Research and Applications behind K-Means Clustering and Initial Centroid Selection

Eitan Romanoff
Mike O'Dell
Kyle Theriault
Dashawn Mitchell

Table of Contents

Introduction

In the world of computation, specifically in areas that deal with large quantities of data, there always exists a problem where relations between instances are either obscure, or difficult to present in meaningful ways. This has become an increasingly important problem as data becomes more abundant in the computing fields. Many techniques exist in attempts to make relational rules, and build models that fit large datasets, but oftentimes these approaches are unclear when it comes to presentation – especially when one is dealing with thousands of instances of data or more.

For some situations, there exists clustering, one method of creating these relations, that has the benefit of being presentable in an intuitive manner. Clustering, or the act of grouping like-instances together to form like-groups, can be done through various pooling techniques. The premise of clustering is simple, but relies on a major assumption with regards to the data, in that the Euclidean distance between instances is an appropriate measure of "likeness". This paper goes into the intricacies of the K-Means clustering algorithm, various research that has been performed on its implementation, as well as its current usages and trends in the real world of computation and industry.

The K-Means Algorithm

K-Means clustering, one of the most powerful and popular clustering techniques, was developed in 1967 by James MacQueen, professor at University

of California at Los Angeles. The algorithm tends follows the same general algorithm – one that is simple, as well as parallelizable and open for many tweaks and refinements. The general idea behind the algorithm is that points, so long as the Euclidean distance can represent their “likeness”, are assigned to a “Cluster”, which acts as a collection of like-points. These clusters are mutually exclusive in their inclusion of instances, and through attempt to create classifications of instances through bucketing on Euclidean distance. The general algorithm is as follows.

```
Select set of initial centroids, S
For each point in set of points P
      Assign P to nearest cluster C, where S_c is at a minimal
      distance.
For each cluster C
      Re-calculate centroid S_c as the average of all points in C
Repeat until convergence (or number of iterations is met)
```

While simple, the algorithm’s is subject to many questions. Firstly, how does one calculate the initial centroids? Secondly, what if these initial centroids are outliers? These two questions are the primary focus of the research conducted, as they greatly affect the accuracy of the centroids representing the underlying data.

As an initial approach to these questions, research was conducted on already explored methods for centroid selection. Many papers that explore alterations and applications of the K-Means clustering algorithm assume some basis of random centroid selection. The implementation is as simple as the

premise, and can be approached in a few different manners. To avoid selecting a centroid in a “gap” between the furthest outlying Euclidean points, the simplest approach is to randomly select N points from the set of points being clustered. These centroids will, after the first pass, be naturally altered to reflect a wider range of data This method, however, is sub-optimal, as the initial clusters may be unrepresentative of the actual data to huge degrees due to outliers being selected as cluster centroids. Naturally, more sophisticated implementations were researched.

Initial Clustering

Finding initial clusters depends entirely on the initial centroids selected, and because of this, it is desirable to choose strongly selected centroids from the start, so that the K-Means algorithm performs refinements on already (somewhat) representative data. In our research we have found two powerful approaches to this issue.

Pillar Approach

In their paper *A Pillar Algorithm for K-Means Optimization by Distance Maximization for Initial Centroid Designation*, Ali Ridho Barakbah and Yasushi Kiyoki explore the idea of selecting centroids in a manner that reflects all data. They assert that by selecting centroids that are found in the data, but are spread out over all instances, the resulting clusters calculated by the K-Means algorithm

will more accurately reflect global optimum, or optimal cluster configuration, rather than simply favoring distance refinements on local clusters.

Barakbah and Kiyoki argue that a strong initial clustering, when spread out, will work similarly to architectural foundations, in that centroids are best found through placement that results in broad and wide-ranged sets, not unlike “pillars” on a table or foundations in a building. Their approach is to therefore examine all instances in the dataset, and find points that are as far apart as possible from one another, without exceeding the bounds defined by the dataset. Furthermore, they determined that in situations where data is actually densely clustered, centroids that rely on random creation might neglect more loosely expressed clusters, should the centroids exist in the center. (Barakbah, 2009)

By selecting centroids on the outer edges of the dataset, clusters will refine from the outer edges inwards, resulting in a set of more representative clusters. One positive consequence that they note in their paper is that this method of selection accounts for all geometric shapes of data. Their paper goes further to also define methods of detecting outliers by examining the frequency of near-neighbors. When they finally implemented their algorithm and examined the accuracy of clusters using the Pillar approach versus clusters using other approaches, they found that their method was the most representative in multiple different cases.

The Refinement Approach

Another approach to initial centroid selection is refinement. Paul Bradley and Usama Fayyad of Microsoft's research division performed research on refinement on centroid selection by looking multiple sets of clusters that result after going through the K-Means algorithm with randomly selected initial centroids. They then perform a refinement technique over the "similar" clusters to find a cluster that would better represent the optimal configuration of clusters.

