Real-time Clustering Algorithm Based on Predefined Level-of-Similarity

Rabindra Lamsal
rabind27@jnu.ac.in

Shubham Katiyar
shubha25@jnu.ac.in

School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi, 110067, India

Abstract
This paper proposes a centroid-based clustering algorithm which is capable of clustering data-points with n-features in real-time, without having to specify the number of clusters to be formed. We present the core logic behind the algorithm, a similarity measure, which collectively decides whether to assign an incoming data-point to a pre-existing cluster, or create a new cluster & assign the data-point to it. The implementation of the proposed algorithm clearly demonstrates how efficiently a data-point with high dimensionality of features is assigned to an appropriate cluster with minimal operations. The proposed algorithm is very application specific and is applicable when the need is perform clustering analysis of real-time data-points, where the similarity measure between an incoming data-point and the cluster to which the data-point is to be associated with, is greater than predefined Level-of-Similarity.

Keywords: Unsupervised learning, Clustering, Centroid, Real-time, Level-of-Similarity

1. Introduction
The main gist behind clustering is to group data-points into various groups (clusters) based on their features i.e. properties. The generation of clusters basically varies application wise (Xu and Wunsch, 2005); because it depends on what factors are to be taken into consideration to form a particular cluster. But, the focus of every clustering algorithm remains same i.e. to group similar data-points to a common cluster. The thing that differs is how this goal of forming a cluster is achieved. Different algorithms use different concept to deal with similarity measure among the data-points. There are many popular clustering algorithms (Hartigan, 1975) that group data-points based on various strategies to define the similarity measure between them. Centroid-based, connectivity-based, density-based, graph-based, etc are the commonly used strategies.

Often used algorithms like k-means, hierarchical clustering, DBSCAN, etc require a set of data-points in space beforehand. Without making fine adjustments to these pre-existing clustering algorithms, it is not possible to cluster the data-points in real-time. Hence, a real world problem exists if there is necessity of grouping data-points which are incoming to a system, in real-time. Besides, algorithm like k-means needs to specify number of clusters to
which the data-points are to be grouped into and also cannot identify clusters with arbitrary shape.

Hence, taking all these limitations into consideration, this paper proposes a different kind of machine learning algorithm that can group incoming data-points without need of specifying number of clusters to be formed. The main agenda behind proposing this algorithm is to facilitate those problems which require clustering of data-points based on some level-of-similarity, and most importantly in real-time. Without having to specify the number of clusters to be formed, the algorithm is capable of:

- grouping of incoming data-points in real-time,
- dealing with n number of features (high set of dimensionality),
- generating clusters based on a level-of-similarity (a similarity measure),
- identifying clusters with arbitrary shape.

There have been few approaches to generation of clusters of data in real-time. D-Stream (Chen and Tu, 2007) used a density-based approach to generate and adjust clusters in real-time. (Aggarwal et al., 2003a) divided the whole clustering process into two components: online and offline. Online component periodically stored detailed summary statistics and offline component facilitated an analyst to use variety of inputs including time horizon and number of clusters to be formed. CluStream (Aggarwal et al., 2003b) did clustering of large evolving data streams by viewing the stream of data as a changing process over time. Work by (Guha et al., 2000) maintained clustering of a sequence of data-points observed at a period of time.

The main focus of our work is to let an analyst to decide the level-of-similarity according as required, and leave the rest to the algorithm. The rest part of the paper is as follows. Section 2 describes all about the proposed algorithm - its core logic and blocks. Implementation of algorithm is done in section 3 and finally the work is concluded in Section 4 along with future directions.

2. The Proposed Algorithm

This algorithm requires initial declaration of cluster strictness. Cluster Strictness is the lowest permitted level of similarity between a data-point and the centroid of a cluster. Here, cluster centroid is the average of features of the data-points present in a cluster, and is an effective way of representing that particular cluster. If cluster strictness is set to a measure of 60; then the variability between a cluster centroid (C) and a data-point (D) is a measure of 40 at-most, if D is to be associated to cluster C.

Hence, this algorithm is capable of generating clusters based on a level-of-similarity. If one needs clusters with very low variance among the data-points, the cluster strictness can be adjusted accordingly; may be around 80-95%. That is why, cluster strictness plays a significance role in generating clusters and its value depends completely on what application the clustering is being done.
2.1 Data-point & Features

A data-point can have multiple features (n-features). Those features are the characteristics which collectively define a data-point. A stream of data-points can be grouped into various clusters, based on their features. Below is an example of a data-point with 7 features.

A data-point: 5 10 15 20 25 30 35

During implementation of the proposed algorithm, this pattern is used to represent a data-point and similarity measure between the data-points and cluster(s) is calculated accordingly. The algorithm takes data-points with n-features, one data-point at a time, and requires all the data-points to have same fixed number of features.

When the algorithm runs for the very first time, and is waiting for a new data-point; till this moment number of clusters is 0. Cluster strictness needs to be defined beforehand. Based on requirements, this value can be adjusted. Higher the value of cluster strictness, less is the variance between the data-points in a cluster. Depending on the data-points, using higher value of cluster strictness may result into more number of clusters. This is more likely to happen, if the incoming data-points are less identical with each other.

