Machine Learning

Unsupervised Learning

Unsupervised Learning

- How is Unsupervised learning different from supervised learning?
  - In Supervised learning, our aim was to predict a response variable Y
  - In unsupervised leaning, we are not interested in prediction. The model is not provided with the correct results during the training
  - The goal is to discover interesting things about data
  - It is more subjective than supervised learning, as there is no simple goal for the analysis, such as prediction of a response.

Unsupervised Learning

- Two techniques of unsupervised learning covered in training
  - Clustering, a broad class of methods for discovering unknown subgroups in data
  - Principal components analysis, produces a low-dimensional representation of a dataset

Machine Learning

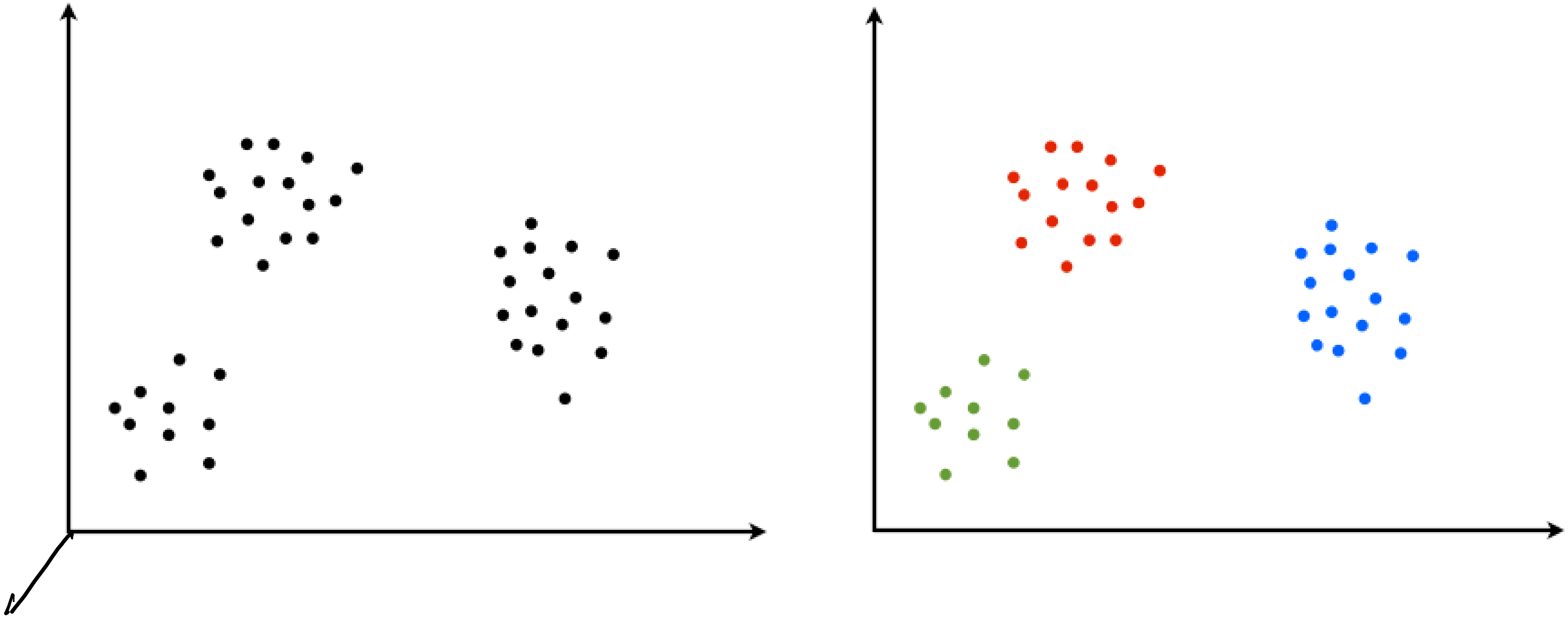
Clustering

Clustering

- What is Clustering?
  - A technique to organize unlabeled data into similar subgroups. It is a techniques of discovering subgroups, or clusters, in a dataset
  - Clustering partitions data into distinct subgroups so that the observations within each subgroup are similar to each other
  - A good cluster shall have **high**
    - **Intra-class similarity**: Similarity between objects in same cluster
    - **Inter-class dissimilarity**: dissimilarity between objects in different clusters
  - This is an unsupervised method to discover clusters in data

