Unsupervised Learning

Unsupervised Learning

Clustering

1. Clustering is primarily an exploratory technique to discover hidden structures of the data, possibly as a prelude to more focused analysis or decision processes
    a. A way to decompose a data set into subsets with each subset representing a group with similar characteristics

    b. Group such that objects in the same group are more similar to each other in some sense than to objects of different groups

    c. The groups are known as clusters and each cluster gets distinct label called cluster id, the centroid of the cluster, and other details

2. Clustering is often used as a lead-in to classification. Once the clusters are identified, labels can be applied to each cluster to classify each group based on its characteristics

Applications of clustering

Some specific applications of k-means are image processing, medical, and customer segmentation

a. **Image processing** : used to cluster of pixels representing objects in each frame. The attributes of each pixel can include brightness, color, and location, the x and y coordinates in the frame. Successive frames are examined to identify any changes to the clusters. These newly identified clusters may indicate unauthorized access to a facility.

b. **Medical** : Patient attributes such as age, height, weight, systolic and diastolic blood pressures, cholesterol level, and other attributes can identify naturally occurring clusters under various health conditions

c. **Customer segmentation** : Cluster customers on basis of frequency of purchase, recency of purchase, value of purchase and look for common attributes among high value customers. Target all potential customers who have similar attributes

Clustering types

1. Two broad categories of clustering include hierarchical (agglomerative, divisive) and non hierarchical

2. Hierarchical clustering
    a) Agglomerative clustering algorithm uses a bottom-up approach and merges smaller clusters into larger ones
    b) Divisive clustering uses top-bottom approach to break a large cluster into smaller clusters

3. Non-hierarchical / partitional clusters are formed on assumption that the clusters are disjoint and there is no hierarchical relation between them. K Means is an example

Clustering – Distance calculations

1. Irrespective of the clustering algorithm, we need a way of defining and calculating distance between two data points in mathematical space (records in the database)

2. We also need a way to define and calculate distance between clusters

3. Distance between two data points is a measure of similarity between the points

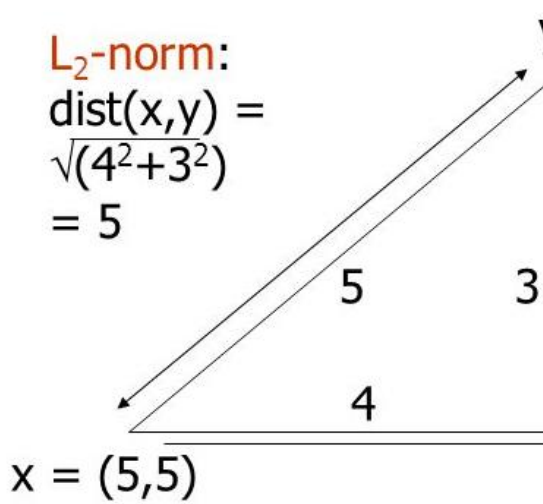
4. The lesser the distance, more similar the data points are

5. There are many ways of calculating distance between two points i.e. if d = f(x,y) then there are many ways in which f can be implemented

Euclidean Distance

1. Euclidean Distance
    a. L2 norm : d(x,y) = square root of the sum of the squares of the differences between x and y in each dimension. The most common notion of “distance”. If there are two dimensions x and y, the distance between two point A and B is –

    b. L1 norm : sum of the differences in each dimension. Manhattan distance = distance if you had to travel along coordinates only



Non Euclidean Distance

2. Non Euclidean Distance
    a. Jaccard distance for sets = 1 minus ratio of sizes of intersection and union.
    b. Cosine distance = angle between vectors from the origin to the points in question.
    c. Edit distance = number of inserts and deletes to change one string into another
    d. Mahalanobis distance

3. Normalizing the numerical measurements
    a. The measures computed in Euclidian methods are highly influenced by the scale of each variable
    b. Variables with larger scale have much greater influence over the total distance
    c. Hence all the measurements are converted to same scale (convert to z scores for e.g.)

Note: distance measurement is just a way of assessing similarity/ dissimilarity. The common parlance of distance will not help in understanding other methods of distance calculations

Distance measures

1. Distance measures and some key points:
    a. Choice of distance measures play a key role in cluster analysis.
    b. Knowledge of the distribution of data (gaussian or otherwise) will help
    c. Are the various attributes independent or influence eachother
    d. Are their outliers in the data on the various dimensions
    e. Though Euclidian distance is the most commonly used distance metric, it has three main features that should be kept in view
        a. It is highly scale dependent. Changing the units of one variable can have a huge influence on the results. Hence standardizing the dimensions is a good practice
        b. It completely ignores the relationship between measurements (Refer to Mahalanobis distance diagram)
        c. It is sensitive to outliers. If the data has outliers that cannot be handled or removed, use of Manhattan distance is preferred

K Means Clustering – Some considerations

1. K-Means (a.k.a Lloyd's algorithm) clusters data by separating data points into groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squared errors

2. It requires the number of clusters to be specified, hence the term "K" in its name

3. It divides the samples into K disjoint clusters Ci, each described by the mean of the samples in the cluster. The means are commonly called "centroids" (they are not the points from the data)

4. The K-Means algorithm chooses centroids that minimizes the inertia across all the clusters

K Means Clustering – Some considerations Pt 2

5. From a computational perspective, the k-means algorithm is indifferent to the units of measure for a given attribute (for example, meters or centimeters for a patient's height). However, the algorithm will identify different clusters depending on the choice of the units of measure.