This approach of having multiple sub-samples of data, and using those to create cluster sets, is quick, and uses its breadth of data selection to minimize cluster inaccuracy due to outlier instances in the overall dataset. Their paper goes on to discuss further methods of refinement, accurate instance-set selection, as well as cluster clustering through examination of cluster entropy.

One problem they do note in their paper, however, is that at the time of their research, there was no defacto method of centroid selection, and their algorithm therefore relies on multiple sets of randomly selected initial configurations. While the effect of outlier selections diminishes as the number of randomly created clusters increases, it is still a persistent problem. (Bradley, 1998)

Exploring Centroid Selection – A Joint Approach

To better understand the impact of initial centroid selection, and the importance of the research done in this particular area regarding the K-Means clustering algorithm, we created a program that implements some of the centroid

selection methods discussed in this paper (user documentation is provided at the end of this paper). As a supplementary exercise, we decided to come up with our own method of centroid selection, which we call a pillar-web approach.

In researching these centroid selection methods, we identified potential problems in the previous two techniques – primarily that pillar selection may select outer centroids unsuccessfully if it identifies these outer points with too many outlier values. Furthermore, we also took into consideration the blatant flaws with random centroid selection, but looked at the benefits of finding refined centroids in multiple passes of random K-Means passes.

To address each of these issues, we devised an algorithm that will perform N random selection passes with K-Means, as well as implement the pillar approach. The initial configuration found using the pillar approach is used as a set of “base points” for the random centroid refinement. The algorithm then looks at which calculated clusters in the set of N random selection configurations are the most similar, and a centroid is calculated by using an average of these centroids – creating a refinement that examines N + 1 “similar” instances. The algorithm is defined as follows.

```
- Find R, a set of N resulting cluster configurations from
calculating K-Means with random centroid selection and 2 passes.

- Find P, a configuration of clusters from calculating K-Means
with pillar centroid selection and 2 passes.

- For each centroid in P
      - Initialize a centroid-sum as P's centroid
      - For each configuration C in R
            - Find Cluster in C with centroid closest to P's
            Add centroid to centroid-sum
      - Take average of the centroid-sum, adding it to the
      initial configuration

- Return initial configuration
```

Tests were performed on an example dataset[1] and the results are as shown below.

Results of Centroid Selection

Random Selection (5 iterations)

K	Average Objective Function
3	4.918
5	3.551
7	3.459

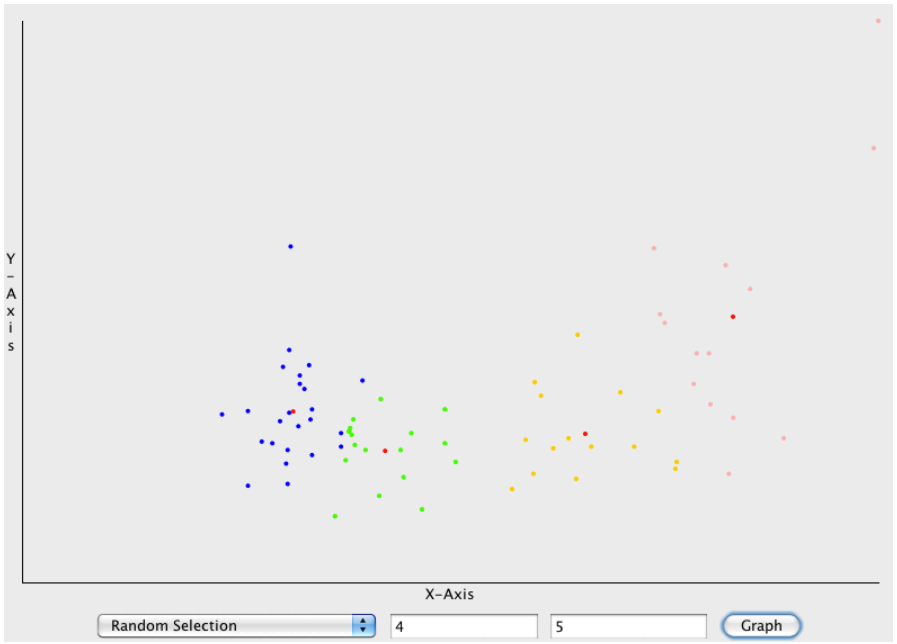
Pillar Selection (5 iterations)

