Department of Computer Science and Engineering
Indian Institute of Technology,Kharagpur

Machine Learning (CS60050)
Spring 2020-2021
Instructor: Dr. Aritra Hazra

Lecture Date: 11^{th} March 2021, Thursday
Scribed by: Anushruti Pandey and Sontu Mistry

March 19, 2021

Summary of Last Class

We have started with the Clustering in Unsupervised Learning and discussed the terminologies and algorithms for the same. To summarize in brief:

- When we only have data points but not the labels we try to cluster the data points in some clusters depending upon the association among the points, which is the essence of *Unsupervised Learning*.
- A good clustering algorithm should ensure very good cohesion among the point of one cluster and good separation among the other cluster points. For this purpose we have declared certain distance matrix with in cluster which satisfies Symmetry, Reflexivity and Triangular inequality. And we have also learned about different distance matrix across cluster which is mean-distance, maximum-distance, and average-distance between two points.
- Then we have also learned two types of clustering technique – *Hierarchical Clustering* and *Partitional clustering*. Hierarchical clustering can further be divided into two categories based on whether we created them top-down or bottom-up manner. In top-down or *Divisive Clustering* we initially consider all the points as a single cluster and then break them down into smaller cluster, whereas in bottom-up or *Agglomerative Clustering* considers all the points as separate clusters initially then depending upon the closeness of the points clusters are merged together. We have also seen hierarchical clustering can be easily visualized using a tree kind of structure called *Dendrogram*. If we need k-clusters we just cut k-1 depths in the dendrogram tree.
- Depending upon the inter-cluster distance we have used in agglomerative clustering can further be divided into three types single, average and complete linkage. If we use minimum distance between two points then it is called *Single Linkage*, if we use maximum distance between two points then it is called *Complete Linkage* and if we use mean distance between two points then it is called *Average Linkage*.
- In partitional clustering we have learned about the K-Means clustering algorithm, where we already have a predefined 'k' number of clusters or buckets. We then arbitrarily chosen some points inside the bucket and based on proximity of other points, we keep on adding points to the buckets. Once we put all the points into the buckets we calculate the mean of the bucket and re-formulate the

bucket several time until and unless we see no change in bucket configuration in two consequent iterations.

Although K-Means algorithm computationally efficient and have good convergence rate, the center of the clusters are dragged toward the outlier points. This very problem can easily be solved if we use a density based algorithm, which is indeed the topic of today's class.

DBSCAN

DBSCAN or *Density-Based Spatial Clustering of Applications with Noise*, as the name suggests it is a clustering algorithm which use density of points for clustering and able to detect and discard outlier or noise in the data. Before going into the algorithm let's see the main two parameters of the algorithm:

- **MinPoint (mp)** : Minimum numbers of points which is needed to be present into a neighbourhood area of a point to declare it as a core point.
- **Epsilon** (ϵ): Radius of a neighbourhood area.

With the above information let us also define some terminologies:

- **Dense / Core point**: A point is called Dense or core point if there are at least mp number of points in the surrounding area with radius.
- **Border point**: If area with radius around a point does not contain mp number of points but contain at least one core point then the point is considered as border point.
- **Noise Point**: If a point is neither core point nor border point then it is a noise point.

Finally, let us define the concept of **connectedness**:

- Single core point (x_i) is always connected.
- All **directly connected point** i.e., points (x_j) present in radius of a core point (x_i) is connected to that core point (x_i).
- Two point x_i and x_j are connected if they are path connected, i.e., if there exist some points x_{k1}, x_{k2}, ..., x_{kn} such that $x_i \to x_{k1} \to x_{k2} \to ... \to x_{kn} \to x_j$. Where a $\to$ b refers to a is connected to b.

Finally, with all the information lets define the DBSCAN algorithm:

Algorithm

1. All points path connected to a core point form a cluster.
2. All other remaining points are noise or outliers.

Advantages

1. It can discover clusters of different size and shape.
2. It can detect and discard the noise or outlier.
3. Unlike K-means clustering algorithm, it does not require a predefined cluster number.