6. Choosing different starting points can result in different clusters. The algorithm is sensitive to the initial starting condition

7. Given enough time, K-means will always converge, however this may be a local minimum. This is highly dependent on the initialization of the centroids

8. Scikit-learn has implemented K-mean++ initialization scheme, which initializes centroids to be distant to one another which provably leads to better results

K-means Clustering objective

The objective function of clustering is –

- Given: a set of n observations $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$, where each observation is a d-dimensional real vector
- Given: a number of clusters k
- Compute: a cluster assignment mapping $C(\boldsymbol{x}_i) \in \{1, \ldots, k\}$ that minimizes the **within cluster sum of squares (WCSS)**:

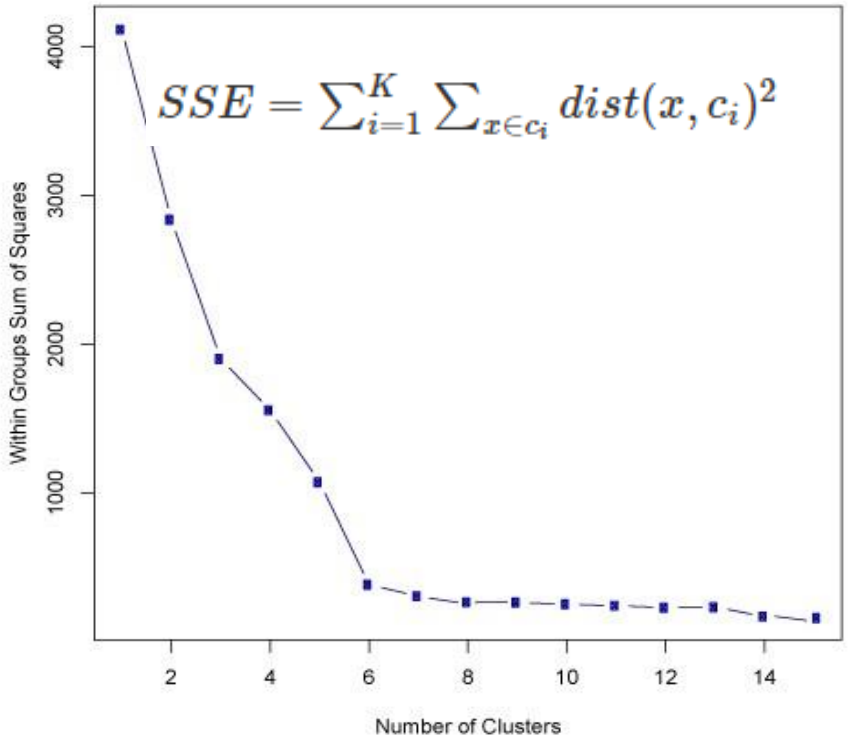
$$\sum_{i=1}^{n} \left\| \boldsymbol{x}_i - \boldsymbol{\mu}_{C(\boldsymbol{x}_i)} \right\|^2$$

where **centroid** $\boldsymbol{\mu}_{C(\boldsymbol{x}_i)}$ is the mean of the points in cluster $C(\boldsymbol{x}_i)$

Elbow method

Elbow Method

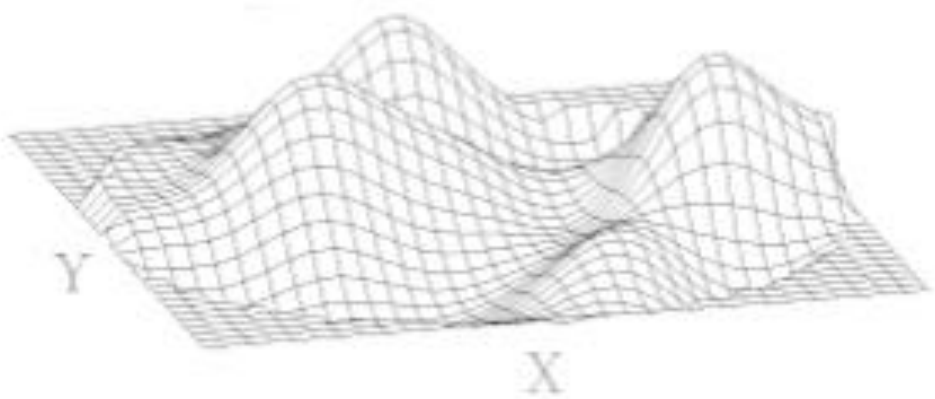
7. Without a priori knowledge, one can use elbow method that measures the homogeneity or heterogeneity within clusters as the number of clusters change (i.e. K is changed). One way to measure is use sum of square errors in each cluster



Visual Analysis for Clustering

Visual Analysis for Clustering

1. Visual analysis of the attributes selected for the clustering may given an idea of the range of values that K should be evaluated in



2. Identifying the attributes on which clusters are clearly demarcated and using them in incremental order to build the multi-dimensional clusters likely to give much better clusters than using all the attributes at one go

Dynamic Clustering

Dynamic Clustering

1. Clustering on correct attributes is the key to good clustering results.

2. We can also consider those attributes who's value changes with time. For e.g. age, income category, years of work experience etc.