Clustering

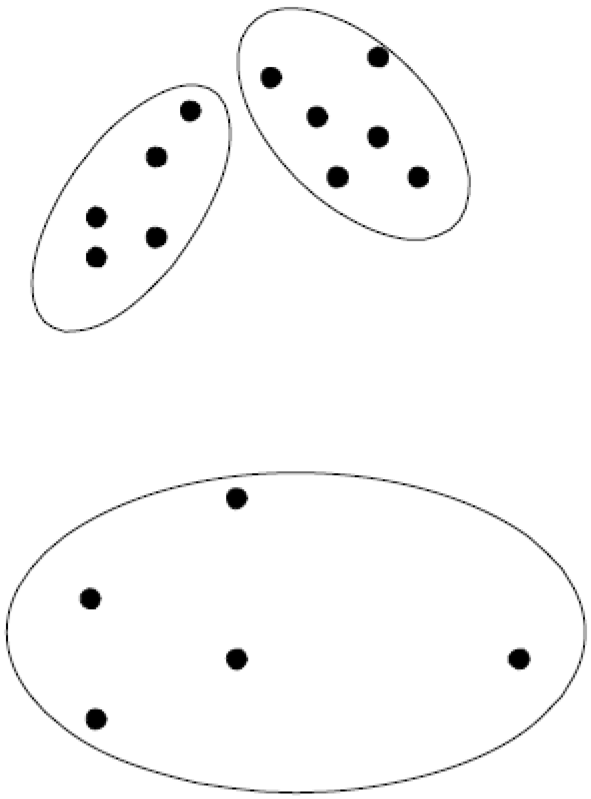
- Idea behind clustering – visual representation of natural clusters



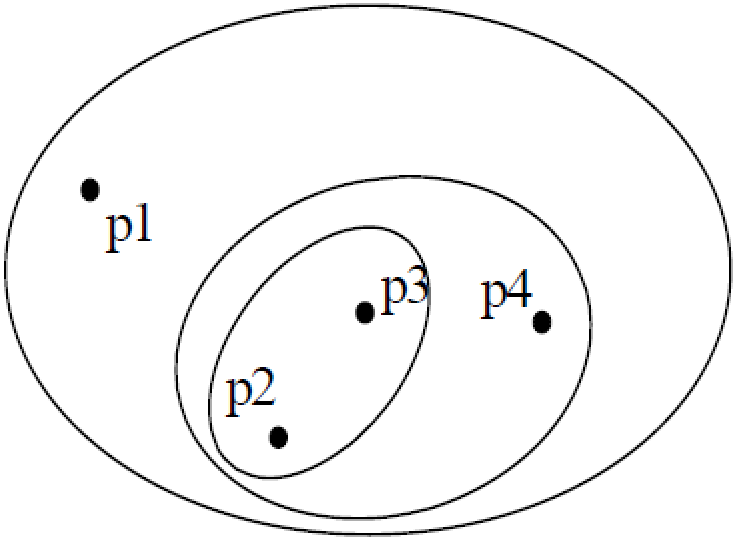
Types of Clustering

- Types of Clustering
  - Hierarchical Clustering
    - A set of nested clusters organized as a hierarchical tree. Example: Agglomerative clustering
  - Partitioning Clustering
    - A division data objects into non-overlapping clusters such that each data object is in exactly one subset. Example K-means clustering

Types of Clustering



Partition clustering



Hierarchical clustering

K Means Clustering

- K-means is a simple way to partition a data set into K distinct, non-overlapping clusters.

- Widely used in data mining

- Each cluster is associated with a centroid (center point of cluster). A centroid may not be a point from data

- Each point is assigned to the cluster with the closest centroid

- Number of clusters, K, must be specified

K Means Clustering

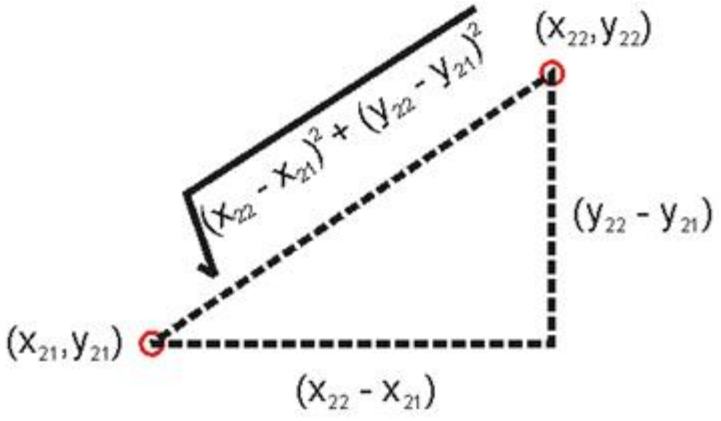
- It is an iterative clustering algorithm
  - Step 1: Pick K random points as cluster centers (C1, C2, . . . Ck)
  - Step 2: Form K clusters by assigning all data points to closest cluster center (each data point belongs to one centroid)
  - Step 3: Recompute centroid of each cluster
  - Step 4: Repeat Step 2 and 3 until no points assignments change take place or fixed number of iterations

Visual demo :

