

# Unsupervised Learning

## MInDS @ Mines

In this lecture, we discuss some clustering methods and how they differ. We will cover hierarchical clustering, specifically agglomerative clustering, and nonnegative matrix factorization as an approach to clustering. We will also go over manifold vs Euclidean space and cover an example of clustering in the manifold space, namely spectral clustering.

We discussed clustering as the grouping of data into similar portions that are different from other portions where each portion is called a cluster. We now examine some methods for determining the clusters in our data.

### Hierarchical Clustering

Hierarchical clustering is an approach to clustering that creates a hierarchy of data samples to determine the clusters that each sample belongs to. There are two main types of hierarchical clustering; agglomerative, and divisive. Agglomerative clustering is a set of hierarchical clustering methods that initialize each sample as their own cluster then agglomerate them into larger groups until they form the desired number of clusters. Divisive clustering is a set of hierarchical clustering methods that initialize all samples as one cluster then proceed to divide the cluster into smaller clusters.

#### Agglomerative Clustering

In a clustering problem with  $n$  samples of data and a desired clustering of  $k$  clusters, we begin by defining  $n$  clusters, where every point is a cluster.

In every iteration, we then merge the two closest clusters together until we reach just 1 cluster.<sup>1</sup> We can then construct a dendrogram and start at the root then move down to the child clusters until we reach our desired  $k$  clusters.

<sup>1</sup> This approach allows us to calculate the clusters for all possible values of  $k$  at once. We can also stop the dendrogram at  $k$  and determine the clusters at the  $k$  level.

Now there are two hyperparameters to this method, mainly related to determining the two closest clusters. First, we select a distance metric between any two points, this can be the Euclidean distance ( $\ell_2$ -norm), Manhattan distance ( $\ell_1$ -norm) or any other distance metric. The second hyperparameter is how we calculate the distance from one cluster to the next. This is called the *linkage criteria*. We will merge the two clusters that have the minimal metric for the linkage criteria.

Three straightforward linkage criteria are the maximum distance, minimum distance, and average distance. The maximum distance, also called *complete linkage*, is the farthest distance between any two points in the clusters. The minimum distance, also called *single linkage*, is the closest distance between any two points in the clusters. The average distance, also called the Unweighted Pair Group Method with Arithmetic Mean (UPGMA), is the av-

erage of the distances between every pair of points from the two clusters. Another linkage criterion, *Ward's method*, utilizes the variance change in each cluster. Ward's method's criterion calculates the increase in variance of merging every two clusters. The method selects the cluster with the minimal criterion increase.

---

**Algorithm 1: Agglomerative Clustering Solution Algorithm**


---

**Input:** Features from data

**Output:**  $k$  cluster labeling for each point

- 1 Initialize  $n$  clusters where every point is a cluster
  - 2 **while** cluster count reached greater than  $k$  **do**
  - 3     Determine the two clusters closest to each other based on distance metric and linkage criteria
  - 4     Merge the two clusters into one
  - 5 **end**
- 