3. We can use sequential k means clustering over time to track individual clusters (how they change in size, shape and position

4. We can also understand how individual data points move across clusters, form new clusters etc.

5. Analyzing the changes in the clusters over time using metrics such as

6. Cluster size, new entries and exits one can analyze the impact of strategies designed based on earlier clustering analysis

K-Means Clustering Strengths and Weakness

K-Means Clustering Strengths and Weakness

Strengths	Weakness
Use simple principles without the need for any complex statistical terms	Computationally intensive How to fix K?
Once clusters and their associated centroids are identified, it is easy to assign new objects (for example, new customers) to a cluster based on the object's distance from the closest centroid	The k-means algorithm is sensitive to the starting positions of the initial centroid. Thus, it is important to rerun the k-means analysis several times for a particular value of k to ensure the cluster results provide the overall minimum WSS
Because the method is unsupervised, using k-means helps to eliminate subjectivity from the analysis.	Susceptible to curse of dimensionality

Lab exercises

Lab exercises

Lab- 1 Analyze auto mpg data set using K means to explore the data in terms of the various attributes

Sol: KMeansClustering_Auto_Mpg.ipynb

Hierarchical (Agglomerative) Clustering

1. The agglomerative clustering starts with each cluster comprising exactly one data point in the feature space

2. It progressively agglomerates / combines the two nearest clusters until there is one grand cluster left in the feature space

3. For the closest cluster analysis, each of the inter cluster distance measurement techniques (single link, complete link, average link, centroid distance) can be implemented
    a. In single linkage method, the minimum distance between nearest points from the two clusters is used to consolidate clusters
    b. In complete linkage, distance between two farthest points form each cluster is considered
    c. Group average clustering is based on the average distance between clusters

4. Prior domain knowledge helps in deciding the inter cluster distance metric selection. If the clusters are likely to be in long chain or sausage like, minimum distance (single linkage) would be a good choice

5. Complete and average linkage are better choice if the clusters are likely to be spherical

Clustering – Measuring distance between clusters

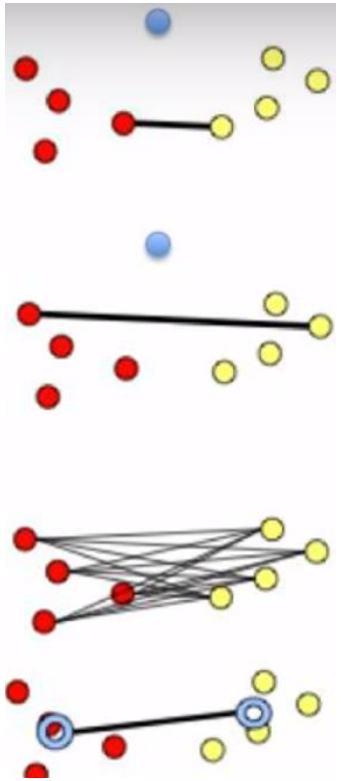
1. Ideally, a good clustering should result in compact clusters separated from one another by maximal distance. This calls for measuring the distance between cluster. The most widely used methods include :

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

   b. Maximum distance(complete linkage) – is the largest distance between the pair of records Ai and Bj that belong to cluster A and B respectively

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

   d. Centroid distance - the distance between centroids of the different clusters.



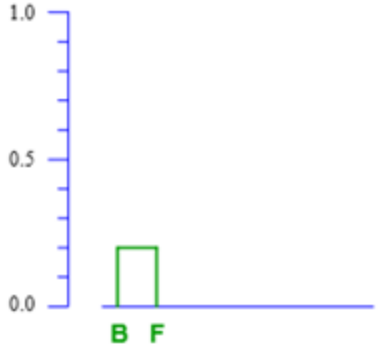
2. Distance between clusters is used in hierarchical clustering

Complete Linkage

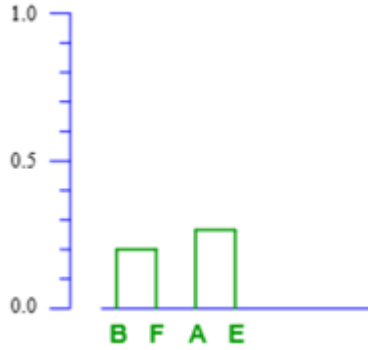
Complete Linkage

samples	A	B	C	D	E	F	G
A	0	0.5000	0.4286	1.0000	0.2500	0.6250	0.3750
B	0.5000	0	0.7143	0.8333	0.6667	0.2000	0.7778
C	0.4286	0.7143	0	1.0000	0.4286	0.6667	0.3333
D	1.0000	0.8333	1.0000	0	1.0000	0.8000	0.8571
E	0.2500	0.6667	0.4286	1.0000	0	0.7778	0.3750
F	0.6250	0.2000	0.6667	0.8000	0.7778	0	0.7500
G	0.3750	0.7778	0.3333	0.8571	0.3750	0.7500	0

samples	A	(B,F)	C	D	E	G
A	0	0.6250	0.4286	1.0000	0.2500	0.3750
(B,F)	0.6250	0	0.7143	0.8333	0.7778	0.7778
C	0.4286	0.7143	0	1.0000	0.4286	0.3333
D	1.0000	0.8333	1.0000	0	1.0000	0.8571
E	0.2500	0.7778	0.4286	1.0000	0	0.3750
G	0.3750	0.7778	0.3333	0.8571	0.3750	0