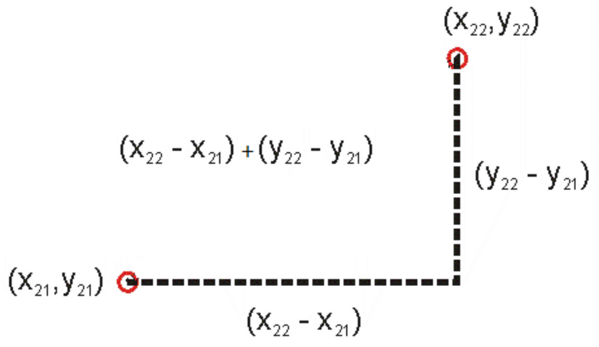
http://tech.nitoyon.com/en/blog/2013/11/07/k-means/

Distance calculation

- Euclidean Distance: Also called L2 norm. This is the most commonly used notion of distance. If there are two dimensions x and y, the distance between two point A and B is as shown



- Manhattan / Taxi Distance: Also called L1 norm. It is the sum of the differences in each dimension.



Scaling data

- Distance computed in Euclidian methods are highly influenced by the scale of each variable

- If scaling is not done then, variables with larger scale have much greater influence over the total distance and hence on clustering

- Common methods of scaling are
  - Z-score
  - Min-max scaler

- So dimensions are converted to standardized scale. Scaling makes all attributes equally important.

- However, Sometimes scaling may have adverse effect on the clusters

K Means Clustering – Key points

- Convergence: The algorithm will typically converge, but number of iterations could be large

- Results can vary based on random seed selection (initial selection of K random centroids). The algorithm is sensitive to the initial seed selection

    - Some seeds can result in poor clustering

    - Scikit-learn has implemented K-mean++ for initial centroid. It initializes centroids to be distant to one another which leads to better results

K Means Clustering – Key points

- Determining number of clusters K is a challenge
- A simple strategy is to compute a clustering for various values of K and choose the best one.
- But how to determine quality of clustering?
- Measure the quality of a clustering by distances of points cluster centers. The common metric used is Within Cluster Sum of Squares :

$$\sum_{i=1}^{K} \sum_{x \in C_i} |x - \mu_i|^2$$

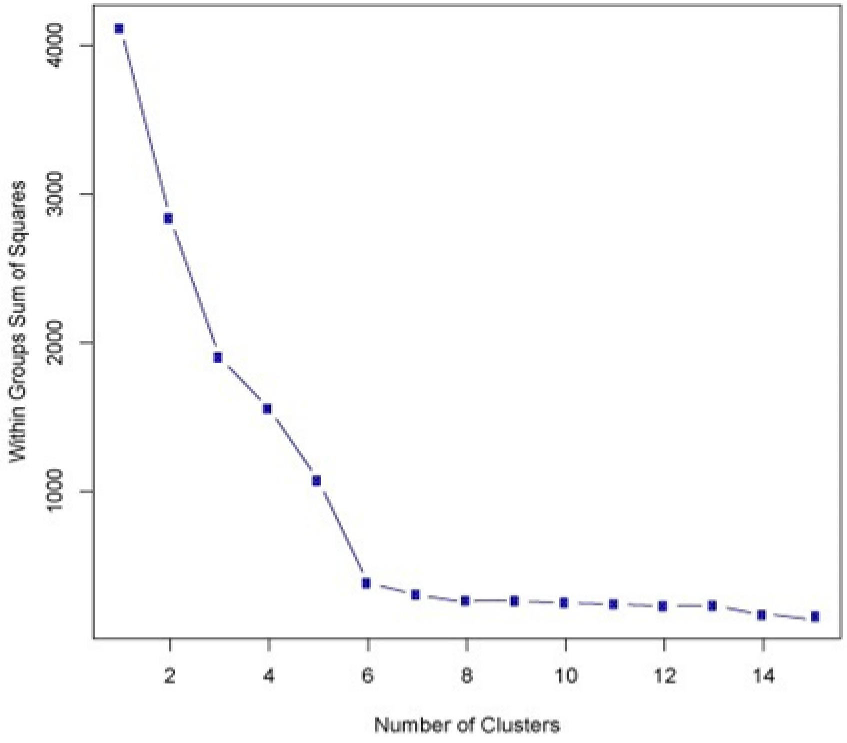
where C_i is ith cluster with μ_i as centroid