After assigning some value to cluster strictness, say 70%, the number of features that should be matched between a data-point and a cluster is calculated using the formula,

\[
\text{should_match_features} = \frac{\text{no_of_features} \times \text{cluster_strictness}}{100}
\]

Suppose, if the data-points that are to be clustered have 20 features. That means, cluster strictness with 70% results into 14 features that must be matched (at least) for a data-point to be associated to a particular cluster.

2.2 Shape & Size of Clusters

It is not possible to illustrate data-points with n-features (where n>2) in a 2D plane. For better understanding of size and shape of the clusters formed by the proposed algorithm, let us consider incoming data-points of only one feature. The centroid with small value has less coverage area than a centroid with larger value. Suppose, there are two clusters with centroid 10 and 100. With similarity measure set to 60, the cluster with centroid 10 accepts a new data-point with its feature between the range 6-14; while the cluster with centroid 100 assigns a new data-point with its feature lying between 60-140. This depicts the fact that cluster size varies based on value of its centroid.

Whenever a new data-point arrives, it’s similarity is checked with centroid of all the clusters. Since, the algorithm is centroid-based, with addition of a new data-point in a cluster, shifts the value of its centroid. The centroid of a cluster shifts to a particular direction, where the density of data-points is more concentrated. Hence, the algorithm is able to identify arbitrary shaped clusters as well.
Algorithm 1: Real-time Clustering Algorithm for Data-points with n Features

1. Declare cluster\_strictness.
2. Initialize cluster\_counter ← 0.
3. Calculate should\_match\_features using formula 1.
4. Read a datapoint \( D \)
5. If cluster\_counter = 0 then
   6. Increment cluster\_counter by 1.
   7. Create a new cluster \( C_{\text{cluster\_counter}} \).
   8. Assign \( D \) to the newly formed cluster.
   9. \( D \) be the centroid of the cluster.
   10. Else
       11. For all clusters \( C_i \), where \( i \) ranges from 1 to cluster\_counter do
           12. For all features \( F_j \), where \( j \) ranges from 1 to no\_of\_features do
               13. Calculate similarity\_measure using formula 2.
               14. If similarity\_measure\((C_i, C_j)\) ≥ (cluster\_strictness)
                   and similarity\_measure\((C_i, C_j)\) ≤\((100 + (100 - \text{cluster\_strictness}))\)
                   then
                     16. Increment matched\_features\_i by 1.
               17. End
           18. If matched\_features\_i ≥ should\_match\_features then
               19. Add \( C_i \) to the list of qualified\_clusters
           20. End
       21. End
       22. If qualified\_cluster list is empty then
           23. Increment cluster\_counter by 1.
           24. Create a new cluster \( C_{\text{cluster\_counter}} \).
           25. Assign \( D \) to the newly formed cluster.
           26. \( D \) be the centroid of the cluster.
       27. End
       28. If qualified\_cluster list contains single cluster then
           29. Assign \( D \) to the cluster.
           30. Re-calculate the centroid.
       31. Else
           32. Calculate cluster(s) with maximum matched\_features.
           33. If single cluster arises then
               34. Assign \( D \) to the cluster.
               35. Re-calculate the centroid.
           36. Else
               37. Find the cluster with max average of qualifying similarity measures.
               38. Assign \( D \) to the cluster.
               39. Re-calculate the centroid.
           40. End
       41. End
   42. End
43. End
2.3 Qualifying Feature

A feature qualifies to be counted as matched feature, when its similarity measure lies between the range: \((\text{cluster strictness}) \text{ and } (100 + (100 - \text{cluster strictness}))\). That is, the valid range is 70-130, when cluster strictness is considered 70. Suppose, when variability check of 5 and 13 is to be done with respect to 9, a basic mathematics can be used. 9 with respect to 9 gives a similarity measure of 100. 5 with respect to 9 gives a similarity measure of 55.56. And, 13 with respect to 9 gives a similarity measure of 144.44. If allowed variability is considered to be a measure of 50 then similarity measure of both 5 and 12 fall under the range 50-150, and are considered to be associated to 9 by at-least a measure of 50.

In case of data-points, following formula is used to calculate the similarity measure.

\[
similarity\_measure(C_i, F_j) = \frac{100 \times \text{datapoint}_j}{\text{centroid}(i, j)}
\]  

\(\text{where } i \text{ is an index for clusters, and } j \text{ is an index for features.}\)

As an example, in formula 2, \(i=3\) and \(j=7\) reflect the fact that cluster 3 is being considered, and 7th feature of a data-point & 7th feature of the centroid of cluster 3 are being used to compute the level of similarity between them. If the resulted similarity measure lies between the range \((\text{cluster strictness}) \text{ and } (100 + (100 - \text{cluster strictness}))\) then 7th feature is said to be matched, and the value of matched feature counter for the cluster 3 is incremented by 1. This process of calculating similarity feature of an incoming data-point is done with all the cluster(s). And if the matched feature counter for a cluster reaches the limit of minimum number of features that should be matched between a data-point and a cluster, that particular cluster is now placed into a list of qualified cluster.