Disadvantages

1. Cannot be applied to higher dimensionality.
2. If point density is not uniform, then this algorithm doesn't work well.

Hybrid Clustering

So far, we have seen various partitional and hierarchical clustering algorithm. However both of them have their own merits and demerits. Hybrid clustering tries to combine the pros of both under the hood. Single Linkage K-means algorithm is an example of a hybrid clustering which works as follows:

1. Partition all the point into large number of cluster(k) with a partitional algorithm k-means.
2. Combine all these k clusters into k' clusters with agglomerative clustering.

Since, this algorithm uses both the concepts of partitional as well as hierarchical, so it is called as a *Single Linkage K-Means* and is also known as CLARA Algorithm.
There can be many other hybrid approaches.

Experimental Purpose of Clustering

- The most appropriate clustering algorithm for a particular problem often needs to be chosen experimentally, unless there is a mathematical reason to prefer one cluster model over another.
- In clustering, the hyperparameter choices are the distance metrics, to calculate the distance between points and the numbers of clusters. So, to obtain the best optimal clusters, the clustering algorithm is iterated some particular number of times. For *Agglomerative Clustering*, we will get the same clusters every time for a particular k but in *k-means*, it depends on the seed values and hence its goodness of fit should be measured.

So, the one of the metrics to measure the accuracy and goodness of our clustering technique is Silhouette Coefficient. This index works well with k-means clustering, and is also used to determine the optimal number of clusters.

Silhouette Coefficient

The *Silhouette Value* is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation). *Cluster cohesion* is the sum of the weight of all links within a cluster and *Cluster separation* is the sum of the weights between nodes in the cluster and nodes outside the cluster.

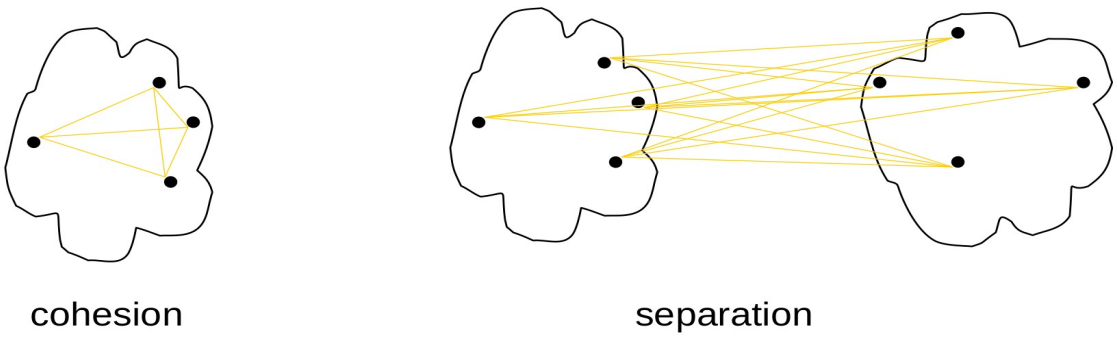

Figure 1: Cohesion and Separation

Assume the data have been clustered into m clusters, $C_1, C_2, \ldots, C_m$.
The cohesion can be defined for each data point $i \in C_i$, that is data point i in the cluster C_i as:

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j) \tag{1}$$

The above given equation gives the mean distance between i and all other datapoints in the same cluster, where $d(i, j)$ is the distance between the datapoints i and j in the cluster C_i, (we divide by $|C_i| - 1$

because we do not include the distance $d(i, i)$ in the sum).
We can interpret $a(i)$ as a measure of how well i is assigned to its cluster.