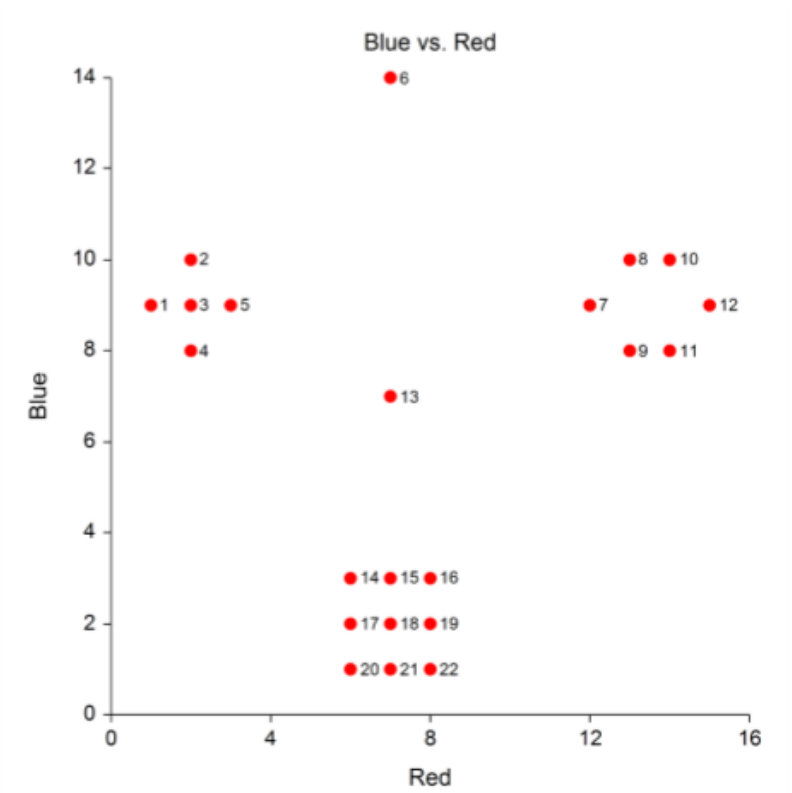


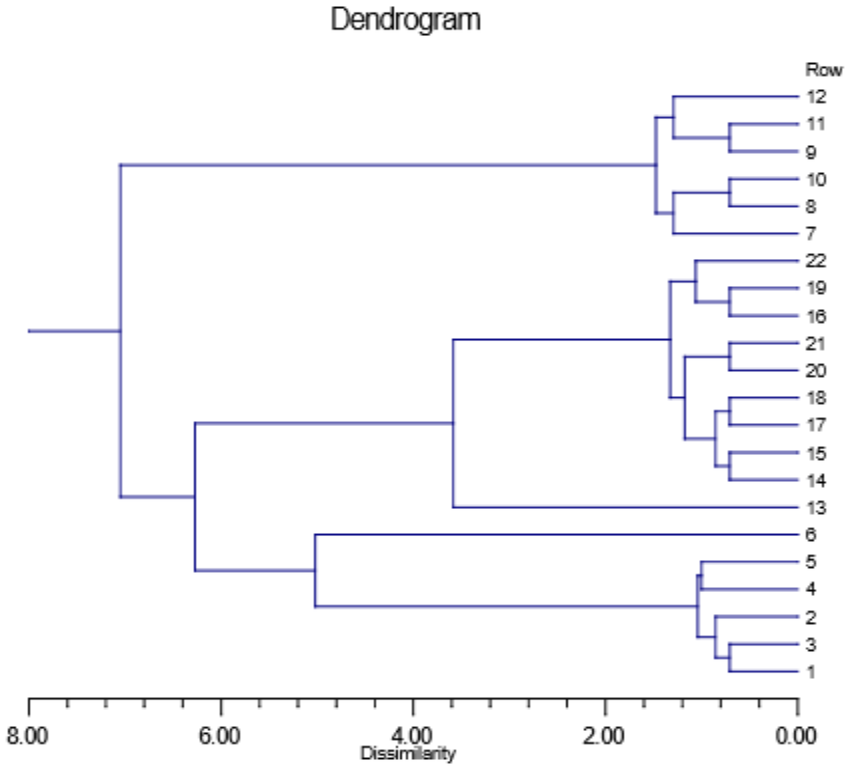
samples	(A,E)	(B,F)	C	D	G
(A,E)	0	0.7778	0.4286	1.0000	0.3750
(B,F)	0.7778	0	0.7143	0.8333	0.7778
C	0.4286	0.7143	0	1.0000	0.3333
D	1.0000	0.8333	1.0000	0	0.8571
G	0.3750	0.7778	0.3333	0.8571	0



Hierarchical (Agglomerative) Clustering

1. Dendrogram is a tree like diagram that summarizes the process of clustering. At the leaf are the records representing the data points
2. Similar records are joined by lines who's vertical length reflects the relative distance between the data points





Source: https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Hierarchical_Clustering-Dendrograms.pdf

Hierarchical (Agglomerative) Clustering Pt 2

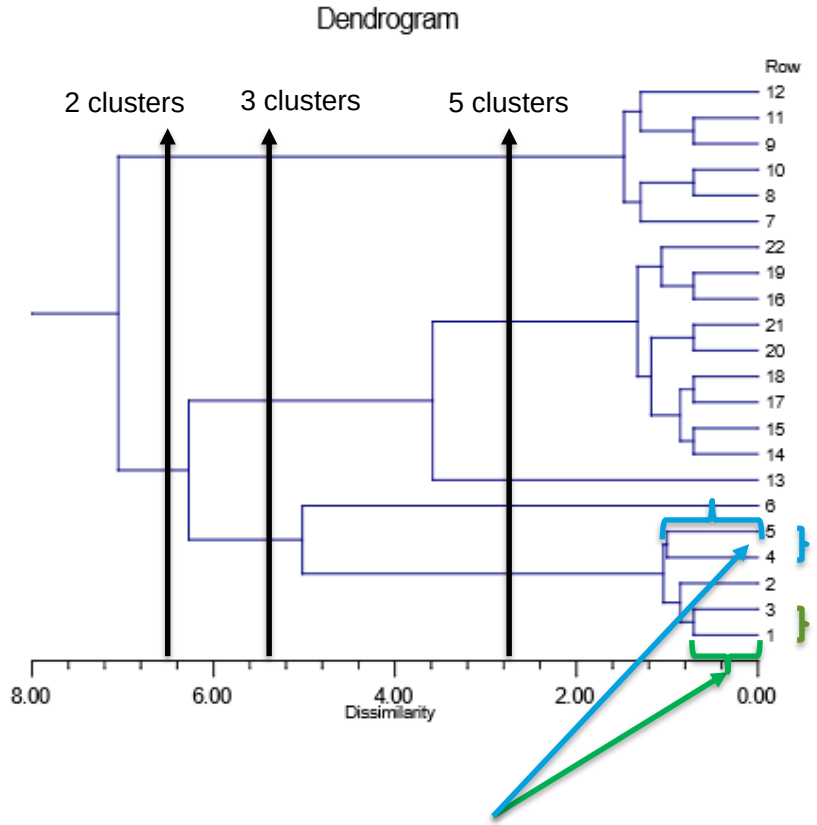
1. The horizontal axis of the dendrogram represents the distance or dissimilarity between clusters (the scale is in reverse order)