K	Average Objective Function
3	4.589
5	3.137
7	2.517

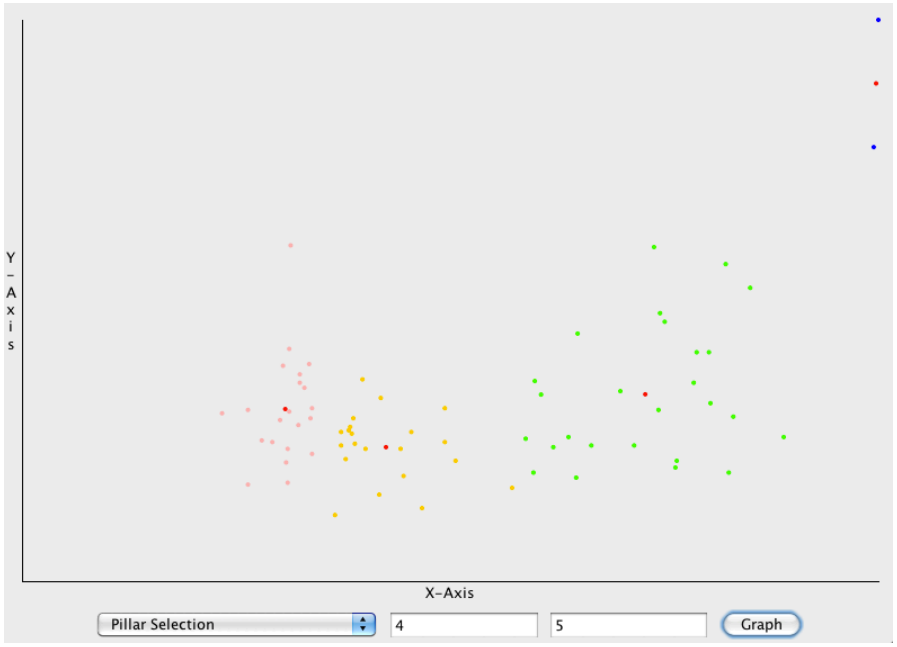
Pillar-Web Selection (5 iterations)

K	Average Objective Function
3	4.879
5	4.118
7	3.382

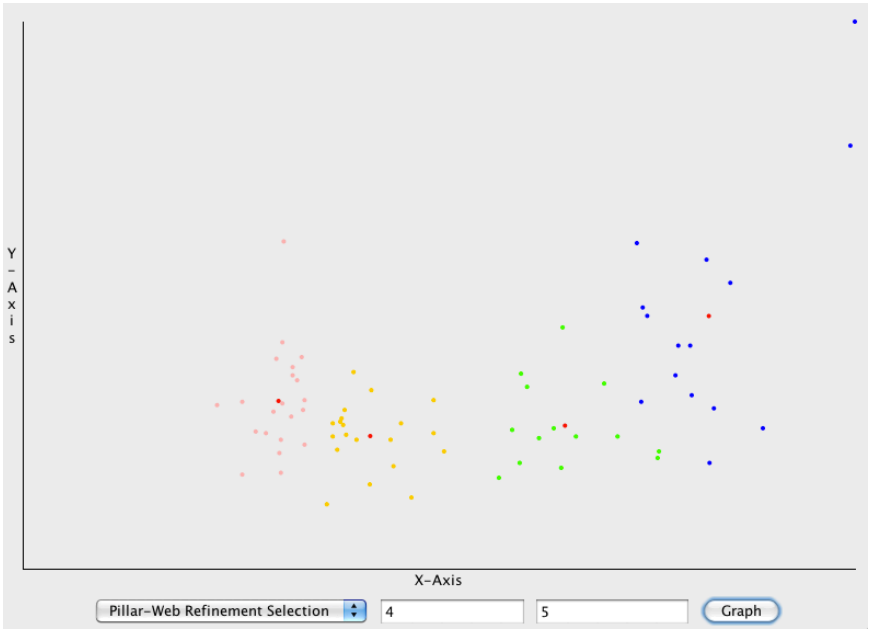
[1] The dataset is of the birth and death rates of 70 countries. The dataset can be found here: http://www.uni-koeln.de/themen/statistik/data/cluster/birth.dat



Random Selection (K = 4, N = 5)



Pillar Selection (K = 4, N = 5)



Pillar-Web Refinement Selection (K = 4, N = 5)

Observations

As expected, the pillar approach was the most accurate in our brief tests, and coincides with the findings described in Barakbah and Kiyoki's paper. Random selection and our custom Pillar-Web approach performed similarly, and our implementation did better in some cases. As seen in the third graph, representing our custom algorithm, our centroids better neglected the two stray values, effectively grouping them with other data instances. This is in contrast to the pillar approach, which created a new cluster for those instances. We speculate that the results would be more interesting with larger datasets (our testing dataset only used 70 instances) where using more clusters is more appropriate.

Applications of Clustering

K-Means, while simple, is extremely useful in the data-mining field, as it can give quick and accurate visualizations of data classifications. Furthermore, the algorithm is highly adaptable, and can be applied to many different situations where data needs to be represented visually. Current research done on K-Means clustering is more application-specific, however, as outlier identification rules, and centroid selection methods may differ a great deal depending on the domain. Because this clustering algorithm is such a powerful and diverse tool for those in the data-mining field, it's helpful to examine a few cases where K-Means clustering is being applied to today.