K Means Clustering – Key points

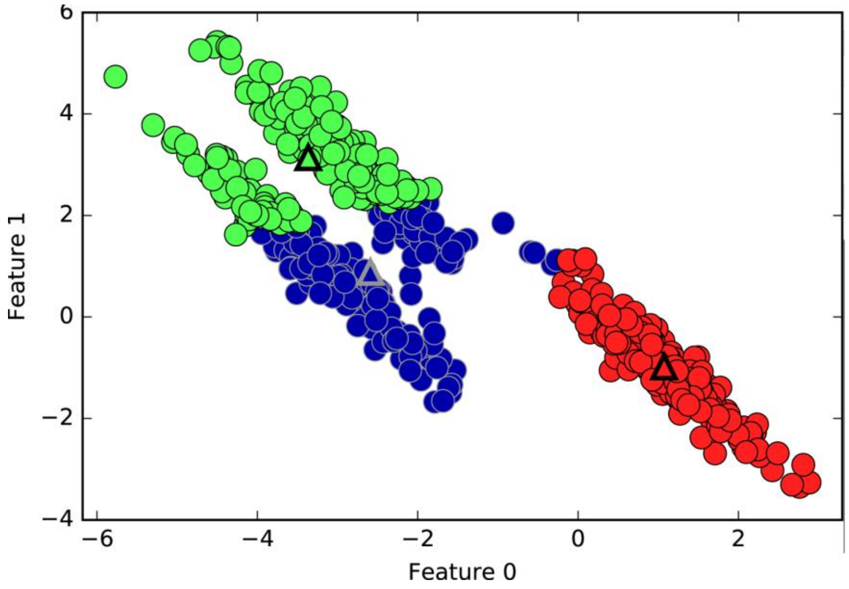
- As K increases, Within Cluster Sum of Squares will reduce. Hence combine Within Cluster Sum of Squares along with number of clusters to determine desired K value.

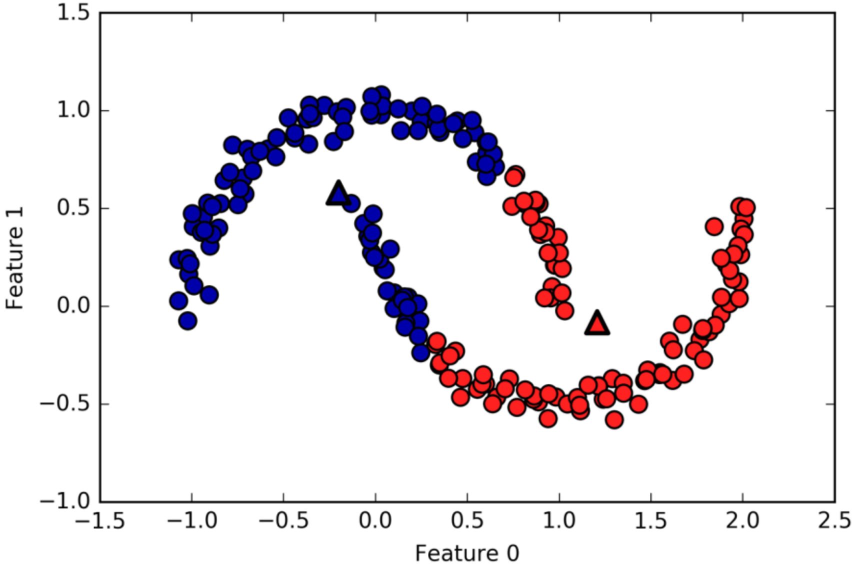
- Use elbow chart method to determine value of K where benefit doesn't increase by "much" by increasing the value of K.

- However, k-Means Clustering is not guaranteed to find the global minimum



Examples of poor clustering





K Means Clustering

- Advantages
    - Easy to understand

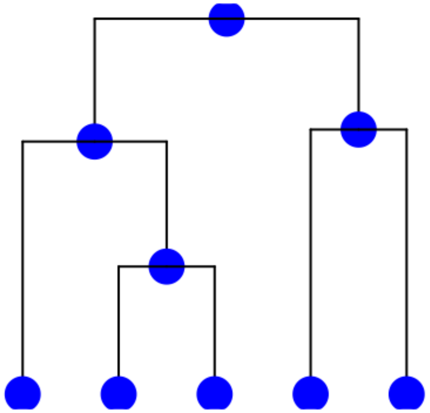
- Disadvantages
    - Computation intensive
    - The user needs to specify k
    - Sensitive to initial positions of centroids
    - Sensitive to outliers

K Means Clustering

- Hands-on exercise

Hierarchical - Agglomerative Clustering

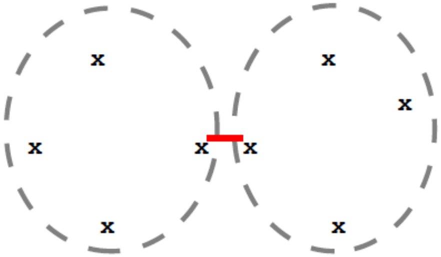
- Produces a tree hierarchy of clusters
- Starts with each data point as a separate cluster
- Then it repeatedly joins the two clusters that are nearest until there is only one cluster (root cluster)
- The history of merging forms a hierarchy or dendrogram
- Clustering can be obtained by cutting the dendrogram at a given level
- All connected components form a cluster



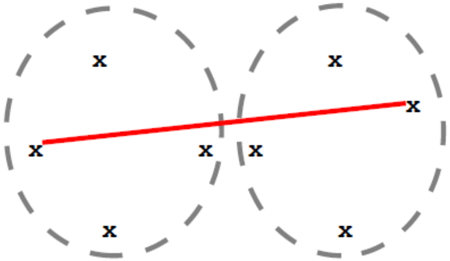
Distance between Clusters

- Agglomerative clustering technique calculates distance between clusters. Distance between clusters A and B can be calculated using:

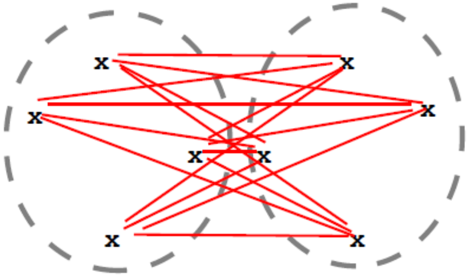
- Minimum distance(single linkage) – is the distance between pair of data points Ai and Bj that belong to clusters A and B respectively and are closest



- Maximum distance(complete linkage) – is the the pair of records Ai and Bj that belong to cluster A and B respectively and are farthest

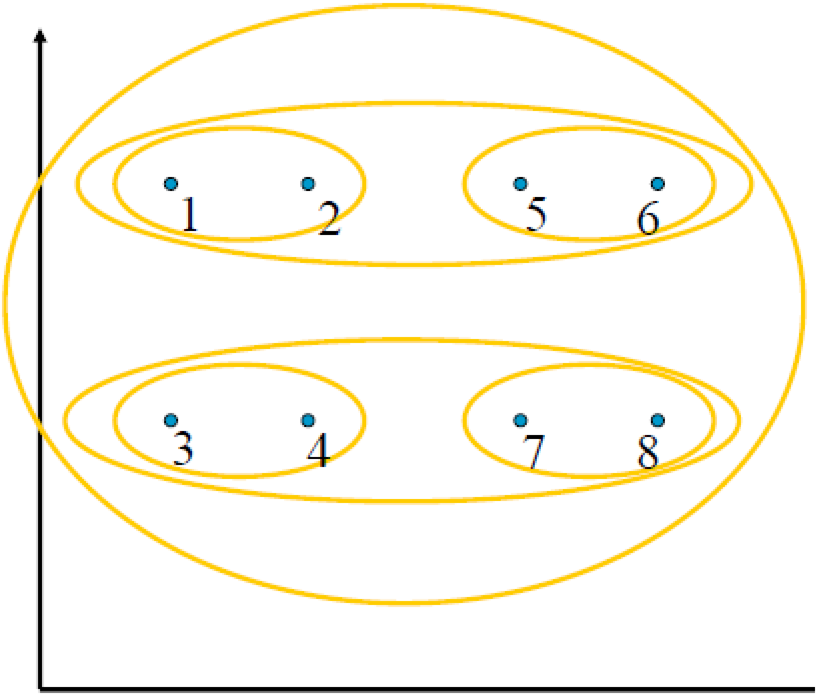


- Average distance (average linkage) - average distance of all possible distances between records in one cluster to records in other cluster

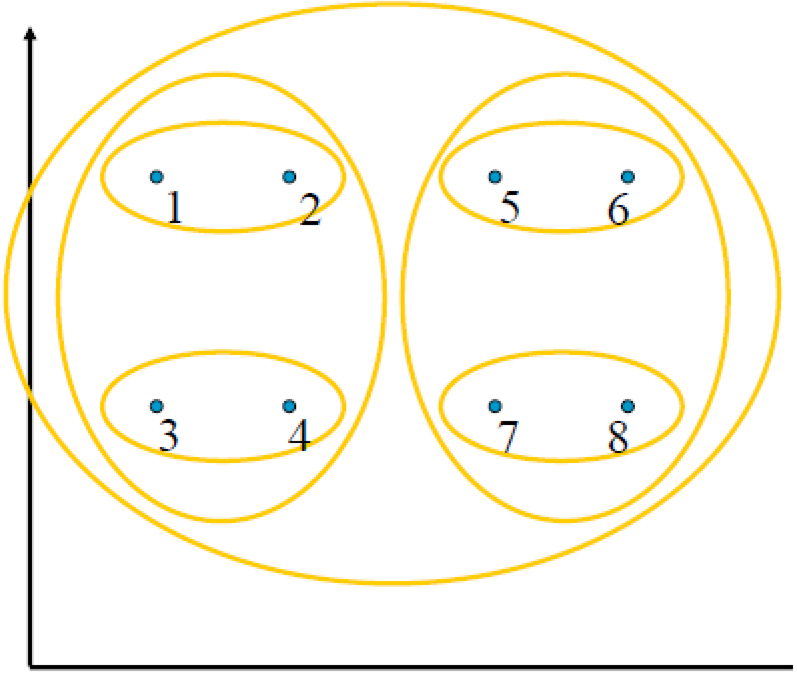


Clustering example

- Example of clustering using different distance calculation methods



Minimum distance
(single linkage)



Maximum distance
(complete linkage)

