EXTENDING K-MEANS TO PRESERVE SPATIAL CONNECTIVITY

Sampriti Soor, Aditya Challa, Sravan Danda, B. S. Daya Sagar

Systems Science and Informatics Unit,
Indian Statistical Institute,
Bangalore, India.

Laurent Najman

ESIEE,
University Paris-Est,
Paris, France.

ABSTRACT

Clustering is one of the most important steps in the data processing pipeline. Of all the clustering techniques, perhaps the most widely used technique is K-Means. However, K-Means does not necessarily result in clusters which are spatially connected and hence the technique remains unusable for several remote sensing, geoscience and geographic information science (G(Sci) data. In this article, we propose an extension of K-Means algorithm which results in spatially connected clusters. We empirically verify that this indeed is true and use the proposed algorithm to obtain most significant group of waterbodies mapped from multispectral image acquired by IRS LISS-III satellite.

Index Terms— Clustering, Graphs, K-Means

1. INTRODUCTION

Clustering is one of the most important steps in data mining and machine learning [1, 2, 3]. Clustering has been used across several data domains ranging from text processing to hyperspectral images. Clustering in the context of image data is also referred to as segmentation. Thanks to its unsupervised nature, it can be used for several tasks such as simplifying and understanding the data, visualizing the important aspects etc. Due to the same reason, the solution to the problem of clustering is also extremely dependent on the domain of application. This requires adapting the existing methods to the domain, ensuring that the appropriate properties are preserved. In this article, we tackle one such problem - using K-Means while preserving spatial connectivity.

K-Means is one of the most widely used methods for clustering data [1, 4]. It is categorized under partition based methods and has some very important properties. It has been shown that using expectation maximization idea for clustering gaussian mixture models is closely related to the K-means method [3]. It is also very efficient and variations of k-means are also used for clustering big-data. K-means used along with map-reduce framework was proposed in [5]. However, for segmentation it is seen that k-means does not preserve the spatial connectivity of the final clusters. This property of connectedness of the clusters is important in the context of remotely sensed spatial images and hence K-Means cannot be directly used. For instance consider the image in figure 1(a). When simple K-Means is used in this case, we obtain the result as in figure 1(b), with number of clusters fixed at 10. Note that the final clustering is highly noisy. The reason for this is because, classical K-Means algorithm does not take into consideration the connectivity of the data.

In this article, we propose an algorithm which extends the classical K-Means algorithm to preserve the connectivity of the final clusters. The result of this algorithm on figure 1(a) is shown in figure 1(c). This is the main contribution of this article. The description of the algorithm is given in section 3, and the fact that the proposed algorithm is indeed an extension of K-Means is verified empirically on toy datasets. We then show an application of the proposed algorithm to obtain k most significant waterbodies mapped from multispectral image taken from IRS LISS-III satellite in section 4.

2. REVIEW

K-Means is a simple algorithm which is widely used in literature [6]. The algorithm starts with picking K initial points as the centroid and repeating the following steps until
convergence-

1. Form $K$ clusters by assigning every point to the closest of the $K$ initial points.

2. Recalculate the centers of the $K$ clusters.

Assume that we have a clustering $C = \{C_1, C_2, \cdots, C_K\}$, then, squared sum of errors (SSE) is defined by

$$SSE(C) = \sum_{k=1}^{K} \sum_{x_i \in C_i} d(x_i, c_i)$$

(1)

where $x_i$ denotes the points, $c_i$ denotes the center of the cluster $C_i$ and $d(\cdot, \cdot)$ denotes the distance metric. It can be shown that K-Means algorithm works by monotonically reducing the $SSE$ error function until it reaches a local minima. (See chapter 4 of [3] for details.)

In this article, we aim to extend the K-Means technique with an additional constraint of resulting in clusters which are connected. This requires specifying the adjacency relation in the problem statement, thus allowing to maintain the connectivity of the clusters. We achieve this by using the framework of the edge-weighted graphs.

An edge weighted graph $G = (V, E, W)$ is a tuple with three sets - a set of nodes/vertices where each node/vertex denotes each data point, a set of edges which defines the adjacency relation on the given data and a function $W : E \rightarrow \mathbb{R}^+$ which denotes the weight of each edge.

Also, one of the important steps of the K-Means algorithm is to calculate the ‘mean’ of the objects in the cluster. This however, is not always possible in general since objects need not necessarily belong to a space where averages are defined. This is especially common in the field of geoscience and remote sensing. For instance, consider the problem of clustering of water bodies (discussed in [7]), or more abstractly sets in the euclidean space. Thus in this article we consider a variant of the K-Means algorithm, known as K-Medoids (see chapter 4 of [3] for more details) where the center is taken to be one of the data points in the set.

### 3. Extending K-Means to Edge Weighted Graphs

Thus, the problem of clustering becomes minimizing the optimization problem in (1) subject to the constraint that each cluster $C_i$ is connected with respect to the adjacency relation given by $E$. To solve this minimization problem, we propose the algorithm 1.

The algorithm proceeds similarly to the K-Means algorithm and starts with picking out $K$ random seeds from the data. At each stage, all the nodes are assigned to the nearest reachable seed, that is there exists a path between the node and a seed and all the nodes in the path are assigned to this seed. This is achieved by using a priority queue, where the priority is given by the distance to the node. These priorities are updated lazily to ensure efficient implementation. This ensures the connectedness of the component once all the nodes are processed as described by the proposition 1.

