Approximation And Evaluation of K-means Clustering Method

Matzoros Christos , Angelis Marios

cmatzoros@inf.uth.gr, mangelis@inf.uth.gr

Supervisor: Christos D. Antonopoulos

July 2019

1 Introduction

The problem of data clustering has been widely studied in data mining and machine learning literature as well as for statistical analysis because of its numerous applications. K-Means method is considered one of the simplest and most classical methods for data clustering and is also possibly one of the most widely used methods in practical implementations because of its simplicity. It is an iterative algorithm that converges in a specific result based on the stopping criterion that has been set. This process may become compute-intensive especially in bigger data-sets to conclude to an accurate final result.

Approximate computing is an emerging model that allows trading-off performance and energy efficiency with accuracy. It based in the idea that specific phases of the computation may provoke higher performance and better energy efficiency without a corresponding contribution to the quality of the final result. Approximate computing surround a broad spectrum of techniques that relax accuracy to improve efficiency. There are different levels of design abstraction where approximate computing methods can be implemented. For the K-means algorithm we will examine some techniques in higher abstraction level in software.

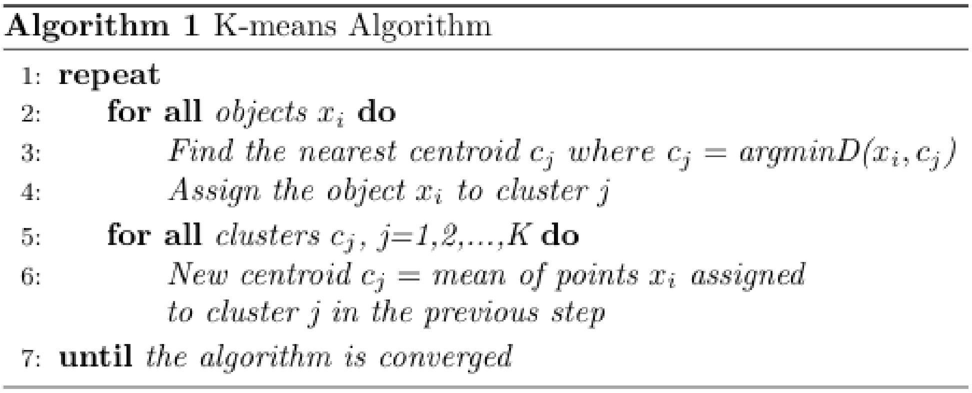
2 K-means Method

Cluster analysis tries to group unlabeled objects based on the information that can be found in them. The objective is to create groups for the objects within a group to be similar to one another and different from the objects in other groups. K-means algorithm appertains to the category of partitioning-centroid clustering where the objects are divided into non-overlapping subsets of objects where every object belongs to exactly one subset.

K-means is the most important flat clustering algorithm. The k-means algorithm partitions the given data into k clusters. The value of K is known upfront. Each cluster has a cluster center, called centroid. Its objective is to minimize the average squared Euclidean distance of objects from their cluster centers or centroids c_i. The cluster center is equal to the mean of the objects in this cluster.

2.1 Algorithm description

The first step is to partition the data space into K clusters and the objects are randomly assigned to the clusters. Secondly, there is an iterative stage where for every object of the data set is calculated the distance from the centers of every cluster. If the object is closest to its own cluster there is no change else we have to select the closest cluster to be assigned to. We repeat the previous process until we reach a convergence point based on the convergence criterion that has been set.



Algorithm 1 K-means Algorithm

```
repeat
    for all objects $x_i$ do
        Find the nearest centroid $c_j$ where $c_j = argminD(x_i, c_j)$
        Assign the object $x_i$ to cluster $j$
    for all clusters $c_j$, $j$=1,2,...,$K$ do
        New centroid $c_j$ = mean of points $x_i$ assigned
        to cluster $j$ in the previous step
until the algorithm is converged
```

Figure 1: K-means Algorithm

2.2 Convergence criterion

K-means method is an iterative method that needs a convergence criterion. The most simple method is to stop the algorithm when there are no re-assignments of objects to different clusters. However, this method may be too expensive as it needs more iterations in order to be achieved. Therefore, it can be stopped earlier when a small number of re-assignments are achieved in one iteration. Another criterion is to stop when there is no or minimum change of the centroids. Furthermore, we can use as a stopping criterion the minimum decrease in the sum of squared error (SSE).

$$SSE = \sum_{j=1}^{K} \sum_{i=1}^{N_j} D(x_i, c_j)^2 \quad where\ x_i \in c_j$$

3 Implementation of approximation techniques

We will examine some approximate computing techniques that can be used to reduce the computation time of the K-means clustering method. We measured the computation time for the K-means clustering problem using as a convergence criterion the state in which a small number of re-assignments are achieved in one iteration. Exactly, when an object changes cluster, a variable(delta) is increasing by one. If the value of the fraction delta/objects is less than the threshold value, the algorithm converges. As long as the threshold falls to 0, the computation becomes more accurate, but the computation time becomes larger.