2.4 Qualified Cluster List

This list contains the cluster(s) that satisfy the condition of having at least a minimum number of features that have matched with an incoming data-point. After calculation of
similarity measure for a data-point with all the cluster(s), there might arise any one of the following three situations.

2.4.1 Qualified cluster list is empty
Qualified list being empty indicates that no similar cluster(s) were found within the provided level of similarity. Hence, a new cluster is generated and the data-point is assigned to the newly formed cluster. The data-point is the centroid of this cluster.

2.4.2 Qualified cluster list contains exactly 1 cluster
In this case, the data-point is simply assigned to the cluster which is present in the list. And, the centroid of the cluster is re-calculated.

2.4.3 Qualified cluster list contains more than 1 cluster
Case 1: If the list contains more than 1 cluster, the cluster with maximum matched features is identified. The data-point is now assigned to the identified cluster. Case 2: Sometimes two or more cluster might come-up with the same highest number of matched features. In this case, the cluster with maximum average of qualifying similarity measures is identified, and the data-point is assigned to that cluster accordingly.

2.5 Conflicting Clusters
Whenever multiple clusters show up in qualified cluster list, while trying to identify the nearest similar cluster based on the highest number of matched features, those clusters are said to be the conflicting ones. In such case, an average of qualifying similarity measures is calculated for all the clusters which are in the qualified list and have same number of matched feature, and finally the cluster with the maximum average is considered to be the cluster to which the data-point is associated.

While calculating an average, only the qualifying similarity measures are being considered. One fact is also being taken into consideration that technically 60 and 140 are same with respect to 100. Both have variability measure of 40. Hence, if a similarity measure lies within the valid range, but is greater than 100 then the similarity measure is brought to the left side of the number system with reference to 100, using formula 3.

3. Implementation of the Algorithm
For best illustration of the implementation of the proposed algorithm, let us consider following 10 data-points, each of 10 features, which are to be clustered in real-time. The data-points are input to the algorithm in serial fashion.

| Data-point 1 | 10 15 20 25 30 35 40 45 50 55 |
| Data-point 2 | 9 35 18 45 10 32 60 41 10 20 |
| Data-point 3 | 18 13 18 27 30 38 41 49 57 |
| Data-point 4 | 20 20 18 5 15 34 50 43 10 50 |
| Data-point 5 | 17 17 18 15 22 35 44 43 10 53 |
Data-point 6: 10 32 20 45 12 55 40 55 9 25
Data-point 7: 22 15 20 7 10 40 25 50 50 60
Data-point 8: 200 300 400 500 600 250 350 450 550 650
Data-point 9: 250 350 450 550 550 200 400 200 100 300
Data-point 10: 10 20 25 40 50 40 50 60 20 50

Let us suppose, level-of-similarity for a cluster is considered to be 60. That means variability measure of 40 between a data-point and a cluster centroid is acceptable. With cluster strictness set to 60, using formula 1 we get minimum number features that must be matched to be 6.

For Data-point 1

Initially when Data-point 1 is input to the algorithm, because of absence of cluster(s), a new cluster C1 is created and Data-point 1 is assigned to it, and features of Data-point 1 is assumed the centroid of the C1.

\[ C1 = \text{Data-point 1} \]
\[ C1 \text{ centroid: 10 15 20 25 30 35 40 45 50 55} \]

For Data-point 2

With cluster C1

\[ \text{similarity}\_\text{measure}(C1, F1) = 90 \]
\[ \text{similarity}\_\text{measure}(C1, F2) = 233.34 \]
\[ \text{similarity}\_\text{measure}(C1, F3) = 90 \]
\[ \text{similarity}\_\text{measure}(C1, F4) = 180 \]
\[ \text{similarity}\_\text{measure}(C1, F5) = 33.34 \]
\[ \text{similarity}\_\text{measure}(C1, F6) = 91.43 \]
\[ \text{similarity}\_\text{measure}(C1, F6) = 150 \]
\[ \text{similarity}\_\text{measure}(C1, F6) = 91.11 \]
\[ \text{similarity}\_\text{measure}(C1, F6) = 20 \]
\[ \text{similarity}\_\text{measure}(C1, F6) = 36.36 \]

Number of qualified features came out to be 4 which is less than 6. C1 cannot be added to list of qualified cluster. Since, there are no any clusters in the list of qualified cluster, a new cluster C2 is generated and Data-point 2 is assigned to C2.

\[ C2 = \text{Data-point 2} \]
\[ C2 \text{ centroid: 9 35 18 45 10 32 60 41 10 20} \]

For Data-point 3

With cluster C1
Lamsal and Katiyar

\[ \text{similarity}\_\text{measure}(C_1, F_1) = 180 \]
\[ \text{similarity}\_\text{measure}(C_1, F_2) = 86.67 \]
\[ \text{similarity}\_\text{measure}(C_1, F_3) = 90 \]
\[ \text{similarity}\_\text{measure}(C_1, F_4) = 108 \]
\[ \text{similarity}\_\text{measure}(C_1, F_5) = 100 \]
\[ \text{similarity}\_\text{measure}(C_1, F_6) = 108.57 \]
\[ \text{similarity}\_\text{measure}(C_1, F_7) = 95 \]
\[ \text{similarity}\_\text{measure}(C_1, F_8) = 91.11 \]
\[ \text{similarity}\_\text{measure}(C_1, F_9) = 98 \]
\[ \text{similarity}\_\text{measure}(C_1, F_{10}) = 103.64 \]

Number of qualified features came out to be 9 which is greater than 6. C1 is added to list of qualified cluster.

