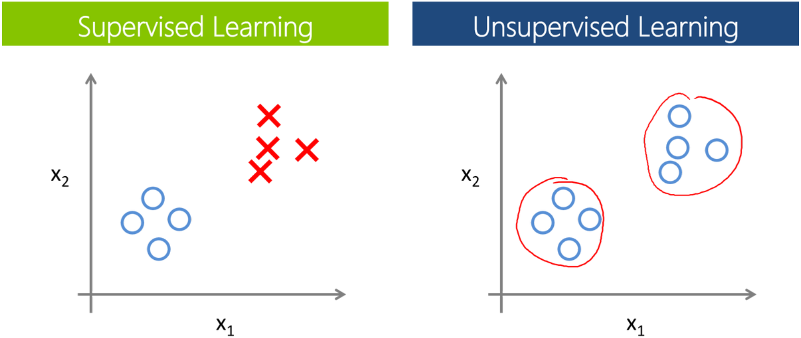
IRON HACK

Week 19 -Unsupervised Learning

Unsupervised Learning



Unsupervised Learning

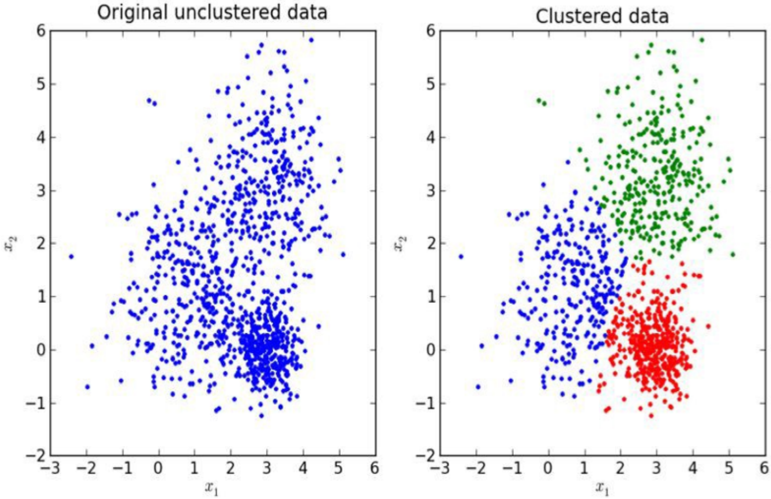
- Dataset does not have any pre-defined labels.
- Using algorithm to form groups and make sense of the existing data points.

Examples of Problems suitable for Unsupervised Learning

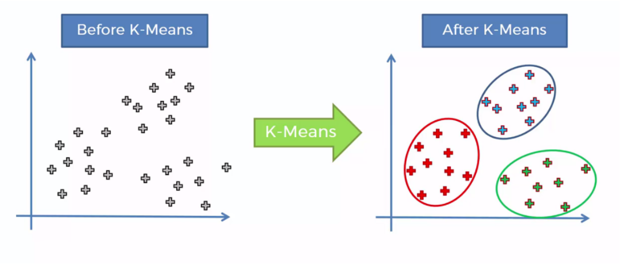
- Recommendation systems: Grouping existing users by their product or listening history, and recommend them new products either that are similar to their past products or liked songs, or by the tastes of other users who have similar habits and tastes.

- Response to medical treatments: Different patients may respond differently to the same medication or medical treatment. Use an algorithm which groups every patient and treatment (represented in a single vector, with attributes like age, gender, weight, height, dosage, intakes per day, start date, end date, known illnesses, symptoms, side effects, other medications, etc) so that cases with similar responses are grouped in the same cluster.

Supervised Learning



K-Means Clustering



In K-Means, we choose how many clusters we would like to create based on domain knowledge (typically we call that number k).

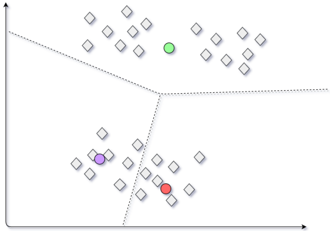
K-Means Clustering

- For each of the clusters we expect to see, we randomly assign a starting point, or centroid. We then compute the distance from this randomly-placed centroid to the data points that are nearest to it (ie. closest to that particular centroid than to any other).
- After calculating these distances, we move the centroid to be closer to those datapoints that we have assigned to its cluster
- We keep repeating this measure-and-move process iteratively until the centroid does not need to be moved to be closer to the other points in its cluster, and the datapoints do not change clusters.