2. The vertical axis represents the objects and clusters.

3. Each fusion of two clusters is represented on the graph by the splitting of a horizontal line

4. The horizontal position of the split, shown by the short vertical bar, gives the distance (dissimilarity) between the two clusters

5. When we draw a vertical at any point on the X axis, the number of lines it cuts indicates number of clusters at that value of dissimilarity



Distance/ dissimilarity between 1,3 is less than between 4 and 5. This is reflected in the length of the horizontal bar which is longer for 4,5 compared to 1,3

Source: https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Hierarchical_Clustering-Dendrograms.pdf

Lab 2

Agglomerative Clustering- Lab 2

Lab- 2 Analyze the wines data set using agglomerative clustering

Description – This data is about red wines. The inputs include objective tests (e.g. PH values) and the output is based on sensory data (median of at least 3 evaluations made by wine experts). Each expert graded the wine quality between 0 (very bad) and 10 (very excellent). We have to use K-means clustering to understand if 10 clusters exist and what their characteristics are

Sol: HierarchialClustering_Wine.ipynb

Clustering effectiveness Cophenetic correlation

1. Suppose that the original data {Xi} have been modeled using a cluster method to produce a dendrogram {Ti}

2. Define the following distance measures :
    1. x(i,j)=|Xi−Xj| , the ordinary Euclidean distance between the i th and j th observations

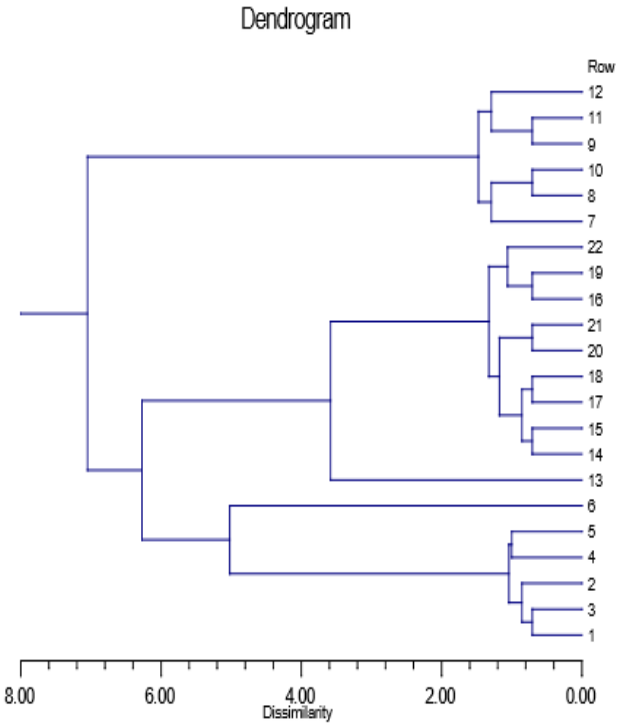
    2. t(i,j)= the dendrogrammatic distance between the model points Ti and Tj. This distance is the height of the node a which these two points are first joined together

    3. Then, letting x be the average of the x(i,j) , and letting t be the average of the t(i,j),

3. Cophenetic correlation coefficient c is defined as

$$c = \frac{\sum_{i<j}(x(i,j)-x)(t(i,j)-t)}{\sqrt{[\sum_{i<j}(x(i,j)-x)^2][\sum_{i<j}(t(i,j)-t)^2]}}$$

4. Values close to 1 is preferred



For comparison of distance methods, clustering techniques and Cophenetic, read

https://journalofinequalitiesandapplications.springeropen.com/track/pdf/10.1186/1029-242X-2013-203?site=journalofinequalitiesandapplications.springeropen.com

Clustering effectiveness Silhouette coefficient

1. Silhouette analysis can be used to study the separation distance between the resulting clusters.

2. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters and thus provides a way to assess parameters like number of clusters visually. This measure has a range of [-1, 1].

3. If the silhouette plot shows values close to one for each observation, the fit was good; if there are many observations closer to zero, it's an indication that the fit was not good.