**Proposition 1.** For every iteration between steps 3-12 in algorithm 1, the connectedness of the cluster is preserved.

The seeds are then recalculated. For each component, the new seed is taken to be the center of the subgraph. Given the distance $d(\cdot, \cdot)$, the center of the graph is defined as

$$\arg\min_x \sum_u d(x, u)$$

(2)

### Algorithm 1 K-means with connectivity constraint

**Input:** An edge weighted graph, $G = (V, E, W)$

**Output:** $C = \{C_1, C_2, \cdots, C_K\}$ - $K$ clusters

1. Pick $K$ random points from the set of nodes $V$ as initial seeds.
2. **while** Convergence is not reached **do**
3. Initialize a priority queue - $Q$ and a set data structure $P$. $P$ indicates the set of pixels already processed.
4. Push all the neighbors of the seeds into $Q$, where priority is given by the distance to the node
5. **while** $Q$ is not empty **do**
6. Pop the vertex $v$ from $Q$
7. if $v$ is not in $P$ **then**
8. Assign it to the nearest seed and add it to the set $P$.
9. for each neighbor $u$ of $v$ **do**
10. Push the neighbor $u$ into $Q$, with priority given by

$$\text{priority}(u) = W(v, u) + \text{priority}(v)$$

11. **end for**
12. **end if**
13. **end while**
14. Update the centers of each cluster with the node that minimizes the largest distance.
15. **end while**

---

1 A complete graph is a graph in which any two nodes are adjacent.
the dilation distance is defined as $d$ distance between the waterbodies. Given two sets $S_1$ and $S_2$, the cluster. For this we use the dilation distance \cite{7} as the 
waterbody which is placed closest to all other waterbodies in 
from the data. The significant waterbody is defined as the 
identify the waterbody which is spatially significant. How-

As an application of the proposed algorithm we consider wa-
terbodies data taken from \cite{7}. In \cite{7}, this data was used to 
identify the waterbody which is spatially significant. How-

Remark: The proposed algorithm can be seen as an it-
erated version of the classical watershed algorithm used in 
image segmentation \cite{8}, where the seeds of the watershed are 
updated after each watershed step.

4. APPLICATION

As an application of the proposed algorithm we consider wa-
ter bodies data taken from \cite{7}. In \cite{7}, this data was used to 
identify the waterbody which is spatially significant. How-

In figure 3(a) we have the multispectral data from which 
the waterbodies are mapped as in figure 3(b). The question 
of interest is to identify the most significant $k$ waterbodies 
from the data. The significant waterbody is defined as the 
waterbody which is placed closest to all other waterbodies in 
the cluster. For this we use the dilation distance \cite{7} as the 
distance between the waterbodies. Given two sets $S_1$ and $S_2$, 
the dilation distance is defined as

$$d(S_1, S_2) = \inf\{\lambda \mid S_1 \oplus \lambda B \supseteq S_2\}$$  

where $B$ is a unit disk structuring element. The adjacency 
relation is obtained by the influence zones of figure 3(b), as 
shown in figure 3(c). Two waterbodies are considered adja-
cent, if their influence zones are adjacent in figure 3(c). Using 
these parameters, significant waterbodies for $k = 2, 4, 6$ are 
calculated as shown in figure 3 (d)-(f).

5. CONCLUSION AND FUTURE WORK

In summary, we have proposed an algorithm which extends 
the K-Means to obtain clusters which are connected. This is 
achieved by considering the framework of edge graphs. The 
algorithm obtained was shown to give consistent results with 
a variant of K-Means in the degenerate case of considering 
a complete graph. This was then applied to the waterbodies 
data from \cite{7} to obtain $k = 2, 4, 6$ significant waterbodies.

In future, on the theoretical front, we hope to obtain a rig-
orous proof of equivalence between algorithm 1 and K-Means 
technique. On the application front we expect to use the al-
gorithm proposed to various other data and perform extensive 
validation.

6. ACKNOWLEDGMENTS

The first three authors would like to thank Indian Statisti-
cal Institute for the funding provided. BDS would like 
to acknowledge the support received from the Science and 
Engineering Research Board (SERB) of the Department 
of Science and Technology (DST) with the grant number 
EMR/2015/000853, and the Indian Space Research Organiza-
tion (ISRO) with the grant number ISRO/SSPO/Ch-1/2016-
17.

7. APPENDIX : PROOF OF PROPOSITION 1

Proof. We only provide the idea of the proof here. Let $S = \{s_i\}$ indicate the seeds and $d(u, w)$ indicate the distance bet-
 tween $u$ and $w$. We assume that the distance metric used sat-
fies the regularity conditions mentioned in \cite{9}. From the 
algorithm it is clear that a node $u$ is assigned to the seed $s_i$ 
which minimizes $d(u, s_i)$.

We show that if $u$ is assigned to the seed $s_i$, then the nodes 
in the shortest path $< u, s_i >$ are also assigned to $s_i$. Assume 
for the sake of contradiction that $v$ is assigned to $s_j$, $j \neq i$. 
This implies that $d(v, s_j) < d(v, s_i)$. Thus,

$$d(u, s_i) = d(u, v) + d(v, s_i)$$  
> $$d(u, v) + d(v, s_j)$$  
> $$d(u, s_j)$$

Hence we get a contradiction. So, all the nodes in the shortest 
path belong to the component, and hence it is connected. \square
Fig. 3. Application of algorithm 1 to waterbodies clustering. Top row: (a) Multispectral image acquired by IRS LISS-III satellite. Blue objects indicate the waterbodies. (b) Waterbodies mapped from the image in (a). (c) Influence zones of the waterbodies in (b). Bottom row: Clustering obtained using algorithm 1. The red color influence zone indicates seed of the cluster, which is also the most significant waterbody in that cluster. The number of clusters taken are - (d) $k=2$ (e) $k=4$ (f) $k=6$

8. REFERENCES