For Data-point 3
With cluster C2

\[ \text{similarity}\_\text{measure}(C_2, F_1) = 200 \]
\[ \text{similarity}\_\text{measure}(C_2, F_2) = 37.14 \]
\[ \text{similarity}\_\text{measure}(C_2, F_3) = 100 \]
\[ \text{similarity}\_\text{measure}(C_2, F_4) = 60 \]
\[ \text{similarity}\_\text{measure}(C_2, F_5) = 300 \]
\[ \text{similarity}\_\text{measure}(C_2, F_6) = 118.75 \]
\[ \text{similarity}\_\text{measure}(C_2, F_7) = 63.33 \]
\[ \text{similarity}\_\text{measure}(C_2, F_8) = 100 \]
\[ \text{similarity}\_\text{measure}(C_2, F_9) = 490 \]
\[ \text{similarity}\_\text{measure}(C_2, F_{10}) = 285 \]

Number of qualified features came out to be 5 which is less than 6. C2 cannot be added to list of qualified cluster. Since, C1 is only the cluster in the list of qualified cluster, Data-point 3 is assigned to C1.

\( C_1 = \text{Data-point 1, Data-point 3} \)
\( C_1 \text{ centroid: 14 14 19 26 30 36.5 39 43 49.5 56} \)

For Data-point 4
With cluster C1

\[ \text{similarity}\_\text{measure}(C_1, F_1) = 142.86 \]
\[ \text{similarity}\_\text{measure}(C_1, F_2) = 142.86 \]
\[ \text{similarity}\_\text{measure}(C_1, F_3) = 94.74 \]
\[ \text{similarity}\_\text{measure}(C_1, F_4) = 19.23 \]
\[ \text{similarity}\_\text{measure}(C_1, F_5) = 50 \]
\[ \text{similarity}\_\text{measure}(C_1, F_6) = 93.15 \]
\[ \text{similarity}\_\text{measure}(C_1, F_7) = 128.21 \]
\[ \text{similarity}\_\text{measure}(C_1, F_8) = 100 \]
similarity\_measure(C1, F9) = 20.20
similarity\_measure(C1, F10) = 89.29

Number of qualified features came out to be 5 which is less than 6. C1 cannot be added to list of qualified cluster.

**For Data-point 4**
**With cluster C2**

\[
\begin{align*}
similarity\_measure(C2, F1) &= 222.22 \\
similarity\_measure(C2, F2) &= 57.14 \\
similarity\_measure(C2, F3) &= 100 \\
similarity\_measure(C2, F4) &= 11.11 \\
similarity\_measure(C2, F5) &= 150 \\
similarity\_measure(C2, F6) &= 106.25 \\
similarity\_measure(C2, F7) &= 83.33 \\
similarity\_measure(C2, F8) &= 104.87 \\
similarity\_measure(C2, F9) &= 100 \\
similarity\_measure(C2, F10) &= 250
\end{align*}
\]

Number of qualified features came out to be 5 which is less than 6. C2 cannot be added to list of qualified cluster. Since, there are no any clusters in the list of qualified cluster, a new cluster C3 is generated and Data-point 4 is assigned to C3.

C3 = Data-point 4
C3 centroid: 20 20 18 5 15 34 50 43 10 50

**For Data-point 5**
**With cluster C1**

\[
\begin{align*}
similarity\_measure(C1, F1) &= 121.43 \\
similarity\_measure(C1, F2) &= 121.43 \\
similarity\_measure(C1, F3) &= 94.74 \\
similarity\_measure(C1, F4) &= 57.69 \\
similarity\_measure(C1, F5) &= 73.33 \\
similarity\_measure(C1, F6) &= 95.89 \\
similarity\_measure(C1, F7) &= 112.82 \\
similarity\_measure(C1, F8) &= 100 \\
similarity\_measure(C1, F9) &= 20.20 \\
similarity\_measure(C1, F10) &= 94.64
\end{align*}
\]

Number of qualified features came out to be 8 which is greater than 6. C1 is added to list of qualified cluster.

**For Data-point 5**
**With cluster C2**
similarity\_measure(C2, F1) = 188.89
similarity\_measure(C2, F2) = 48.57
similarity\_measure(C2, F3) = 100
similarity\_measure(C2, F4) = 33.33
similarity\_measure(C2, F5) = 220
similarity\_measure(C2, F6) = 109.38
similarity\_measure(C2, F7) = 73.33
similarity\_measure(C2, F8) = 104.88
similarity\_measure(C2, F9) = 100
similarit\_measure(C2, F10) = 265

Number of qualified features came out to be 5 which is less than 6. C2 cannot be added to list of qualified cluster.