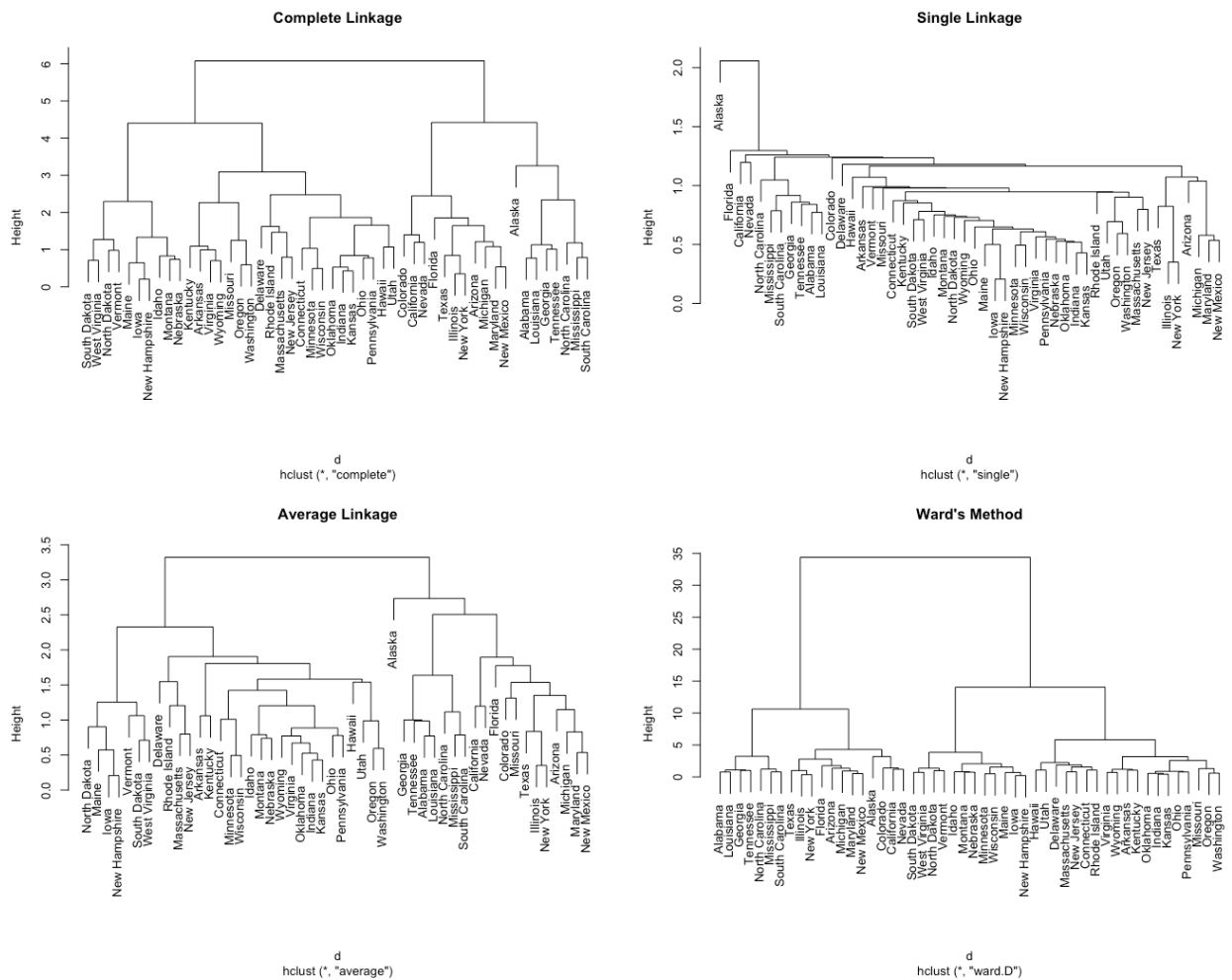


Figure 1: An overview of dendrograms created with different linkage criteria. Source

### Nonnegative Matrix Factorization

A matrix  $\mathbf{X}$  can be deconstructed and represented as the product of two matrixes,  $\mathbf{F}, \mathbf{G}, \mathbf{X} = \mathbf{FG}$ . If  $\mathbf{X}$  is a  $d \times n$  matrix, we can produce a  $d \times k$ ,  $\mathbf{F}$  matrix, and  $k \times n$ ,  $\mathbf{G}$  matrix.

From a clustering perspective,  $\mathbf{F}$  represents the centroids of the  $k$  clusters and  $\mathbf{G}$  represents the clustering membership of the  $n$  samples. This creates a direct relationship between matrix factorization and clustering. However, in order for this to be effective, both  $\mathbf{F}, \mathbf{G}$  must be nonnegative resulting in the name of this approach as Nonnegative Matrix Factorization or NMF.

The objective for NMF is,

$$\begin{aligned} \min_{\mathbf{F}, \mathbf{G}} \|\mathbf{X} - \mathbf{FG}\|_F^2 \\ \text{s.t. } \mathbf{F} \geq 0, \mathbf{G} \geq 0. \end{aligned} \quad (1)$$

In other words, the goal is to find the values for  $\mathbf{F}$  and  $\mathbf{G}$  whose product is as close to  $\mathbf{X}$  as possible while ensuring that they are both nonnegative.

### Euclidean vs Manifold Space

So far, whenever we've considered data, we consider its position in Euclidean space, by that we mean its  $x$  and  $y$  coordinates for example. This approach to data representation can be extremely effective but it is not the only approach and others may be effective in different cases. One such approach to data representation is the *manifold space*. The manifold space looks at the relationship between points instead of the position of points in a constant Euclidean space. Instead of points receiving an  $x$  and  $y$  coordinate, they receive a distance metric to all other points. We can represent our data as a graph and investigate its topology.

A good example of data that is better represented using its topology instead of position in space is a set of concentric circles. If each circle in the set is considered a class, using the relationship between the points instead of their absolute position will allow algorithms to perform significantly better in classifying them.

### Spectral Clustering

Spectral clustering is an example of a very powerful clustering method in the manifold space. It is in fact simply applying the KMeans algorithm to data in the transformed manifold space. First, we must investigate how to transform our data into a representation extracted from the manifold.

The way we represent our data as a graph can be extracted from the points' position in absolute space. We can represent a graph using an  $n \times n$  affinity matrix,  $\mathbf{A}$ , where  $a_{ij}$  is the distance between point  $i$  and point  $j$ .<sup>2</sup> In addition to the affinity matrix,  $\mathbf{A}$ , we define the degree matrix,  $\mathbf{D}$ , which is a

The reason NMF has this ability to produce results similar to KMeans clustering is described in Section 3 of "Orthogonal Nonnegative Matrix Tri-Factorizations for Clustering"

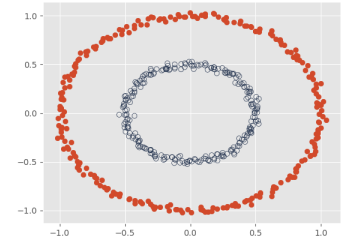


Figure 2: An example of concentric circles data that can be better represented using its topology than its absolute position in space.

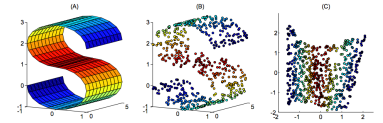


Figure 3: Example transformation using the manifold representation of data.

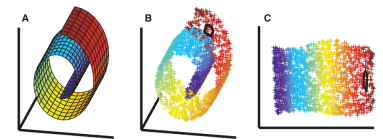


Figure 4: Example transformation using the manifold representation of data.

<sup>2</sup> Here again, we can decide on the distance metric we wish to use whether it be Euclidean, Manhattan or others.

diagonal matrix where  $d_{ii} = \sum_{j=1}^n a_{ij}$ . We also define the Laplacian matrix  $\mathbf{L} = \mathbf{D} - \mathbf{A}$ .

From the Laplacian matrix, we can learn a new representation, called an *embedding*<sup>3</sup>, of our points from the first  $r$  eigenvectors of the matrix. We define  $\mathbf{U}$  as the  $n \times r$  matrix where column  $\mathbf{u}_i$  is the  $i^{th}$  eigenvector of  $\mathbf{L}$ . The new representation of each data sample is the corresponding row in  $\mathbf{U}$  which is called the *Laplacian embedding*. Spectral clustering is applying the KMeans algorithm on the points using this new representation.

<sup>3</sup> We will further discuss embeddings in the feature learning portion of this course.