3.1 Loop Perforation

The loop perforation technique provides a general approach to trade accuracy for performance by transforming the loops of the code to execute a subset of their iterations. The goal is to reduce the amount of computational work and therefore the amount of time and/or other resources such as the power that the computation requires to produce its result. This approximation method may cause the computation to produce a different result,but approximate computations can often tolerate such changes as long as they do not unacceptably reduce the accuracy under some specific threshold. The value of the threshold that can be accepted in order to tolerate the output error, is determined by the behaviour of the specific application and the input data.

3.2 Precision Scaling

Precision scaling method is used to minimize the computation overhead. Reducing the precision (bit-width) of the intermediate variables, the storage and the computing requirements are also reduced. Observing the accuracy of the result, a bit-width threshold can be found and if the precision scaling method is applied, a negligible accuracy loss will be noticed at the output.

About the k-means clustering problem, we converted the type of the variables relating to the euclidean distance and to the cluster center computations, from float to double. However, we noticed that increasing the precision, the computation overhead is lower[Figure 2]. This happens because of the convergence iteration count. As the accuracy of the calculation grows, the coordinates of the cluster centers are more accurate, so the algorithm converges faster. [Figure 3] shows that if doubles are used, the iterations for full convergence are less than the use of floats.

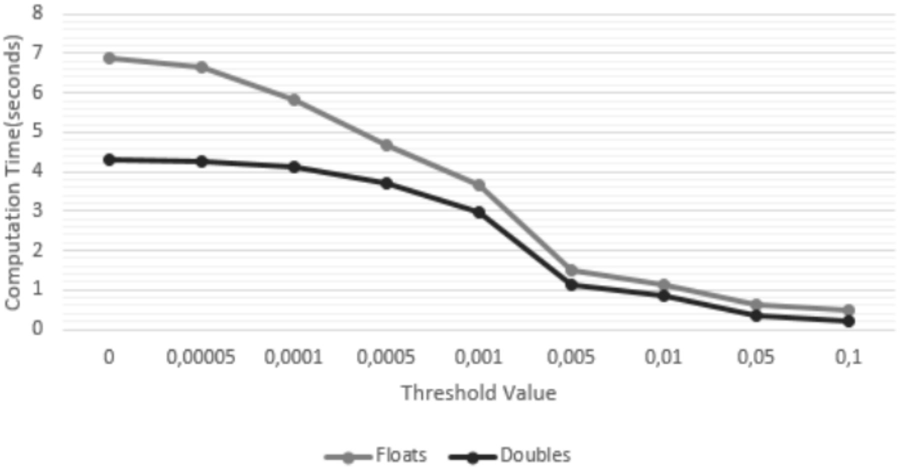


Figure 2: Computation time(in seconds) versus threshold value

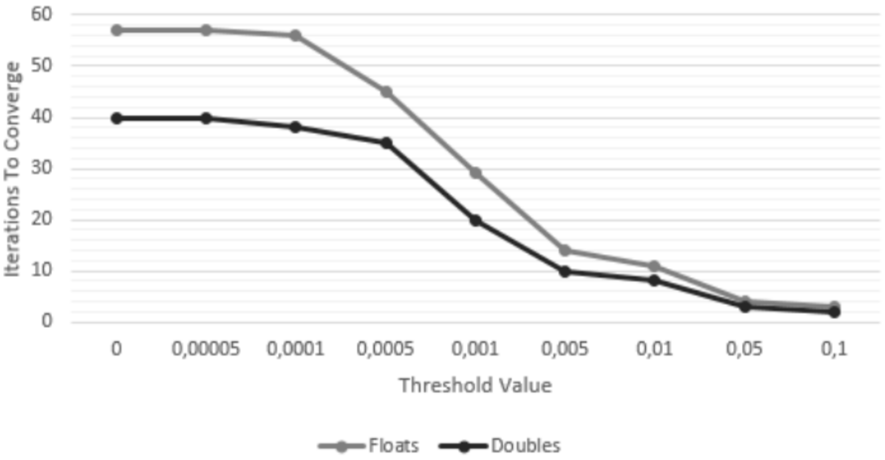


Figure 3: Iterations to converge versus threshold value

3.3 Object Sampling

An acceleration of the computational process can be the choice of a smaller subset of the data set. In that case, the K-means method can be used with a smaller sample of data. We have implemented this technique by randomly selecting a subset of the initial object data set. To make our measurements,we used a data set which has 17695 objects. We measured the accuracy of the approximated algorithm looking at the divergence of the clusters centers in relation to the values that existed before the implementation of the object sampling. Due to the fact that each cluster centroid must be mapped to only one another, we mapped each centroid of the approximated version(object sampling is applied), with a centroid of the