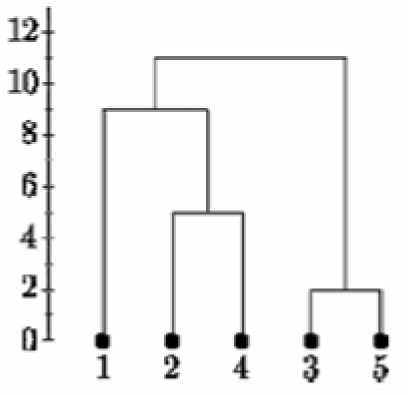
Agglomerative Clustering - example

- Example using Maximum distance method

	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	2	8	0

	35	1	2	4
35	0			
1	11	0		
2	10	9	0	
4	9	6	5	0

	35	24	1
35	0		
24	10	0	
1	11	9	0



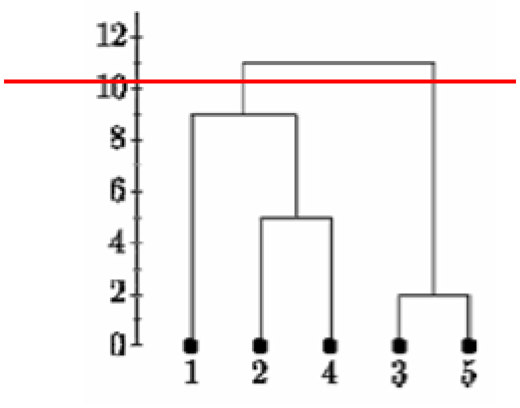
	35	124
35	0	
124	11	0

Determine Clusters

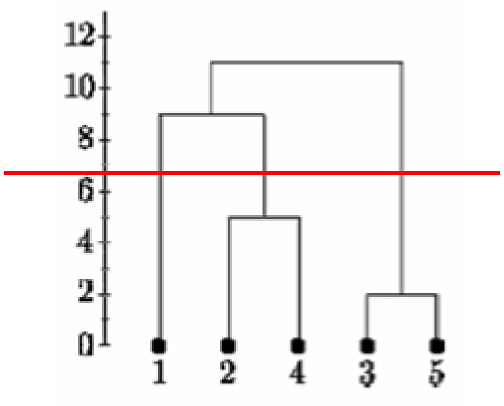
- Hierarchical clustering does not provide information about how many clusters there are

- Depending on where we cut the tree, we will get different number of cluster

- Determine the cut points depending on distance and domain knowlege



2 clusters



3 clusters

Hierarchical Clustering

- Hands-on exercise

Machine Learning

Principle Component Analysis

Principle Component Analysis

- A common use of principal component analysis (PCA) in machine learning is to
  - Reduce the dimensionality of a data set consisting of a large number of interrelated variables
  - Yet retain as much as possible of the variation present in the data set.

- This is achieved by
  - Transforming variables into a new set of variables, called the principal components (PCs). PCs are uncorrelated, and are ordered so that the first few retain most of the variation present in all of the original variables

Objective of PCA

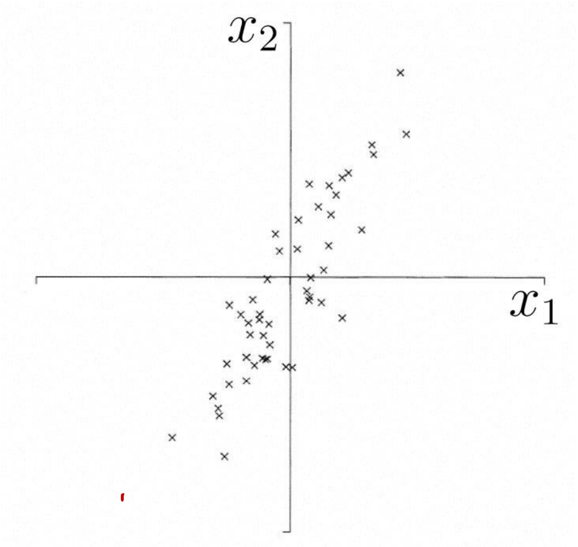
- PCA is used to reduce dimension space from a larger number of factors to a smaller number of factors

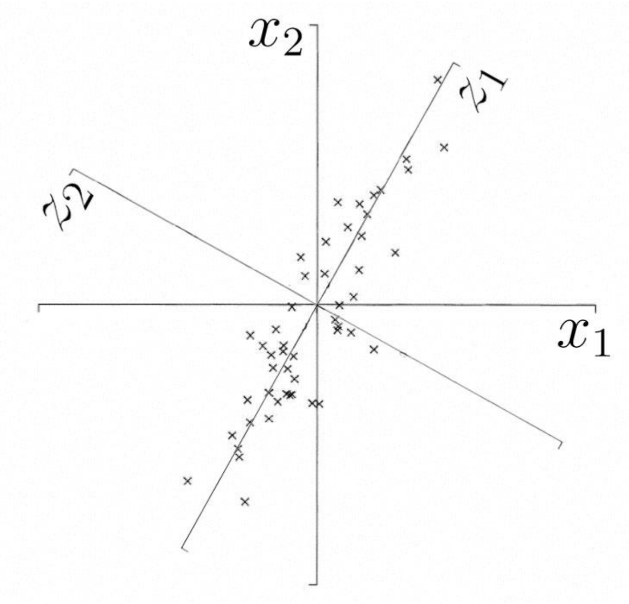
- It is a dimensionality reduction method. The goal is dimension reduction and there is no guarantee that the new dimensions are interpretable

- To select a subset of variables from a larger set, based on which original variables have the highest correlations with the principal component.

