Introduction to Clustering in Machine Learning

Shrikant patro

School of computer science and engineering, VIT Chennai, Tamil Nadu, India 600 127
Email: Shrikantjagannath.2018@vitstudent.ac.in

Abstract

The Machine learning is defined as an application of Artificial Intelligence that provide system the ability to automatically learn and improve from experience without explicitly programmed. This process is broadly classified into three category Supervised Learning, Unsupervised Learning and Semi-Supervised learning. Clustering falls into the category of Unsupervised learning where the class label is unknown. K-Means Clustering is one of the popular form of clustering. In this paper we will have brief discussion on the working and results of the K-Means clustering. Also, we will look for hierarchical clustering that include Agglomerative hierarchical clustering (AGNES) and Divisive Hierarchical clustering (DIANA). This paper also have references to the implementation of the algorithm on the Blood Transfusion module of Woman Fertility dataset.

Introduction

Clustering in machine learning is the assignment of a set of observation into subsets (Cluster) so that observation of the cluster are similar in some sense. Clustering is one of the method of unsupervised learning, and a common technique of Statistical data analysis in many field. Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data.
A loose definition of clustering could be "the process of organizing objects into groups whose members are similar in some way". A cluster is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters.

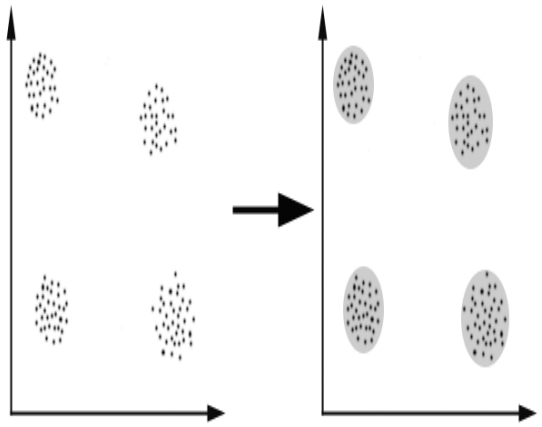


Fig 1. Basic Clustering of Data points

In this case we easily identify the 4 clusters into which the data can be divided; the similarity criterion is distance: two or more objects belong to the same cluster if they are "close" according to a given distance (in this case geometrical distance). This is called distance-based clustering.
Another kind of clustering is conceptual clustering: two or more objects belong to the same cluster if this one defines a concept common to all that objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures. Clustering is so

popular that Clustering can be applied in many filed. This include Marketing, Biology, Libraries, Insurance, City Planning, Earthquake studies and Document classification.

Clustering Algorithm are broadly classified as Exclusive Clustering, Overlapping Clustering, Hierarchical clustering and Probabilistic Clustering. This explore more on the Hierarchical clustering methods. This Hierarchical method can be classified into Agglomerative and Divisive clustering. In this Agglomerative clustering is described as the clustering process that begin from the bottom level where all the node are considered as individual entity. Then, they are grouped slowly from small cluster size to larger cluster. When the end of clustering is reached entire data point is considered as single cluster. In Divisive clustering Algorithm the clustering process begin from Top and then fragmented into much small cluster size. Till the end each member is single cluster. Hierarchical clustering has no backtracking, if particular merge or split turn out to be poor choice, then it cannot be corrected. AGNES and DIANA are example of Agglomerative clustering. AGNES include Cluster C1 and C2 may be merged if an object in C1 and C2 form the minimum Euclidean distance any two object from different cluster. DIANA, a cluster is split according to some principle. E.g , the maximum Euclidean distance between the closest point in the cluster. The Agglomerative and Divisive algorithm are represented using dendogram.

Methodology:

The clustering performed so far are distance based. So, the calculation of the distance takes place as follows:

$$d_{\min}(C_i, C_j) = \min_{p \in C_i, p' \in C_j} |p - p'|$$

|p-p'| is the distance between two objects p and p'. An algorithm that uses the maximum distance to measure the distance between clusters is called sometimes farthest-neighbor clustering algorithm. If the clustering process terminates when the maximum distance between nearest clusters exceeds an arbitrary threshold, it is called complete-linkage algorithm.

Algorithm for the K-Means Clustering algorithm:

Step 1. Select the number of the cluster. Also, select the same number of the centroid for the cluster.

Step 2. Find the distance between the unlabeled point and the choses cluster center. The nearest cluster center is labeled as new label for unlabeled data point.

Step 3. Update the cluster centroid based on the number of the point in the cluster.

Step 4. Repeat step 2 to 3 multiple time, until the number point converge at only label.(i.e) no change between this iteration and previous iteration.

Algorithm for the Agglomerative clustering

Step 1. Initially all the data point on the surface is an independent entity. The number of cluster id same as number of data point.

Step 2. Choose the source point find the distance between to all other point. Select the point with the least distance between them and combine them to form single cluster.

Step 3. New cluster form will be treated as independent entity. Find the distance

between this groups and combine them based on the proximity of the cluster.

Step 4. Repeat step 2 and 3 until the all the data point are part of single large cluster. The algorithm can be stopped if sufficient amount of the cluster is formed.