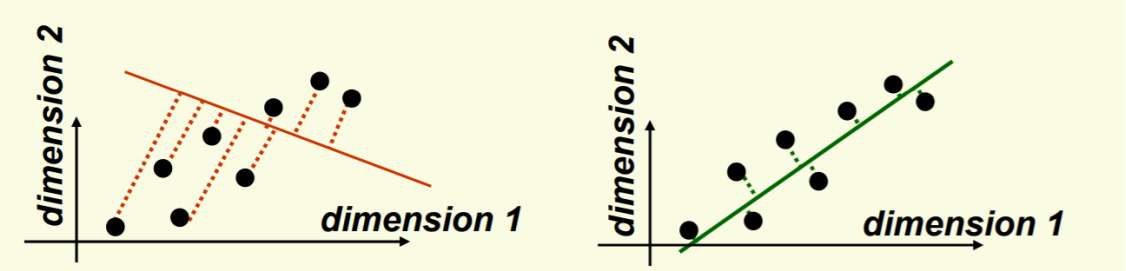
PCA

Principal Component Analysis Concepts

Principal Component Analysis

1. Main idea: seek most accurate data representation in a lower dimensional space

2. Example in 2-D, project data to 1-D subspace (a line) with minimal projection error



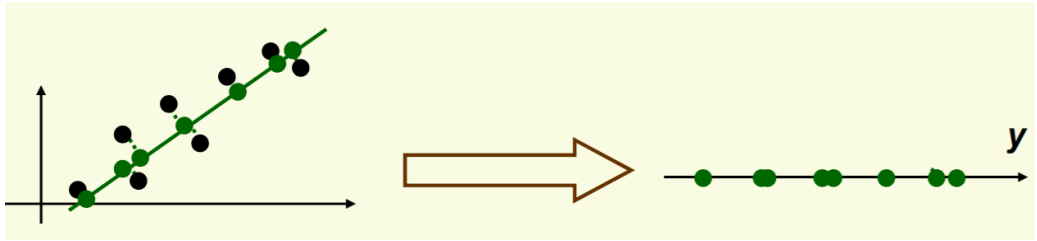
3. In both the pictures above, the data points (black dots) are projected to one line but the second line is closer to the actual points (less projection errors) than first one

4. Notice that the good line to use for projection lies in the direction of largest variance

Ref: http://www.cs.haifa.ac.il/~rita/uml_course/add_mat/PCA.pdf

PCA Pt 2

5. After the data is projected on the best line, need to transform the coordinate system to get 1D representation for vector y



6. Note that new data y has the same variance as old data x in the direction of the green line

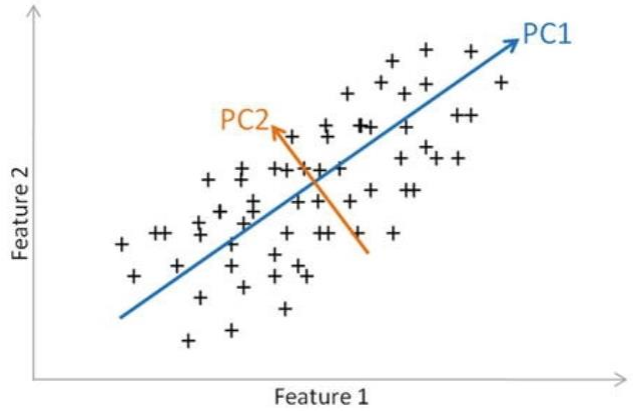
7. PCA preserves largest variances in the data

Ref: http://www.cs.haifa.ac.il/~rita/uml_course/add_mat/PCA.pdf

PCA Pt 3

8. In general PCA on n dimensions will result in another set of new n dimensions. The one which captures maximum variance in the underlying data is the principal component 1, principal component 2 is orthogonal to it

9. Example in 2-D, project data to 1-D subspace (a line) with minimal projection error



Ref: http://www.cs.haifa.ac.il/~rita/uml_course/add_mat/PCA.pdf

Mechanics of Principal Component Analysis

Mechanics of Principal Component Analysis

http://setosa.io/ev/principal-component-analysis/

Principal Component Analysis steps

1. Begins by standardizing the data. Data on all the dimensions are subtracted from their means to shift the data points to the origin. i.e. the data is centered on the origins

2. Generate the covariance matrix / correlation matrix for all the dimensions

3. Perform eigen decomposition, that is, compute eigen vectors which are the principal components and the corresponding eigen values which are the magnitudes of variance captured

4. Sort the eigen pairs in descending order of eigen values and select he one with the largest value. This is the first principal component that covers the maximum information from the original data

Ref: http://www.cs.haifa.ac.il/~rita/uml_course/add_mat/PCA.pdf

Principal Component Analysis (Performance issues)

1. PCA effectiveness depends upon the scales of the attributes. If attributes have different scales, PCA will pick variable with highest variance rather than picking up attributes based on correlation

2. Changing scales of the variables can change the PCA

3. Interpreting PCA can become challenging due to presence of discrete data

4. Presence of skew in data with long thick tail can impact the effectiveness of the PCA (related to point 1)

5. PCA assumes linear relationship between attributes. It is ineffective when relationships are non linear

Lab 3

Principal Component Analysis steps

Lab-3 Principal Component Analysis on iris data set

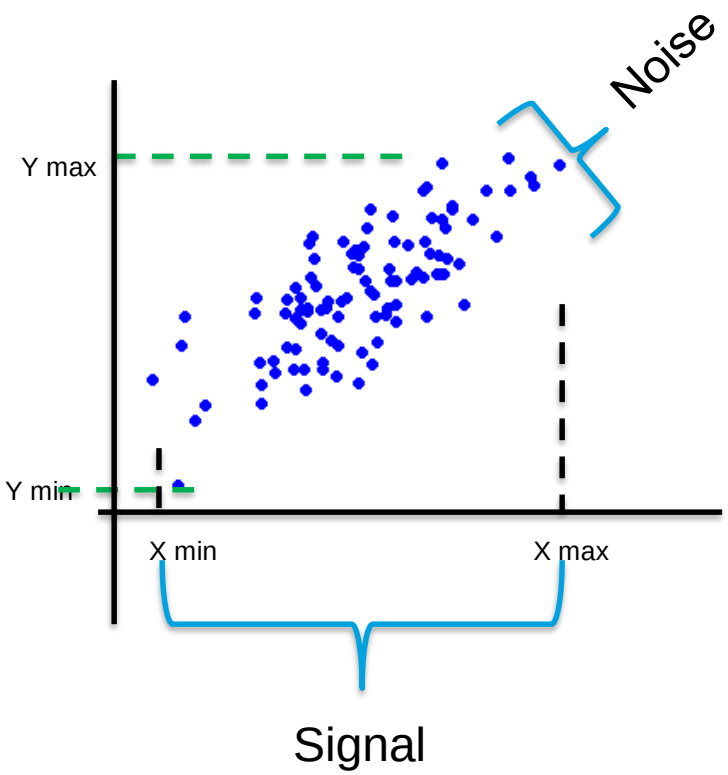
Description – Explore the iris data set and perform PCA