**For Data-point 5**

**With cluster C3**

similarity\_measure(C3, F1) = 85
similarity\_measure(C3, F2) = 85
similarity\_measure(C3, F3) = 100
similarity\_measure(C3, F4) = 300
similarity\_measure(C3, F5) = 146.67
similarity\_measure(C3, F6) = 102.94
similarity\_measure(C3, F7) = 88
similarity\_measure(C3, F8) = 100
similarity\_measure(C3, F9) = 100
similarity\_measure(C3, F10) = 106

Number of qualified features came out to be 8 which is greater than 6. C3 is added to list of qualified cluster. Since, C1 and C3 are in the list of qualified cluster and both the clusters have same number of qualified features. Now, average of the qualifying features is calculated for both the clusters. Qualifying features beyond 100 are brought to below 100 using the conversion formula:

\[
100 - (\text{value\_beyond\_hundred} - 100) \tag{3}
\]

Average of qualifying feature \((C1) = \frac{[(100 - (121.43 - 100)) + (100 - (121.43 - 100)) + 94.74 + 73.33 + 95.89 + (100 - (112.82 - 100)) + 100 + 94.64]}{8} = 87.87
\]

Average of qualifying feature \((C3) = \frac{[85 + 85 + 100 + (100 - (102.94 - 100)) + 88 + 100 + 100 + (100 - (106 - 100))]}{8} = 93.63
\]

Since, the average of qualifying features for C3 is greater, Data-point 5 is assigned to C3.

\(C3 = \text{Data-point 4, Data-point 5}\)

C3 centroid: \(18.5 \quad 18.5 \quad 18 \quad 10 \quad 18.5 \quad 34.5 \quad 47 \quad 43 \quad 10 \quad 51.5\)
For Data-point 6
With cluster C1

\[
\begin{align*}
\text{similarity \_measure}(C1, F1) & = 71.43 \\
\text{similarity \_measure}(C1, F2) & = 228.57 \\
\text{similarity \_measure}(C1, F3) & = 105.26 \\
\text{similarity \_measure}(C1, F4) & = 173.08 \\
\text{similarity \_measure}(C1, F5) & = 40 \\
\text{similarity \_measure}(C1, F6) & = 150.68 \\
\text{similarity \_measure}(C1, F7) & = 102.56 \\
\text{similarity \_measure}(C1, F8) & = 127.91 \\
\text{similarity \_measure}(C1, F9) & = 18.18 \\
\text{similarity \_measure}(C1, F10) & = 44.64
\end{align*}
\]

Number of qualified features came out to be 4 which is less than 6. C1 cannot be added to list of qualified cluster.

For Data-point 6
With cluster C2

\[
\begin{align*}
\text{similarity \_measure}(C2, F1) & = 111.11 \\
\text{similarity \_measure}(C2, F2) & = 91.43 \\
\text{similarity \_measure}(C2, F3) & = 111.11 \\
\text{similarity \_measure}(C2, F4) & = 100 \\
\text{similarity \_measure}(C2, F5) & = 120 \\
\text{similarity \_measure}(C2, F6) & = 171.88 \\
\text{similarity \_measure}(C2, F7) & = 66.67 \\
\text{similarity \_measure}(C2, F8) & = 134.15 \\
\text{similarity \_measure}(C2, F9) & = 90 \\
\text{similarity \_measure}(C2, F10) & = 125
\end{align*}
\]

Number of qualified features came out to be 9 which is greater than 6. C2 is added to list of qualified cluster.

For Data-point 6
With cluster C3

\[
\begin{align*}
\text{similarity \_measure}(C3, F1) & = 54.05 \\
\text{similarity \_measure}(C3, F2) & = 172.97 \\
\text{similarity \_measure}(C3, F3) & = 111.11 \\
\text{similarity \_measure}(C3, F4) & = 450 \\
\text{similarity \_measure}(C3, F5) & = 64.86 \\
\text{similarity \_measure}(C3, F6) & = 159.42 \\
\text{similarity \_measure}(C3, F7) & = 85.11 \\
\text{similarity \_measure}(C3, F8) & = 127.91 \\
\text{similarity \_measure}(C3, F9) & = 90
\end{align*}
\]
similarity_measure(C3, F10) = 48.54

Number of qualified features came out to be 5 which is less than 6. C3 cannot be added to list of qualified cluster. Since, C2 is only the cluster in the list of qualified cluster, Data-point 6 is assigned to C2.

C2 = Data-point 2, Data-point 6
C2 centroid:  9.5  33.5  19  45  11  43.5  50  48  9.5  22.5

For Data-point 7
With cluster C1

similarity_measure(C1, F1) = 157.14
similarity_measure(C1, F2) = 107.14
similarity_measure(C1, F3) = 105.26
similarity_measure(C1, F4) = 26.92
similarity_measure(C1, F5) = 33.33
similarity_measure(C1, F6) = 109.59
similarity_measure(C1, F7) = 64.10
similarity_measure(C1, F8) = 116.28
similarity_measure(C1, F9) = 101.01
similarity_measure(C1, F10) = 107.14

Number of qualified features came out to be 7 which is greater than 6. C1 is added to list of qualified cluster.