Algorithm for the Divisive hierarchical clustering

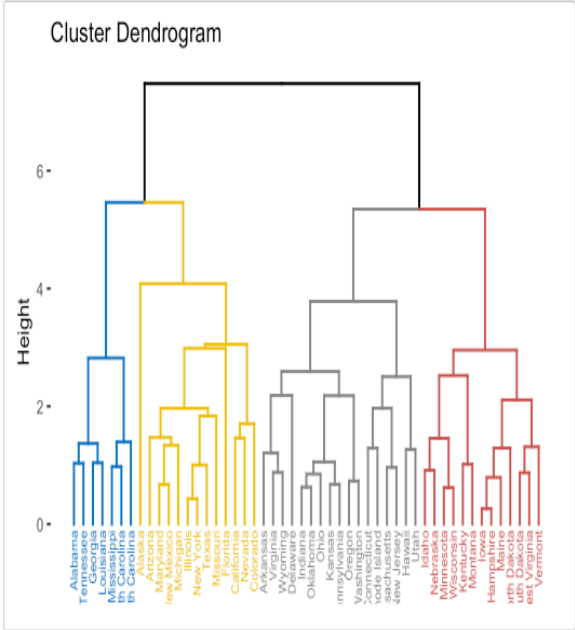
Step 1. Initially the algorithm will treat all the data point as one group or single cluster.

Step 2. Then it will find maximum distance between the points that belong to same cluster. It will separate out the given data point from the current cluster.

Step 3. Repeat step 2. Multiple until all the data point are divided into single entity.

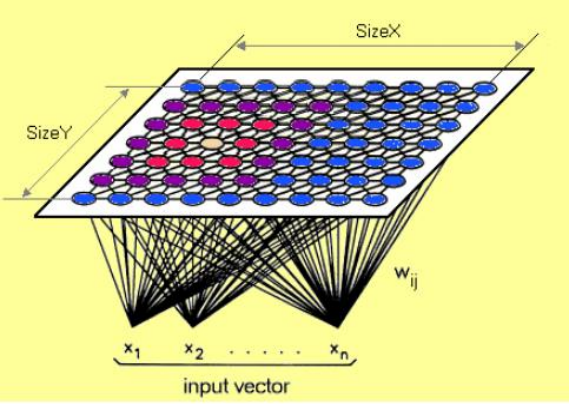
Step 4. Finally, all the data point are belonging to their own cluster. But, the stopping criteria can be specified quite before it will stop on its own.

The dendogram is very popular visualization technique used for the representation of the hierarchical clustering. The two end point represent the element to be combined while the inverted 'U' shape represent they are part of one cluster. The figure 2 below represent the demonstration about the look and arrangement of the element in the dendogram.



Self- organizing Map (SOM)

A self-organizing map **(**SOM**)** is a type of artificial neural network (ANN) that is trained using unsupervised learning to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a **map**, and is therefore a method to do dimensionality reduction. Self-organizing maps differ from other artificial neural networks as they apply competitive learning as opposed to error-correction learning (such as backpropagation with gradient descent), and in the sense that they use a neighborhood function to preserve the topological properties of the input space.



Dimensionality reduction in SOM

Fig 3. Dimensionality reduction in SOM

Each data point in the data set recognizes themselves by competeting for representation. SOM mapping steps starts from initializing the weight vectors. From there a sample vector is selected randomly and the map of weight vectors is searched to find which weight best represents that sample. Each weight vector has neighboring weights that are close to it. The weight that is chosen is rewarded by being able to become more like that randomly selected sample vector. The neighbors of that weight are also rewarded by being able to become more like the chosen sample vector. This allows the map to grow and form different shapes. Most generally, they form square/rectangular/hexagonal/L shapes in 2D feature space.

The Algorithm for SOM:

1. Each node's weights are initialized.
2. A vector is chosen at random from the set of training data.
3. Every node is examined to calculate which one's weights are most like the input vector. The winning node is commonly known as the **Best Matching Unit** (BMU).
4. Then the neighborhood of the BMU is calculated. The amount of neighbors decreases over time.
5. The winning weight is rewarded with becoming more like the sample vector. The neighbors also become more like the sample vector. The closer a node is to the BMU, the more its weights get altered and the farther away the neighbor is from the BMU, the less it learns.
6. Repeat step 2 for N iterations.

Best Matching Unit is a technique which calculates the distance from each weight to the sample vector, by running through all weight vectors. The weight with the shortest distance is the winner. There are numerous ways to determine the distance, however, the most commonly used method is the Euclidean Distance, and that's what is used in the following implementation.

Conclusion

Clustering is one of the popular machine learning technique suitable for the unlabeled data. The popular among them are distance based, density based and hierarchical clustering algorithm. The hierarchical clustering algorithm basically work with two different approach top to bottom (Divisive) and bottom to top (Agglomerative). The SOM is very popular to reduce the dimension of the large dataset and perform clustering. Few disadvantage of the of SOM based clustering are It does not build a generative model for the data, i.e, the model does not understand how data is created, It does not behave so gently when using categorical data, even worse for mixed types data, The time for preparing model is slow, hard to train against slowly evolving data.

References:

1. http://www.inf.unibz.it/dis/teaching/DWDM/slides2010/lesson9-Clustering.pdf
2. https://scikitlearn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html
3. https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/
4. http://mines.humanoriented.com/classes/2010/fall/csci568/portfolio_exports/mvoget/cluster/cluster.html
5. https://github.com/SSQ/Coursera-UW-Machine-Learning-Clustering-Retrieval/tree/master/Week%206%20PA%201
6. https://www.tutorialspoint.com/python_pandas/python_pandas_dataframe.htm
7. https://www.datanovia.com/en/lessons/divisive-hierarchical-clustering/
8. https://towardsdatascience.com/self-organizing-maps-ff5853a118d4
9. https://github.com/abhinavralhan/kohonen-maps
10. https://github.com/abhinavralhan/kohonen-maps/blob/master/somoclu-iris.ipynb