`https://youtu.be/4b5d3muPQmA?t=20`

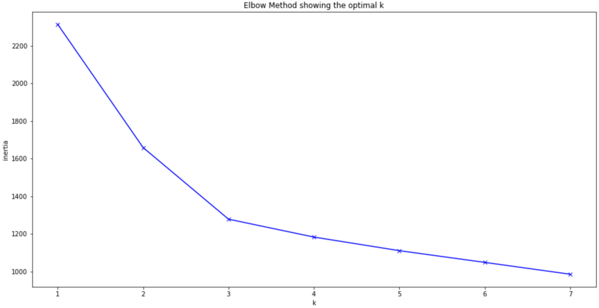
Disadvantage of K-Means Clustering

- Number of clusters, and the first location of the centroids are chosen almost randomly. Too many clusters could result in clusters that have too much in common with each other to be meaningfully distinguished, whereas too few would result in some different groups being put together in to one cluster.



How to pick the number of clusters?

'Inertia' is the mean squared distance between each instance and its closest centroid.



This is called an elbow curve.

Silhouette Score

- Silhouette Score also measures how similar is an observation is to its own cluster compared to other clusters.
- For the ith observation, the Silhouette Score is:

$$\frac{(b - a)}{max(a, b)}$$

where:

  - a is the mean intra-cluster distance (the average distance between i and every other observation in its cluster)
  - b is the mean nearest-cluster distance (the average distance between i and the observations of the nearest cluster that i is not part of)
  - The silhouette score for the whole model is the average of all the silhouette scores of each instance.

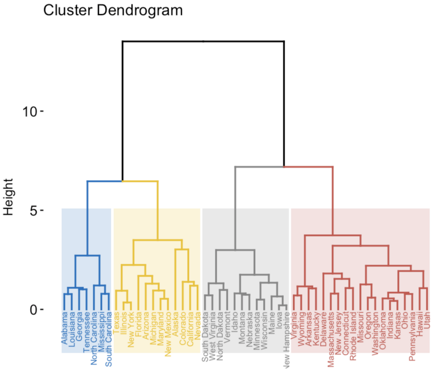
Silhouette Score

- Because we divide the subtraction of (b-a) by the max of the two distances (which will always be b unless the observation has been wrongly assigned to a cluster it should not belong), we obtain a "normalized score", that ranges from -1 to 1, and that makes it easier to interpret.

- A score of 1 means that the cluster is dense and well-separated from the other clusters. If the value is close to 0 then clusters are overlapping with samples being very close to the boundary of their neighboring clusters. If the score gets negative, this indicates that samples might have been assigned to the wrong clusters.

- Look for upward spikes!

Hierarchical Clustering



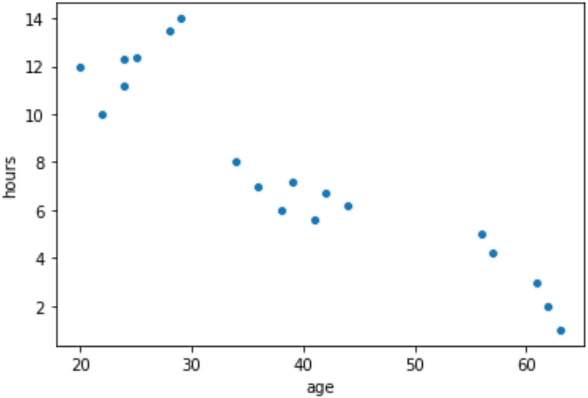
Hierarchical clustering is a clustering technique where we create a 'hierarchy' of clusters.

Hierarchical Clustering

Example: App usage data, we have age and hours of app engagement for a number of users

1. Find the two observations, which are most similar ("closest") to each other, and make them part of a group.
2. Find the two observations, which next closest to each other, and make them part of another group.
3. Keep iterating until all points are assigned.