For Data-point 7
With cluster C2

similarity_measure(C2, F1) = 231.58
similarity_measure(C2, F2) = 44.78
similarity_measure(C2, F3) = 105.26
similarity_measure(C2, F4) = 15.56
similarity_measure(C2, F5) = 90.90
similarity_measure(C2, F6) = 91.95
similarity_measure(C2, F7) = 50
similarity_measure(C2, F8) = 104.17
similarity_measure(C2, F9) = 526.32
similarity_measure(C2, F10) = 266.67

Number of qualified features came out to be 4 which is less than 6. C2 cannot be added to list of qualified cluster.

For Data-point 7
With cluster C3
similar\_measure(C3, F1) = 118.92
similar\_measure(C3, F2) = 81.08
similar\_measure(C3, F3) = 111.11
similar\_measure(C3, F4) = 70
similar\_measure(C3, F5) = 54.05
similar\_measure(C3, F6) = 115.94
similar\_measure(C3, F7) = 53.19
similar\_measure(C3, F8) = 116.28
similar\_measure(C3, F9) = 500
similar\_measure(C3, F10) = 116.50

Number of qualified features came out to be 7 which is greater than 6. C3 is added to list of qualified cluster. Since, C1 and C3 are in the list of qualified cluster and both the clusters have same number of qualified features. Now, average of the qualifying features is calculated for both the clusters.

Average of qualifying feature (C1) = \([\frac{(100 - (107.14 - 100)) + (100 - (105.26 - 100)) + (100 - (109.59 - 100)) + 64.10 + (100 - (116.28 - 100)) + (100 - (101.01 - 100)) + (100 - (107.14 - 100))}{7}\) = 88.24

Average of qualifying feature (C3) = \([\frac{(100 - (118.92 - 100)) + 81.08 + (100 - (111.11 - 100)) + 70 + (100 - (115.94 - 100)) + (100 - (116.28 - 100)) + (100 - (116.50 - 100))}{7}\) = 81.76

Since, the average of qualifying features for C1 is greater, Data-point 7 is assigned to C1.

C1 = Data-point 1, Data-point 3, Data-point 7
C1 centroid: 16.67 14.33 19.33 19.67 23.33 37.67 34.33 45.33 49.67 57.33

For Data-point 8
With cluster C1

similar\_measure(C1, F1) = 1200
similar\_measure(C1, F2) = 2093.02
similar\_measure(C1, F3) = 2068.97
similar\_measure(C1, F4) = 2542.37
similar\_measure(C1, F5) = 2571.43
similar\_measure(C1, F6) = 663.72
similar\_measure(C1, F7) = 1019.42
similar\_measure(C1, F8) = 992.65
similar\_measure(C1, F9) = 1107.38
similar\_measure(C1, F10) = 1133.72

Number of qualified features came out to be 0 which is less than 6. C1 cannot be added to list of qualified cluster.
For Data-point 8
With cluster C2

\[
\begin{align*}
similarity\_measure(C2, F1) &= 2105.26 \\
similarity\_measure(C2, F2) &= 895.52 \\
similarity\_measure(C2, F3) &= 2105.26 \\
similarity\_measure(C2, F4) &= 1111.11 \\
similarity\_measure(C2, F5) &= 5454.55 \\
similarity\_measure(C2, F6) &= 574.71 \\
similarity\_measure(C2, F7) &= 700 \\
similarity\_measure(C2, F8) &= 937.5 \\
similarity\_measure(C2, F9) &= 5789.47 \\
similarity\_measure(C2, F10) &= 2888.89 \\
\end{align*}
\]

Number of qualified features came out to be 0 which is less than 6. C2 cannot be added to list of qualified cluster.

For Data-point 8
With cluster C3

\[
\begin{align*}
similarity\_measure(C3, F1) &= 1081.08 \\
similarity\_measure(C3, F2) &= 1621.62 \\
similarity\_measure(C3, F3) &= 2222.22 \\
similarity\_measure(C3, F4) &= 5000 \\
similarity\_measure(C3, F5) &= 3243.24 \\
similarity\_measure(C3, F6) &= 724.64 \\
similarity\_measure(C3, F7) &= 744.68 \\
similarity\_measure(C3, F8) &= 1046.51 \\
similarity\_measure(C3, F9) &= 5500 \\
similarity\_measure(C3, F10) &= 1262.14 \\
\end{align*}
\]

Number of qualified features came out to be 0 which is less than 6. C3 cannot be added to list of qualified cluster. Since, there are no any clusters in in the list of qualified cluster, a new cluster C4 is generated and Data-point 8 is assigned to C4.

C4 = Data-point 8
C4 centroid: 200 300 400 500 600 250 350 450 550 650

For Data-point 9
With cluster C1

\[
\begin{align*}
similarity\_measure(C1, F1) &= 1500 \\
similarity\_measure(C1, F2) &= 2441.86 \\
similarity\_measure(C1, F3) &= 2327.59 \\
similarity\_measure(C1, F4) &= 2796.61 \\
similarity\_measure(C1, F5) &= 2357.14 \\
\end{align*}
\]
Real-time Clustering Algorithm Based on Predefined Level-of-Similarity

\[
\text{similarity} \_\text{measure}(C_1, F_6) = 530.97 \\
\text{similarity} \_\text{measure}(C_1, F_7) = 1165.05 \\
\text{similarity} \_\text{measure}(C_1, F_8) = 441.18 \\
\text{similarity} \_\text{measure}(C_1, F_9) = 201.34 \\
\text{similarity} \_\text{measure}(C_1, F_{10}) = 523.26
\]

Number of qualified features came out to be 0 which is less than 6. C1 cannot be added to list of qualified cluster.