Principle Component Analysis

- Following example shows a sample of n observations in the 2-D space X = (x_1, x_2)
- Z1 is the direction along with maximum variance of projected values along the line





PCA Steps

- **Standardization**

  - The aim of this step is to standardize the variables so that each one of them contributes equally to the analysis

  - This can be done by calculating Z score for each value of each variable

$$z = \frac{value - mean}{standard\ deviation}$$

PCA Steps

- **Compute Covariance Matrix**
  - If there are n variables, then covariance matrix is ‘n x n’ symmetric matrix
  - Example of 3 x 3 covariance matrix

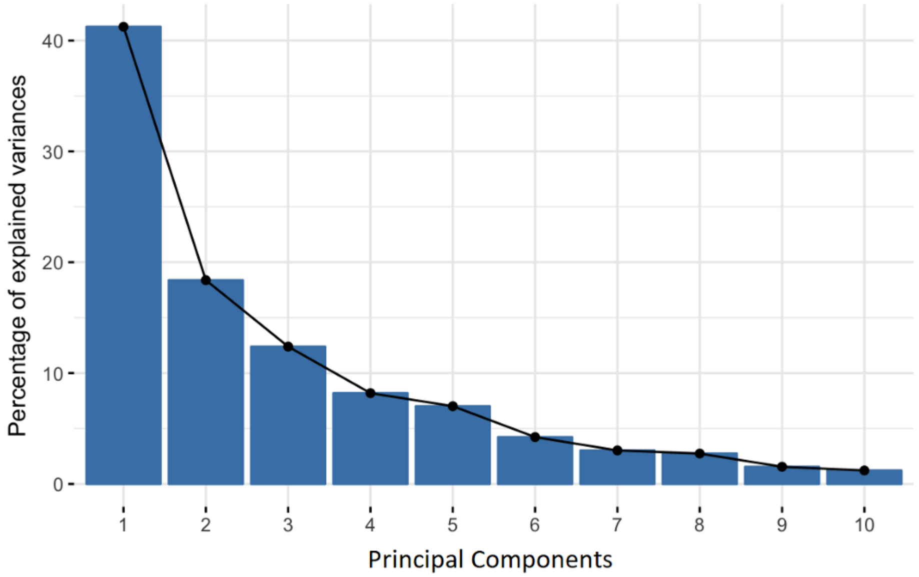
$$\begin{bmatrix} Cov(x,x) & Cov(x,y) & Cov(x,z) \\ Cov(y,x) & Cov(y,y) & Cov(y,z) \\ Cov(z,x) & Cov(z,y) & Cov(z,z) \end{bmatrix}$$

PCA Steps

- **Compute the eigenvectors and eigenvalues to identify the principal components**
  - Principal components are constructed in such a manner that the first principal component accounts for the largest possible variance, the second principal component is uncorrelated with (i.e., perpendicular to) the first principal component and that it accounts for the next highest variance
  - Continues until a total of n principal components have been calculated, equal to the original number of variables
  - Eigenvalues are the coefficients attached to eigenvectors, which give the amount of variance carried in each Principal Component
  - By sorting the eigenvalues, highest to lowest, we get the principal components in order of significance

PCA Steps

- Example: in a 10-dimensional data, we get 10 principal components sorted descending in the order of significance