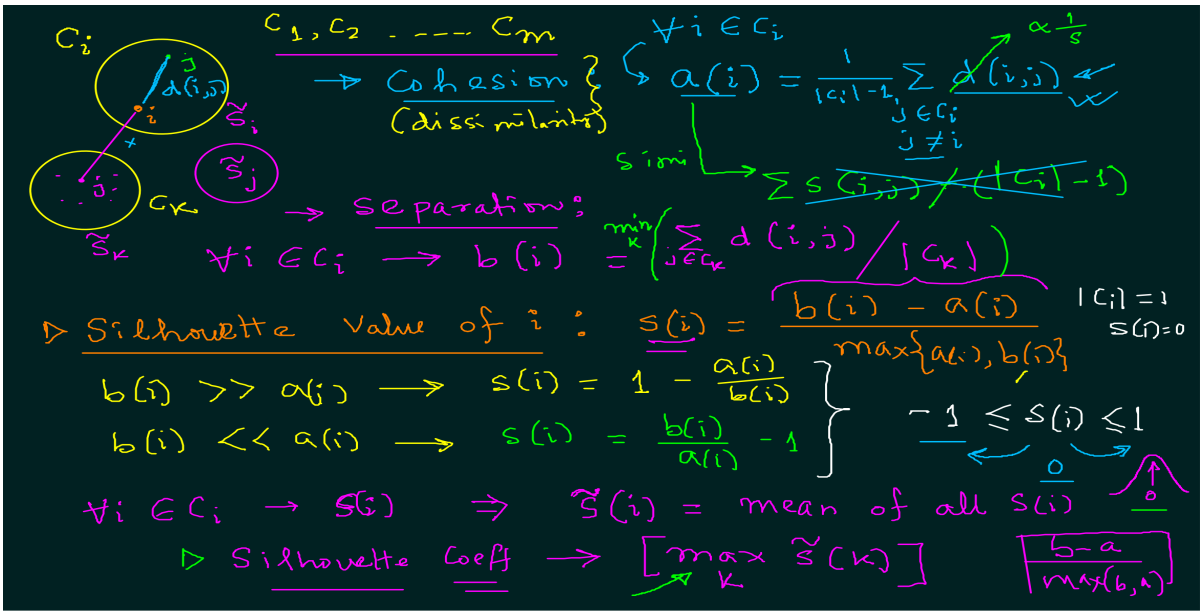

Figure 2: Silhouette Coefficients

We then define the Separation, of point i to some cluster C_k as the mean of the distance from i to all points in C_k (where $C_k \neq C_i$).
The Separation for each data point $i \in C_i$ can be expressed as:

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \tag{2}$$

We now define a **Silhouette** (value) of one data point i

$$s(i) = \frac{b(i) - a(i)}{\max\left(a(i), b(i)\right)} \tag{3}$$

The above equation hold true for $|C_i| > 1$
For $|C_i| = 1$, $s(i) = 0$

The above equation can also be expressed in 3 ways:

1. When $a(i) == b(i)$

$$s(i) = 0 \tag{4}$$

2. When $a(i) \gg b(i)$

$$s(i) = \frac{b(i)}{a(i)} - 1 \tag{5}$$

3. When $a(i) \ll b(i)$

$$s(i) = 1 - \frac{a(i)}{b(i)} \tag{6}$$

From the above definition it is clear that -1 $\leq s(i) \leq$ 1.

Some Observations from the above equations are:

1. The Silhouette score is bounded between -1 for incorrect clustering and +1 for highly dense clustering. Scores around zero indicate overlapping clusters and is the worst case scenario, as it is difficult to solve or proceed.

2. The Silhouette score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster.

The term silhouette coefficient for the maximum value of the mean $s(i)$ over all data of the entire dataset.

$$SC = \max_{k} \tilde{s}\,(k) \tag{7}$$

Where $\tilde{s}\,(k)$ represents the mean $s(i)$ over all data of the entire dataset for a specific number of clusters k.

To summarise the above, The Silhouette Coefficient tells us how well-assigned each individual point is. If $s(i)$ is close to 0, it is right at the inflection point between two clusters. If it is closer to -1, then we would have been better off assigning it to the other cluster. If $s(i)$ is close to 1, then the point is well-assigned and can be interpreted as belonging to an 'appropriate' cluster.