\textbf{For Data-point 9} \\
\textbf{With cluster C2}

\[
\text{similarity} \_\text{measure}(C_2, F_1) = 2631.58 \\
\text{similarity} \_\text{measure}(C_2, F_2) = 1044.78 \\
\text{similarity} \_\text{measure}(C_2, F_3) = 2368.42 \\
\text{similarity} \_\text{measure}(C_2, F_4) = 1222.22 \\
\text{similarity} \_\text{measure}(C_2, F_5) = 5000 \\
\text{similarity} \_\text{measure}(C_2, F_6) = 459.77 \\
\text{similarity} \_\text{measure}(C_2, F_7) = 800 \\
\text{similarity} \_\text{measure}(C_2, F_8) = 416.67 \\
\text{similarity} \_\text{measure}(C_2, F_9) = 1052.63 \\
\text{similarity} \_\text{measure}(C_2, F_{10}) = 1333.33
\]

Number of qualified features came out to be 0 which is less than 6. C2 cannot be added to list of qualified cluster.

\textbf{For Data-point 9} \\
\textbf{With cluster C3}

\[
\text{similarity} \_\text{measure}(C_3, F_1) = 1351.35 \\
\text{similarity} \_\text{measure}(C_3, F_2) = 1891.89 \\
\text{similarity} \_\text{measure}(C_3, F_3) = 2500 \\
\text{similarity} \_\text{measure}(C_3, F_4) = 5500 \\
\text{similarity} \_\text{measure}(C_3, F_5) = 2972.97 \\
\text{similarity} \_\text{measure}(C_3, F_6) = 579.71 \\
\text{similarity} \_\text{measure}(C_3, F_7) = 851.06 \\
\text{similarity} \_\text{measure}(C_3, F_8) = 465.12 \\
\text{similarity} \_\text{measure}(C_3, F_9) = 1000 \\
\text{similarity} \_\text{measure}(C_3, F_{10}) = 582.52
\]

Number of qualified features came out to be 0 which is less than 6. C3 cannot be added to list of qualified cluster.

\textbf{For Data-point 9} \\
\textbf{With cluster C4}
similarity_measure(C4, F1) = 125
similarity_measure(C4, F2) = 116.67
similarity_measure(C4, F3) = 112.5
similarity_measure(C4, F4) = 110
similarity_measure(C4, F5) = 91.67
similarity_measure(C4, F6) = 80
similarity_measure(C4, F7) = 114.29
similarity_measure(C4, F8) = 44.44
similarity_measure(C4, F9) = 18.18
similarity_measure(C4, F10) = 46.15

Number of qualified features came out to be 7 which is greater than 6. C4 is added to list of qualified cluster. Since, C4 is only the cluster in the list of qualified cluster, Data-point 9 is assigned to C4.

C4 = Data-point 8, Data-point 9
C4 centroid: 200 300 400 500 600 250 350 450 550 650

For Data-point 10
With cluster C1

similarity_measure(C1, F1) = 60
similarity_measure(C1, F2) = 139.53
similarity_measure(C1, F3) = 129.31
similarity_measure(C1, F4) = 203.39
similarity_measure(C1, F5) = 214.29
similarity_measure(C1, F6) = 106.19
similarity_measure(C1, F7) = 145.63
similarity_measure(C1, F8) = 132.35
similarity_measure(C1, F9) = 40.27
similarity_measure(C1, F10) = 87.21

Number of qualified features came out to be 6. C1 is added to list of qualified cluster.

For Data-point 10
With cluster C2

similarity_measure(C2, F1) = 105.26
similarity_measure(C2, F2) = 59.70
similarity_measure(C2, F3) = 131.58
similarity_measure(C2, F4) = 88.89
similarity_measure(C2, F5) = 454.55
similarity_measure(C2, F6) = 91.95
similarity_measure(C2, F7) = 100
similarity_measure(C2, F8) = 125
similarity_measure(C2, F9) = 210.53
similarity\_measure(C2, F10) = 222.22

Number of qualified features came out to be 6. C2 is added to list of qualified cluster.

**For Data-point 10**
**With cluster C3**

similarity\_measure(C3, F1) = 54.05
similarity\_measure(C3, F2) = 108.11
similarity\_measure(C3, F3) = 138.89
similarity\_measure(C3, F4) = 400
similarity\_measure(C3, F5) = 270.27
similarity\_measure(C3, F6) = 115.94
similarity\_measure(C3, F7) = 106.38
similarity\_measure(C3, F8) = 139.53
similarity\_measure(C3, F9) = 200
similarity\_measure(C3, F10) = 97.09

Number of qualified features came out to be 6. C3 is added to list of qualified cluster.