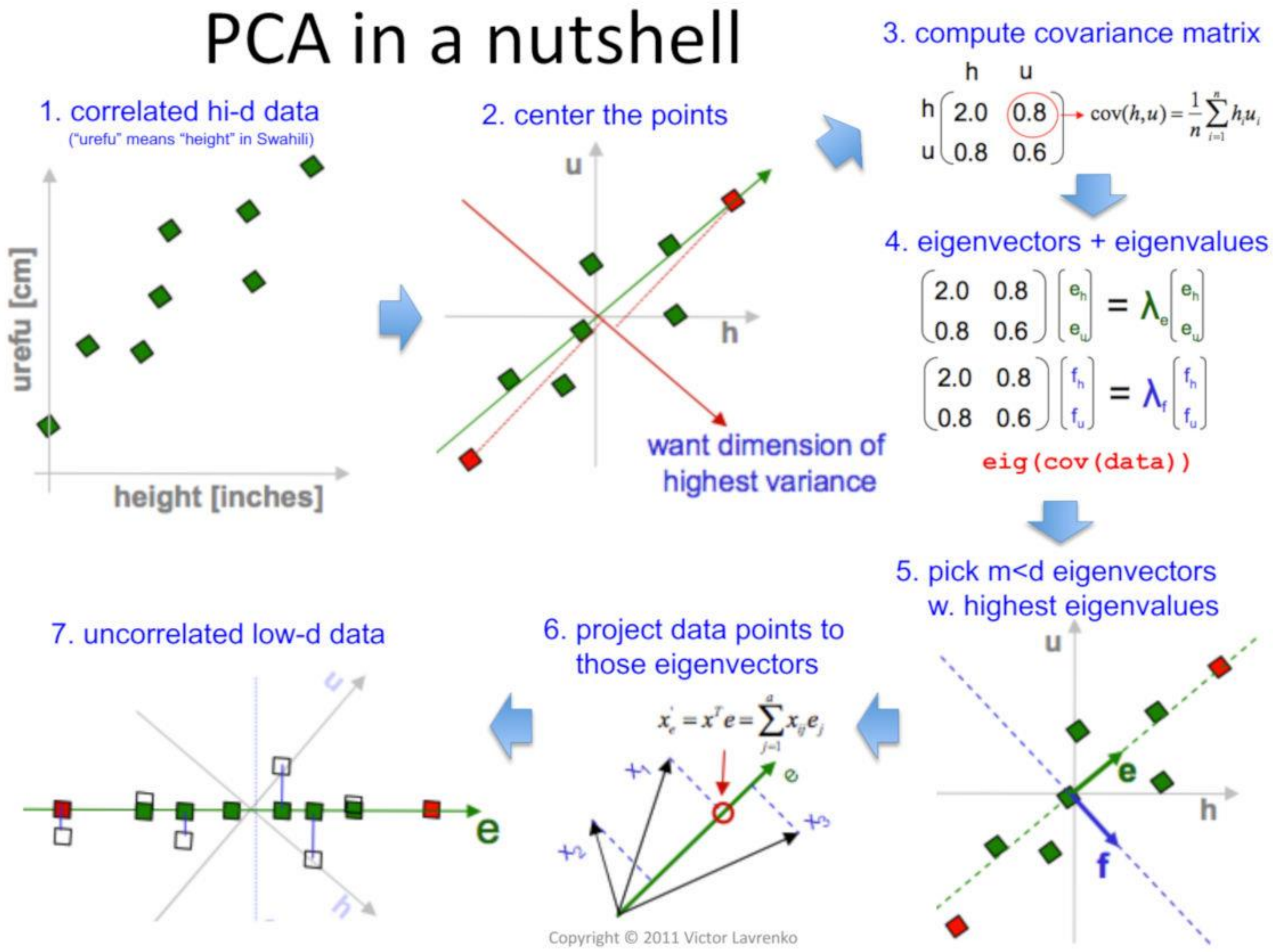
PCA Steps

- **Choose how many Principle components to be retained**
  - Based on % of information content in Principle components, choose whether to keep all these components or discard those of lesser significance (of low eigenvalues)
  - Form feature vector, which is a matrix, having columns as the eigenvectors of the components that we decide to keep

PCA Steps

- **Project the data along the retained principal component axes**
  - Reorient the data from the original axes to the ones represented by the principal components
  - This is achieved using matrix multiplication
  - Data with reduced dimension is ready for further analysis

PCS in a nutshell



PCA in a nutshell

1. correlated hi-d data

("urefu" means "height" in Swahili)

2. center the points

want dimension of highest variance

3. compute covariance matrix

$$\begin{array}{c|cc} & h & u \\ \hline h & 2.0 & 0.8 \\ u & 0.8 & 0.6 \end{array} \quad \text{cov}(h,u) = \frac{1}{n}\sum_{i=1}^{n} h_i u_i$$

4. eigenvectors + eigenvalues

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}\begin{pmatrix} e_h \\ e_u \end{pmatrix} = \lambda_e \begin{pmatrix} e_h \\ e_u \end{pmatrix}$$

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}\begin{pmatrix} f_h \\ f_u \end{pmatrix} = \lambda_f \begin{pmatrix} f_h \\ f_u \end{pmatrix}$$

`eig(cov(data))`

5. pick m<d eigenvectors w. highest eigenvalues

6. project data points to those eigenvectors

$$x_e' = x^T e = \sum_{j=1}^{d} x_{ij} e_j$$

7. uncorrelated low-d data

Copyright © 2011 Victor Lavrenko

Ref: https://devopedia.org/principal-component-analysis

Advantages and Disadvantages

- Advantages
  - Minimizes information loss even when fewer dimensions are considered for analysis
  - Although Gaussian distribution of data is assumed, PCA doesn't need this assumption.
  - Fewer dimensions means less computation and lower error rate. PCA reduces noise and makes algorithms work better.

- Disadvantages
  - PCA works well if the observed variables are linearly correlated.
  - Since each principal components is a linear combination of the original features, interpretability with regard to original features is lost