match between a protein and a native sequence [3].
2. SONG — 90 features from 515’345 songs of the Million Song datasets used for predicting the year of songs [5].

Kmeans clustering is performed on both datasets. Kmedoids clustering is only used for dataset KDD.

1. **Expect results**
2. **References**
3. Olivier Bachem, Mario Lucic, Andreas Krause. Scalable k-Means Clustering via Lightweight Coresets. KDD 2018, August 19-23, 2018, London, United Kingdom.
4. Mario Lucic, Olivier Bachem, and Andreas Krause. 2016. Strong Coresets for Hard and Soft Bregman Clustering with Applications to Exponential Family Mixtures. In Artificial Intelligence and Statistics.
5. KDD Cup. 2004. Protein Homology Dataset. Available at <http://osmot.cs.cornell.edu/kddcup/datasets.html>.
6. Matthew Faulkner, Michael Olson, Rishi Chandy, Jonathan Krause, K. Mani Chandy, and Andreas Krause. 2011. The Next Big One: Detecting Earthquakes and other Rare Events from Community-based Sensors. In *Information Processing in Sensor Networks*.
7. Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. 2011. The Million Song Dataset. In *International Conference on Music Information Retrieval*.
8. Andrew V Uzilov, Joshua M Keegan, and David H Mathews. 2006. Detection of non-coding RNAs on the basis of predicted secondary structure formation free energy change. *BMC Bioinformatics* (2006).