Big Data, Big Problems–Parallelizing Kmeans

Reuben K. McCreanor
Department of Statistics
Duke University
Durham, NC 27708
`reuben.mccreanor@duke.edu`

Wei (Emily) Shao
Department of Statistics
Duke University
Durham, NC 27708
`wei.shao@duke.edu`

Abstract

K-means is a data-partitioning algorithm that separates n observations into k partitions. Due to its simplicity of implementation, its easy interpretability, and its ability to categorize data based on desired features, it remains one of the most popular algorithms in fields of statistics and machine learning. However, the key issue with k-means comes from its efficiency and scalability. As the size of datasets continues to increase, implementing k-means on data of any magnitude becomes computationally infeasible. A recently proposed variation of k-means, k-means++, provides a robust method of selecting the initial centers, essentially giving an optimal solution. However, due to the number of passes over the data required in a naive implementation, even clusters a million data points into 100 partitions would be exceedingly slow. In order to combat this, a parallelized version of k-means++ is proposed, k-means $||$. This version uses a sampling factor ℓ to dramatically reduce the number of passes while still arriving at an equivalent solution of partitions. This paper will implement k-means++ and k-means$||$ in both sequential and parallel setting and compare the results both in terms of efficiency and equivalency of partition arrangements.

1 Introduction

In general, clustering is a means of grouping observations into a set such that the other observations in this set are more similar to each other than to those in other sets. Starting with a data set, we use the use the k-means algorithm to define which observations are most similar to each other, and thus categorize these observations into k groups.

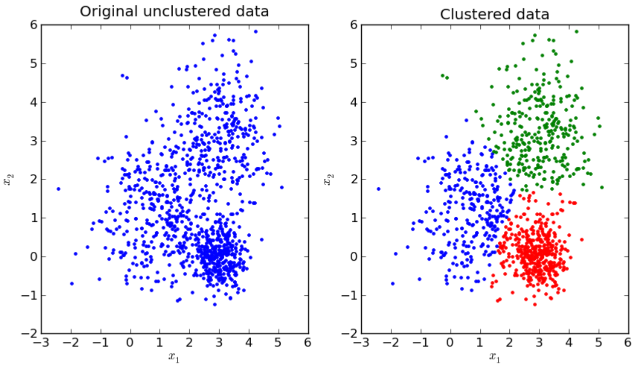

Figure 1: The left hand side shows unclustered data, which the right hand side shows the same data categorized into three clusters

In the example above, we can data can be clustered into three partitions based on the similarity between observations. In industry, a common application of clusters is in market segmentation. Taking the example of Lenovo, an online hardware retailer, they use k-means clusters as a way of understand their total addressable market. Given that there total market is comprised of people from many different incomes, locations, and type of consumers, clustering allows them to understand which groups are most similar based on these attributes and thus design tarketing marketing campaigns for each discinct group.

2 Overview of the Algorithms

2.1 K-means++

2.2 K-means||

3 Implementation

3.1 Sudo Code

3.2 Data Simulation

3.3 Testing

4 Optimization

4.1 Numba

4.2 Cython

4.3 MapReduce and Hadoop

5 Results

5.1 Comparison of Numerical Results

5.2 Comparison Clustering Arrangements

5.3 Comparison of Efficiency

6 Conclusions

References

Bahmani, Bahman, Benjamin Moseley, Andrea Vattani, Ravi Kumar, and Sergei Vassilvitskii. *Scalable K-means.* Proc. VLDB Endow. Proceedings of the VLDB Endowment 5.7 (2012): 622-33. Web.

7 Appendix