non-approximated version(no sampling is applied).The euclidean distance between these mapped centroids is the minimum. For example, lets assume that c1, c2 are the centroids of the non-approximated version and c1', c2' are the centroids of the approximated version. If the minimum euclidean distance for center c1' is the distance from centroid c2, then c1' is mapped with c2. If the minimum euclidean distance for centroid c2' is the distance from centroid c2 again, it must be checked if c1' or c2' is closer to the centroid c2. So, if (c1'-c2) < (c2'-c2), then c1' is mapped with c2 and c2' is mapped with c1. Last but not least, after mapping all the centroids, our algorithm calculates the accuracy of the approximated algorithm. For each pair of the mapped centroids, if the divergence is larger than a threshold(for example 0.0001),then an error increases by one. Finally, accuracy is calculated from the mathematical formula below : accuracy = [100 − (100*error)/numClusters]% . Last but not least,our algorithm calculates the average distance.

The average distance is the average of the deviations of the mapped centroids.[Figure 4] shows that as we decrease the objects data set,the total accuracy becomes even smaller. However, if we decrease the data set to half, the accuracy remains high(99,4 percent) and the computation time is reduced to half. Also, [Figure 5] shows that as we decrease the objects data set, the total average distance becomes even greater. However,if we decrease the data set to half, the average distance remains small(0.000009) and the computation time is reduced to half.

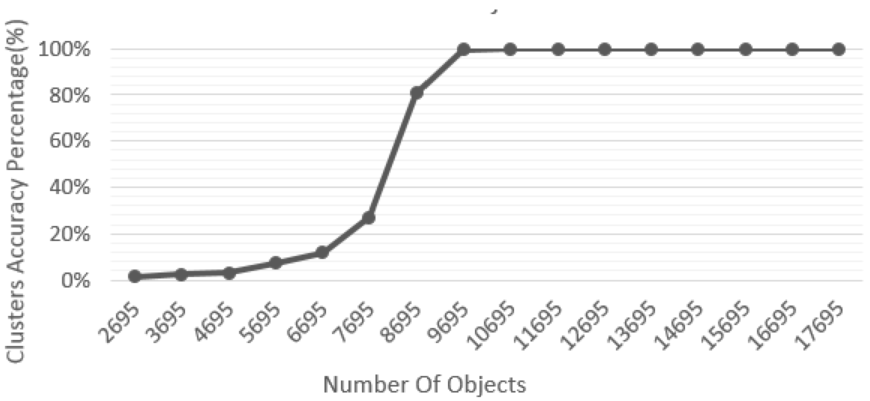


Figure 4: Percentage of clusters that do not change coordinates in comparison with the accurate version versus the number of objects

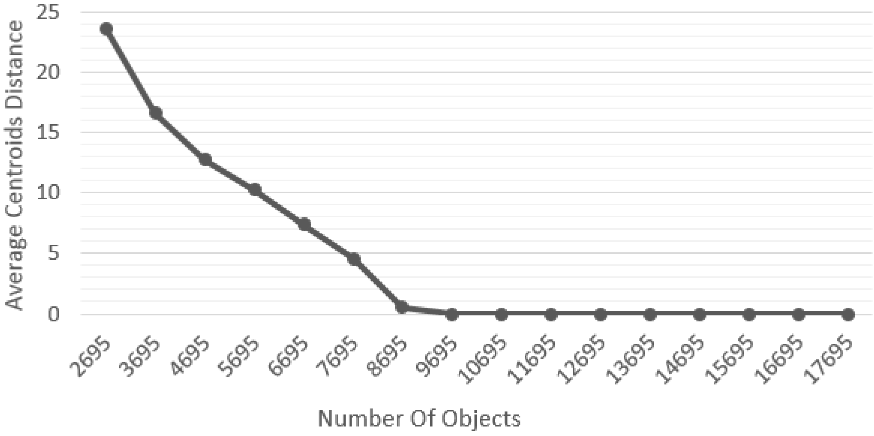


Figure 5: Average centroids distance versus the number of objects

3.4 Dimension Sampling

After the implementation of the object sampling technique, we thought that except from the objects, the dimensions can be reduced.[Figure 6] shows that as we decrease the dimensions, the average distance is between values 4.9 and 6.4, except in one case(dimensions=3). This happens because the average distance depends on the divergence between the mapped centroids. Also, the divergence depends on the values of the objects in each dimension. Thus, for a sample with r dimensions, if the initial dimensions are n, the number of the combinations is computed from the mathematical formula below :

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}$$

In the [Figure 6], we have noted the minimum average centroids distance values of all the combinations for each dimension number.