**For Data-point 10**
**With cluster C4**

similarity\_measure(C4, F1) = 4.44
similarity\_measure(C4, F2) = 6.15
similarity\_measure(C4, F3) = 5.88
similarity\_measure(C4, F4) = 7.62
similarity\_measure(C4, F5) = 8.69
similarity\_measure(C4, F6) = 17.78
similarity\_measure(C4, F7) = 13.33
similarity\_measure(C4, F8) = 18.46
similarity\_measure(C4, F9) = 6.15
similarity\_measure(C4, F10) = 10.53

Number of qualified features came out to be 0 which is less than 6. C4 cannot be added to list of qualified cluster. Since, C1, C2 and C3 are in the list of qualified cluster and all three of them have same number of qualified features. Now, average of the qualifying features is calculated for all the clusters.

Average of qualifying feature (C1) = \([60 + (100 - (139.53 - 100)) + (100 - (129.31 - 100)) + (100 - (106.19 - 100)) + (100 - (132.65 - 100)) + 87.21] / 6 = 73.255\)

Average of qualifying feature (C2) = \([(100 - (105.26 - 100)) + (100 - (131.58 - 100)) + 88.89 + 91.95 + 100 + (100 - (125 - 100))] / 6 = 86.5\)

Average of qualifying feature (C3) = \([(100 - (108.11 - 100)) + (100 - (115.94 - 100)) +
\[
\frac{(100 - (106.38 - 100)) + (100 - (139.53 - 100)) + 97.07}{6} = 71.185
\]

Since, the average of qualifying features for C2 is greater, Data-point 10 is assigned to C2.

\[
C2 = \text{Data-point 2, Data-point 6, Data-point 10}
\]

\[
\text{C2 centroid: 9.67 29 21 43.33 24 42.33 50 52 13 31.67}
\]

The data-points were successfully grouped into 4 various clusters based on similarity measure of their features. Cluster strictness of 60 was considered, hence permitted variability measure between a data-point and a cluster centroid was 40.

4. Conclusion

We presented the theoretical aspect of the proposed algorithm and implemented it on 10 data-points (all of them with 10-features) with supposition that they are input to the algorithm one after another, just like a stream-of-data in serial fashion. We also discussed how formation of clusters can be manipulated just by increasing/decreasing the level-of-similarity. The algorithm is applicable in real-world scenario where the task is to group a stream of data-points, incoming to a system, based on their similarity with the average of features (centroid) of data-points existing in their respective previously formed clusters. If none of the existing clusters satisfy the similarity measure, new clusters are formed and data-points are assigned to appropriate clusters accordingly.

5. Future Directions

In our literature review, we did not find any real-time algorithm which resembles to the working theme of application of this algorithm. It would definitely be meaning-less to test the performance of this algorithm with existing real-time clustering algorithms which collect data for a quantum of time and perform clustering analysis on the collected data. Hence, calculation of performance of this algorithm with relative to another clustering algorithm that groups data-points one at a time, would be the main future direction to work on.

When number of clusters in the space is relatively large, the complexity of algorithm also increases proportionately. Hence, it would be an interesting problem to try decreasing the number of similarity checks that is done when a new data-points arrives. Instead of checking the similarity of data-points with all the pre-existing clusters, an approach can be made so that only a certain number of clusters are taken into consideration for similarity check. This way the number of required computations can be decreased relatively.
Graphical representation of how 10 example data-points were clustered
Figure 2: Step-by-Step graphical representation of how incoming data-points were clustered

Acknowledgments

We would like to gratefully acknowledge Intel AI DevCloud for providing cloud compute for computational needs. Also, special thanks to Amit Sharma for his support in drafting this paper in LaTeX, and Rohan Negi for his healthy discussions during the overall period of this research work.

References

Charu C. Aggarwal, Jiawei Han, Jianyong Wang, and Philip S. Yu. A framework for clustering evolving data streams. In Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29, VLDB ’03, pages 81–92. VLDB Endowment, 2003a. ISBN 0-12-722442-4. URL http://dl.acm.org/citation.cfm?id=1315451.1315460.

Charu C. Aggarwal, Jiawei Han, Jianyong Wang, and Philip S. Yu. A framework for clustering evolving data streams. In VLDB, 2003b.

Yixin Chen and Li Tu. Density-based clustering for real-time stream data. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’07, pages 133–142, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-609-7. doi: 10.1145/1281192.1281210. URL http://doi.acm.org/10.1145/1281192.1281210.

S. Guha, N. Mishra, R. Motwani, and L. O’Callaghan. Clustering data streams. In Proceedings of the 41st Annual Symposium on Foundations of Computer Science, FOCS ’00, pages 359–, Washington, DC, USA, 2000. IEEE Computer Society. ISBN 0-7695-0850-2. URL http://dl.acm.org/citation.cfm?id=795666.796588.

John A. Hartigan. Clustering Algorithms. John Wiley & Sons, Inc., New York, NY, USA, 99th edition, 1975. ISBN 047135645X.

Rui Xu and D. Wunsch. Survey of clustering algorithms. IEEE Transactions on Neural Networks, 16(3):645–678, May 2005. ISSN 1045-9227. doi: 10.1109/TNN.2005.845141.