Hierarchical Clustering: App Usage Data



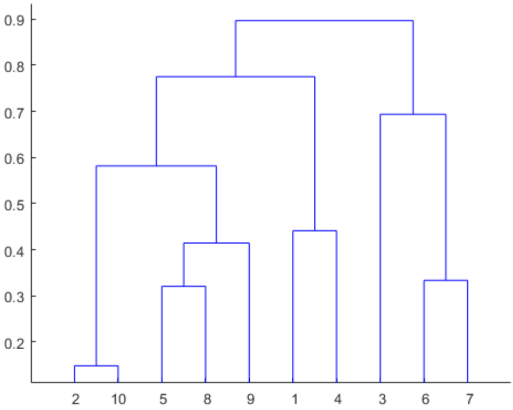
Hierarchical clustering is a clustering technique where we create a 'hierarchy' of clusters.

Hierarchical Clustering

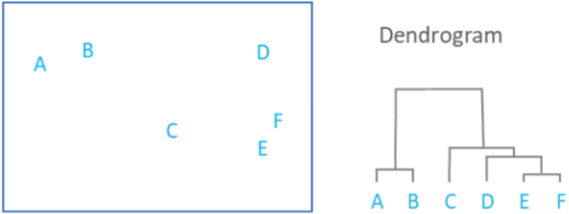
- We determine the cluster of a point by completing a process of grouping or un-grouping all of the observations in the dataset, sequentially.
- This can be done either forward or backward, and groupings are based on the euclidean distance (straight line) between the points.
- A dendrogram can be used to visualize the relationships between observations.

`https://www.youtube.com/watch?v=QXOkPvFM6NU`

Hierarchical Clustering: Dendrogram



Hierarchical Clustering: Dendrogram



Types of Hierarchical Clustering

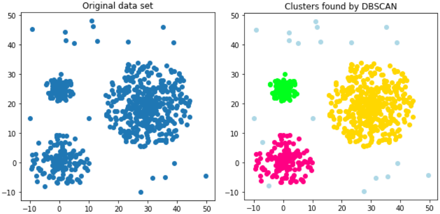
1. **Agglomerative**: This is a bottom up approach. We start off with a cluster for each observation and then combine similar clusters until we are left with only one large cluster.
2. **Divisive**: This is a top down approach. We start with one large cluster and keep dividing until we are left with observations.

Advantages / Disadvantages of Hierarchical Clustering

- Algorithm treats the entire dataset sequentially and requires a lot of memory / can be slow to run.

- Advantage over K-Means: number of clusters does not have to be determined, and the optimal number can be discovered by examining the changes in distance between the clusters as they appear on the dendrogram.

DBSCAN



DBSCAN stands for ’Density Based Spatial Clustering of Applications with Noise’. This clustering algorithm starts by picking a random point from the dataset, and defining what is called a ’neighborhood’.

DBSCAN

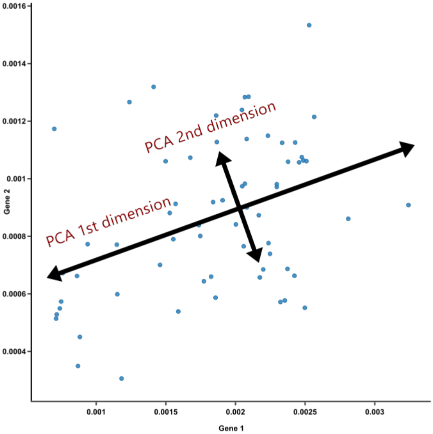
- Algorithm counts how many points are inside the 'neighborhood' of the chosen point and if there are more than a minimum number of points (chosen as a parameter of the the algorithm), these points are determined to be of the same cluster as the chosen point.
- If there are not enough points in the neighborhood to reach the minimum, the chosen point is marked as 'noise', and as not belonging to a cluster.
- Next: scanning the neighborhoods of those points that have been added to the cluster in order to add even more points to the existing cluster, or mark points as noise.

Advantages and Disadvantages of DBSCAN

- DBSCAN results can vary significantly depending on the size of the neighborhood chosen and the number of points needed to define a cluster.

- DBSCAN works best in datasets that have a relatively uniform density, as the minimum number of points needed to define a cluster might be different in areas of that dataset that are very dense or very sparse.

Dimensionality Reduction: PCA



PCA is a method that takes a dataset with k features that are assumed to be somewhat correlated and produces k new features, called Principal Components.

Dimensionality Reduction: PCA

- In PCA, features are ordered in such a way that the first few Principal Components explain most of the variability in the dataset.
- This particularity means that the last Principal Components can be completely discarded since they have almost no explanatory power - hence reducing the dimensionality of the dataset.
- Downside: loses interpretability of features.