**Clustering:**

**Overview:**

* If labels are available for clustering, they are used to evaluation instead.
* Partitional clustering: assign one object to one cluster (non-overlapping clusters)
* Hierarchical clustering (agglomerative, divisive): divide data into a set of nested clusters organized as hierarchy
* Well separated, center-based, contiguous, density based, property or conceptual, objective function based
* Density based: Used when the clusters are irregular or intertwined, and when noise and outliers are present.
* Property or conceptual: two circles
* Data characteristics that affect proximity and/or density - Dimensionality (Sparseness), Attribute type, Special
* relationships in the data (For example, autocorrelation), Distribution of the data
* K-means converges, O(n\*k\*l\*d), can have empty clusters
* Increasing k decreases SSE for clustering
* K means struggles when data is of varying size, density and non-globular and contains outliers
* Number of ways to select initial points: k! / K\*\*K (Not in our favor)
* K-mean++ (produces consistent results in terms of SSE)
* Updating centers incrementally

**Handling empty clusters:**

* Choose the point that contributes most to SSE.
* Choose a point from the cluster with the highest SSE.
* If there are several empty clusters, repeat the methods above multiple times.
* Updating clusters incrementally.

**Pre-processing:**

* Normalize the data
* Eliminate outliers

**Post-processing:**

* Eliminate small clusters that may represent outliers
* Split ‘loose’ clusters, i.e., clusters with relatively high SSE
* Merge clusters that are ‘close’ and that have relatively low SSE
* Can use these steps during the clustering process

**Hierarchical Clustering:**

* Produces a set of nested clusters organized as a hierarchical tree
* Can be visualized as a dendrogram
* Agglomerative:
  + Start with the points as individual clusters
  + At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
* Divisive:
  + Start with one, all-inclusive cluster
  + At each step, split a cluster until each cluster contains an individual point (or there are k clusters).

**Strengths of Hierarchical Clustering:**

* You do not have to assume any particular number of clusters
* Any desired number of clusters can be obtained by ‘cutting’ the dendrogram at the proper level.
* The hierarchy may correspond to meaningful taxonomies:
  + Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction)

**K-means:**

* K random Initial centroids. Compute distance of all point from each centroid. Assign to nearest. Recompute centroid. Repeat till no change.
* Not good with non-globular, different sizes, densities.
* Can yield empty clusters.
* SSE.

**Bisecting K means:**

* A variant of K-means that can produce either a partitional or a hierarchical clustering.
* Every point belongs to a single cluster. Divide into two. Take one of cluster and divide into two. Repeat.
* **Min** (Single link): handles non-elliptical shapes only distance which is sensitive to noise, outliers.
* **Max**: (Complete linkage): Resilient to noise. Tends to break large globular clusters, biased towards globular clusters.
* **Group average**: Average between distances. Biased towards globular clusters.
* **Ward’s method**: For Ward’s Method, the proximity between two clusters is the increase in squared error that results when the two clusters are merged.
* Increase in SSE when two clusters are merged.

**Hierarchical:** (Time - (O(N\*\*2), Space - O(N\*\*3))

**DBSCAN:**

* Handles clusters of different size, resistant to noise. Does not work for varying densities and high dimensional data.
* Eps (radius), MinPts (Minimum Points).
* Points with MinPts in eps radius is a **core point**.
* Points part of a core point is a **border** point but can’t has MinPts points in its Eps.
* If it’s neither it’s **noise** point.
* Border point belongs to some cluster (core point) but does not have MinPts to be **core point.**
* Does not work well: Varying Densities. High-dimensional data.

**Cluster Validity:**

* SSE, Correlation.

**Anomaly Detection:**

* Model Based
  + Unsupervised: statistical, clusters, regression, geometric, graph
  + Supervised: rare classes
* Proximity Based
* Density Based
* Pattern Matching

**Dimension Reduction:**

**PCA:**

* Finds a projection of the high dimensional data along an axis with the highest variance. (new data points are uncorrelated)
* Centers the data by subtracting mean from all data points to center the data.
* Tends to identify strong patterns in data. Can be used as a pattern finding technique.
* Deals with curse of dimensionality, removes redundant/irrelevant features/noise, more visualizable.
* 2D representation -> First dimension - max variability as possible, second dim – remaining variability.
* Pick number of dimensions based on how much variance they capture.
* Goal: Covariance Matrix

**Others:**

* PCA is equivalent to SVD if mean removal is not done for PCA.
* The PCA components are orthogonal to each other.
* MDS preserve pairwise distance as measured by stress.
* MDS, PCA are not good at dimensionality reduction when there are nonlinearities / complications
* Only FastMap has linear time and space complexity
* PCA, SVD, LLE produce same answer on each run.
* ISOMAP is good for nonlinear (MDS, ISOMAP)