The data set is winequality-red.csv

Sol: PCA-iris.ipynb

Principal Component Analysis (Signal to noise ratio)

Principal Component Analysis (Signal to noise ratio)




```
X_std_df = pd.DataFrame(X_std)
axes = pd.plotting.scatter_matrix(X_std_df)
plt.tight_layout()
```

Signal – all valid values for a variable (show between max and min values for x axis and y axis). Represents a valid data

Noise – The spread of data points across the best fit line. For a given value of x, there are multiple values of y (some on line and some around the line). This spread is due to random factors

Signal to Noise Ratio – Variance of signal / variance in noise. $\frac{\sigma^2_{signal}}{\sigma^2_{noise}}$

Greater the SNR the better the model will be

Principal Component Covariance Matrix

Principal Component Covariance Matrix

1. Variance is measured within the dimensions and co-variance is among the dimensions

$$\mathrm{var}(X) = \frac{\sum_{i=1}^{n} (X_i - \overline{X})(X_i - \overline{X})}{(n-1)}$$

$$\mathrm{cov}(X, Y) = \frac{\sum_{i=1}^{n} (X_i - \overline{X})(Y_i - \overline{Y})}{(n-1)}$$

2. Express total variance (variance and cross variance between dimensions as a matrix (variance matrix)

3. Covariance matrix is a mathematical representation of the total variance of individual dimension and across dimensions .

$$C = \begin{bmatrix} \mathrm{cov}(X,X) & \mathrm{cov}(X,Y) & \mathrm{cov}(X,Z) \\ \mathrm{cov}(Y,X) & \mathrm{cov}(Y,Y) & \mathrm{cov}(Y,Z) \\ \mathrm{cov}(Z,X) & \mathrm{cov}(Z,Y) & \mathrm{cov}(Z,Z) \end{bmatrix}$$

Covariance matrix for three dimensions x,y and z

```
eig_vals, eig_vecs = np.linalg.eig(cov_matrix)
```

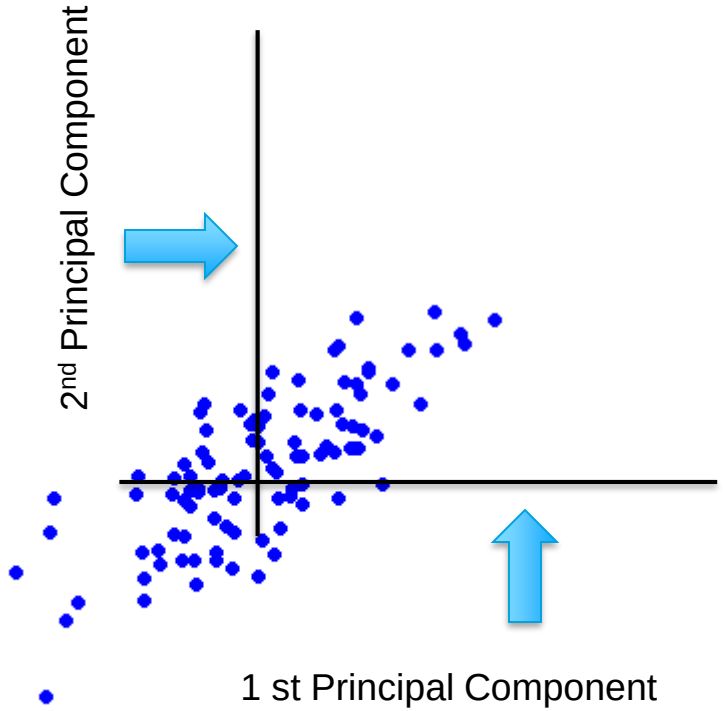
Improving SNR through PCA (Scaling the dimensions)

Improving SNR through PCA (Scaling the dimensions)

1. The mean is subtracted from all the points on both dimensions i.e. (xi – xbar) and (yi – ybar)

2. The dimensions are transformed using algebra into new set of dimensions

3. The transformation is a rotation of axes in mathematical space




```
X_std = StandardScaler().fit_transform(X)
eig_vals, eig_vecs = np.linalg.eig(cov_matrix)
```

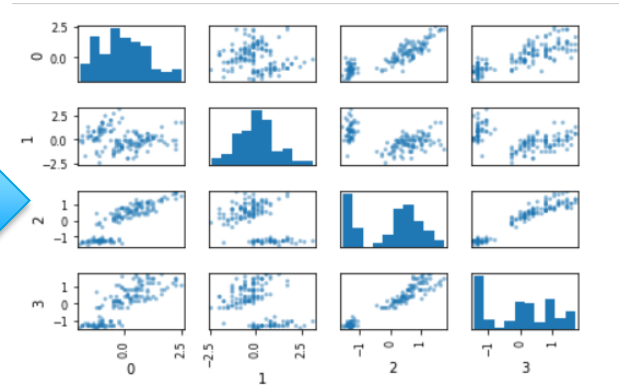
(Calculating total variance (covariance and variance)

PCA (Calculating total variance (covariance and variance)