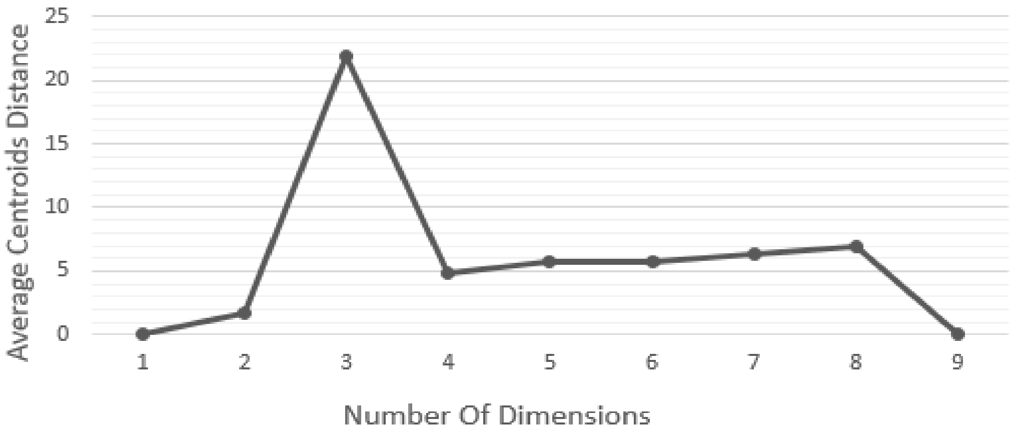
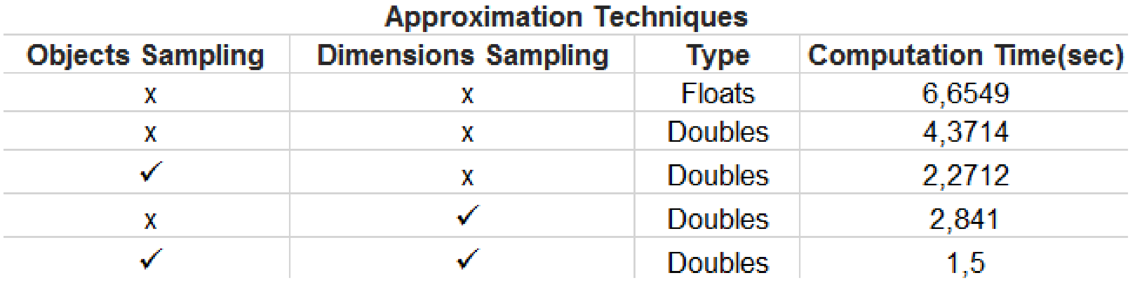


Figure 6: Average centroids distance versus the number of dimensions

4 Conclusion

So far, we demonstrated some higher level approximation techniques that improve the performance of the K-means method. Using loop perforation, we achieve to terminate earlier the computation, appending a negligible error to the final result. We control the scale of this error by fluctuate the value of a threshold variable. In addition, the use of double precision variables instead of floats, leads to a better computation time as it converges faster. Also, with the correct magnitude of data sampling, we can choose a smaller amount of data from the initial object data set and gain better performance. The data set object values that will be selected through the sampling method may affect the behavior of the computation differently. We summed up all the above approximation techniques and we have concluded that in order to see a negligible output error and reduced computation time, the best combination is the use of the object sampling technique and the use of double variables. [Figure 7] presents the k-means algorithm's computation time for each combination of approximation techniques. For the diagram below, the sample used for the object sampling technique, is the 50 % of the initial data set and the dimensions used for the dimension sampling technique are 4 in total.



Approximation Techniques

Objects Sampling	Dimensions Sampling	Type	Computation Time(sec)
x	x	Floats	6,6549
x	x	Doubles	4,3714
✓	x	Doubles	2,2712
x	✓	Doubles	2,841
✓	✓	Doubles	1,5

Figure 7: The approximation techniques and their corresponding computation times

References

[1] The Elements of Statistical Learning Data Mining, Inference, and Prediction
https://web.stanford.edu/ hastie/Papers/ESLII.pdf

[2] Convergence Properties of the K-Means
http://www.iro.umontreal.ca/ lisa/pointeurs/kmeans-nips7.pdf

[3] The Stanford NLP Group, Introduction to Information Retrieval
https://nlp.stanford.edu/IR-book/html/htmledition/k-means-1.html

[4] MIT, Computational Aspects of Biological Learning, Fall 2014
http://www.mit.edu/ 9.54/fall14/slides/Class13.pdf

[5] The Complexity of the k-means Method, Tim Roughgarden and Joshua R. Wang

[6] Managing Performance vs. Accuracy Trade-offs With Loop Perforation:Stelios Sidiroglou, Sasa Misailovic, Henry Hoffmann, Martin Rinard
https://people.csail.mit.edu/stelios/papers/fse11.pdf

[7] A Survey of Techniques for Approximate Computing,Sparsh Mittal, Oak Ridge National Laboratory