Case Study – GPA Prediction

In 2010, three computer scientists in Covenant University, Nigeria, published a paper on using K-Means clustering to effectively predict a student's Grade Point Average given other attributes. They assert that, due to the ambiguity of simply examining a GPA value, professors in higher level learning institutions are unable to examine key aspects of student performance, or be able to identify key factors that limit student performance. (Oyelade, 2010)

The paper asserts that a student's GPA value, and the difference in student GPA values, is a valid example of data where likeness can be expressed as Euclidean distance. To find the performance of any given grouping of students, the individual scores in the cluster were summed, and an average was

taken. Elements such as the number of students in each cluster were taken into consideration, and the results were compared to a deterministic model. By examining the Objective Function values of each cluster, they were able to rate the potential performance of the students in the cluster. The paper goes on to claim that their prediction methods using K-Means clustering were more accurate than previous models used. (Oyelade, 2010)

Case Study – Gene Expression Data

A thesis presented in 2009 by Dr. Chiang in the University of London goes over (in very great detail) variations on K-Means clustering algorithms that vary based on sampling methods on large datasets, applying filters on instances with many attributes. As an example application, Chiang references new research being performed by Prof. B. Chain, Virology Department at University College London, where their team has compiled two sets of genes and gene fragments where they compare normal gene expression in dendritic cells to gene expression in cancerous dendritic cells. (Chiang, 2009)

The problem, however, is that there are a lot of attributes associated with that type of data. Dr. Chiang exploits the disparity in calculated clusters between the various proposed methods in isolating sets of varying gene expression. The thesis goes on to assert that coupling K-Means clustering with smart k selection techniques, data normalization, and analysis techniques, it's very possible to

effectively analyze differences in gene expression between two samples with these data mining techniques. (Chiang, 2009)

Other Factors and Thoughts

The goal of this paper was to pinpoint a specific aspect of data mining and go into its current trends and research. The hope is that the simplicity of the K-Means clustering algorithm is now clear, but there are many factors that are not studied in this paper that need mentioning. In addition to initial centroid selection, K-selection, convergence rules, and outlier exclusion rules are big parts of creating a nicely tailored K-Means algorithm that best suits a particular application. Furthermore, there also exist many other methods of clustering that may work well compared to K-Means in certain scenarios. To conclude, it's best to note that it is the adaptability of the K-Means algorithm that promotes its usage as a highly tailored method of creating instance groupings in datasets.

User Manual for Clustering Program

Running the Program:

The program can be run using the following syntax

java KMeans <file_name> [--debug]

or

java -jar KMeans.jar <file_name> [--debug]

file_name – name of the datafile to use with the program

--debug – an optional argument that will send cluster assignment choices to standard output for analysis.

Notes on the file:

The file must have all of its instances in the following format

Instance_Name,x_coordinate,y_coordinate

Example: ALGERIA,36.4,14.6

Limitations on the program:

As it uses Euclidean distances, discrete values will be transformed into floating point continuous values. Furthermore, this program displays all data on a 2-D plane, and therefore can't handle more than two attributes.

Using the program:

It's got a nice GUI interface! Clusters have their own colors. Red is designated for centroid points. Note – centroids may not be actual instances depending on how many iterations, and what method used for centroid selection.

Work Cited

Barakbah, A. R, & Kiyoki, Y. A Pillar Algorithm for K-Means Optimization by Distance Maximization for Initial Centroid Designation. *IEEE*,

Bradley, P. S, & Fayyad, U. M (May 1998). Refining Initial Points for K-Means Clustering. *Proceedings of the 15th International Conference on Machine Learning*, pp. 91- 99.

Bramer, Max (2007). *Principles of Data Mining*. London: Springer.

Chiang, M. (June 2009). Intelligent K-Means Clustering in L2 and L1 Versions: Experimentation and Application. *University of London* ,

Deelers, S., & Auwatanamongkol, S. (2007). Enhancing K-Means Algorithm with Initial Cluster Centers Derived from Data Partitioning along the Data Axis with the Highest Variance. *International Journal of Electrical and Computer Engineering*, 2(4),

Lupsen, H. (05/12/2008). Data Sets for Clustering Techniques. Retrieved 04/22/2011, from http://www.uni-koeln.de/themen/statistik/data/cluster/

Oyelade, O. J, Oladipupo, O. O, & Obagbuwa, I. C (2010). Application of k-Means Clustering algorithm for prediction of Students' Academic Performance. *International Journal of Computer Science and Information Security*, 7(1),

Prasad, A. Parallelization Of K-Means Clustering Algorithm. *University of Colorado*