4. Multiplying the two matrices produces a matrix of total variance also called covariance matrix (a square and symmetric matrix).

$$A \times A^T = C = \begin{bmatrix} \mathrm{cov}(X,X) & \mathrm{cov}(X,Y) & \mathrm{cov}(X,Z) \\ \mathrm{cov}(Y,X) & \mathrm{cov}(Y,Y) & \mathrm{cov}(Y,Z) \\ \mathrm{cov}(Z,X) & \mathrm{cov}(Z,Y) & \mathrm{cov}(Z,Z) \end{bmatrix}$$

```
Covariance Matrix
%s [[ 1.00671141 -0.11010327  0.87760486  0.82344326]
 [-0.11010327  1.00671141 -0.42333835 -0.358937  ]
 [ 0.87760486 -0.42333835  1.00671141  0.96921855]
 [ 0.82344326 -0.358937    0.96921855  1.00671141]]
```



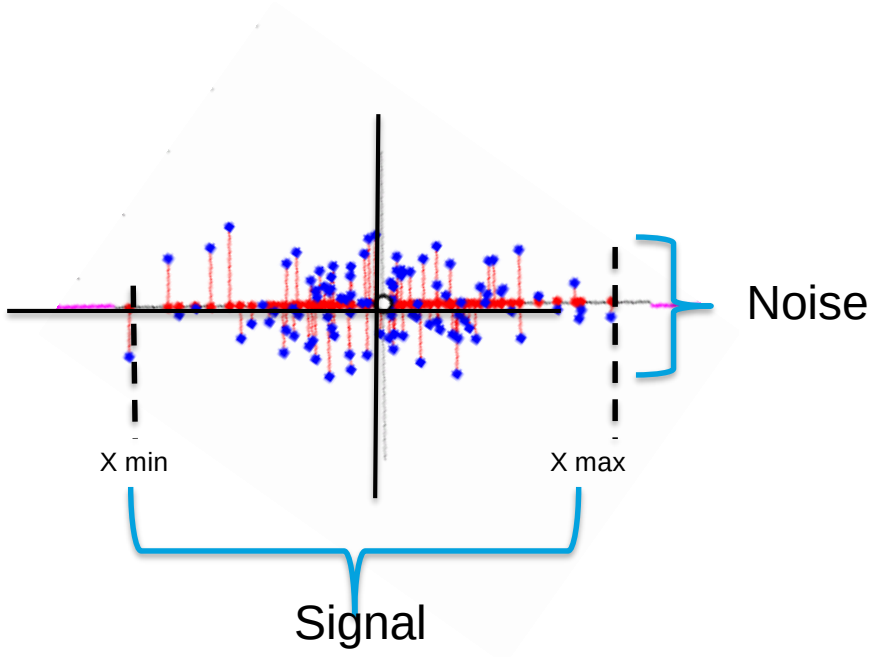
Improving SNR through PCA (Principal components)

Improving SNR through PCA (Principal components)

5. The original data points are now represented by the red dots on new dimensions

6. It also introduces error of representation (vertical red lines from the blue dots to corresponding red dots on the new dimension)

7. The axis rotation is done such that the new dimension captures max variance in the data points and also reduces total error of representation




```
print('Eigen Vectors \n%s', eig_vecs)
print('\n Eigen Values \n%s', eig_vals)
```

Properties of principal components and their covariance matrix

Properties of principal components and their covariance matrix

8. Thus to find principal components we need to get the diagonal matrix $\mathbf{B}\mathbf{B}^{\mathrm{T}}$ from the original covariance matrix $A\,A^{T}$

$$\begin{bmatrix} \mathrm{cov}(X,X) & \mathrm{cov}(X,Y) & \mathrm{cov}(X,Z) \\ \mathrm{cov}(Y,X) & \mathrm{cov}(Y,Y) & \mathrm{cov}(Y,Z) \\ \mathrm{cov}(Z,X) & \mathrm{cov}(Z,Y) & \mathrm{cov}(Z,Z) \end{bmatrix}$$

$A\,A^{T}$

$$\Rightarrow \begin{pmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_p^2 \end{pmatrix}$$

$\mathbf{B}\mathbf{B}^{\mathrm{T}}$

9. For this we have to transform the matrix **A** to a new matrix **B** such that the covariance matrix of **B** ($\mathbf{B}\mathbf{B}^{\mathrm{T}}$), is a **<u>diagonal matrix</u>** (Ref to part 2, bullet 5)

PCA for dimensionality reduction

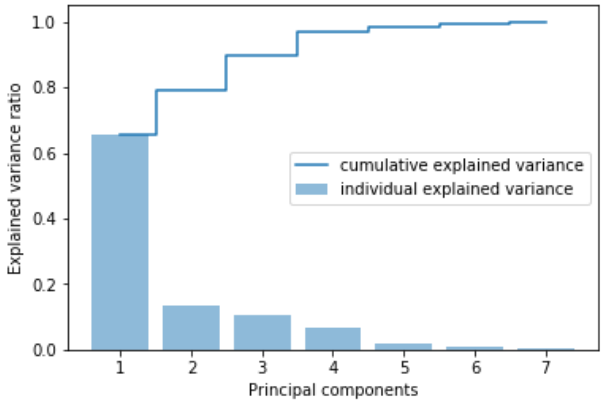
PCA for dimensionality reduction

1. PCA can also be used to reduce dimensions

2. Arrange all eigen vectors along with corresponding eigen values in descending order of eigen values

3. Plot a cumulative eigen_value graph as shown below

4. Eigen vectors with insignificant contribution to total eigen values can be removed from analysis (for e.g. eigen vector 6 and 7 below)



END

Thanks