Packages to be known for Machine learning

1. Numpy
2. Pandas
3. Scipy
4. Scikit-learn
5. Matplotlib
6. Seaborn

Scikit-learn :

* Lots of machine learning libraries.
* In built classification, Regression and clustering algorithms
* Build with flexibility to use Numpy and Scipy which helps easy to play with data.
* Most preprocessing tasks are inbuilt in scikit-learn such as Data preprocessing, Future Selection, extraction and train – test splitting, algorithm, prediction, evaluation and exploring the model.

**Supervised vs Unsupervised learning:**

**Supervised learning:**

Train the model with labelled dataset.

1. Classification – process of predicting discrete class label or category.

Eg. Predicting the class such as finding the cancer in cells of human, customer churn in telecom.

1. Regression – process of predicting a continuous value.

Eg. Predicting the house price, stock price.

**Unsupervised learning:**

We do not supervise the model, but the model trains on its own and discover information that are invisible to human eye.

1. Dimension reduction – reducing the redundant future to make classification easier.
2. Density estimation – explore data to find substructure
3. Market basket analysis or Association – based on theory of if a customer buys certain product, then he is likely to buy another product as well.
4. Clustering – grouping of data points that are similar somehow. Eg. Segment customer based on credit score in banking. Used for Discovering structure, summarization and Anomaly detection.

**Few more Machine learning techniques:**

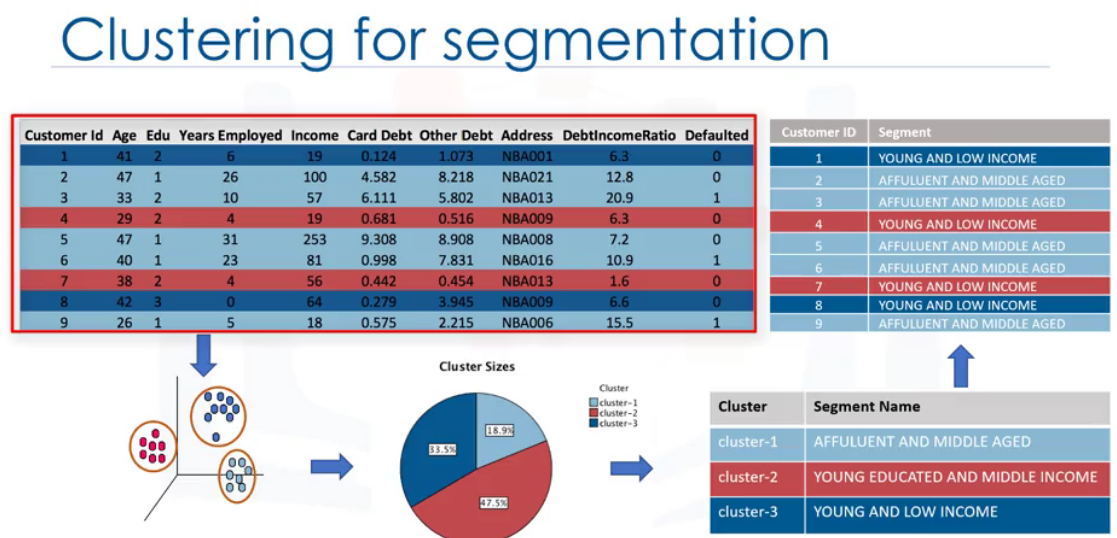
Anomaly detection – discovering abnormal cases eg. Credit card fraud detection

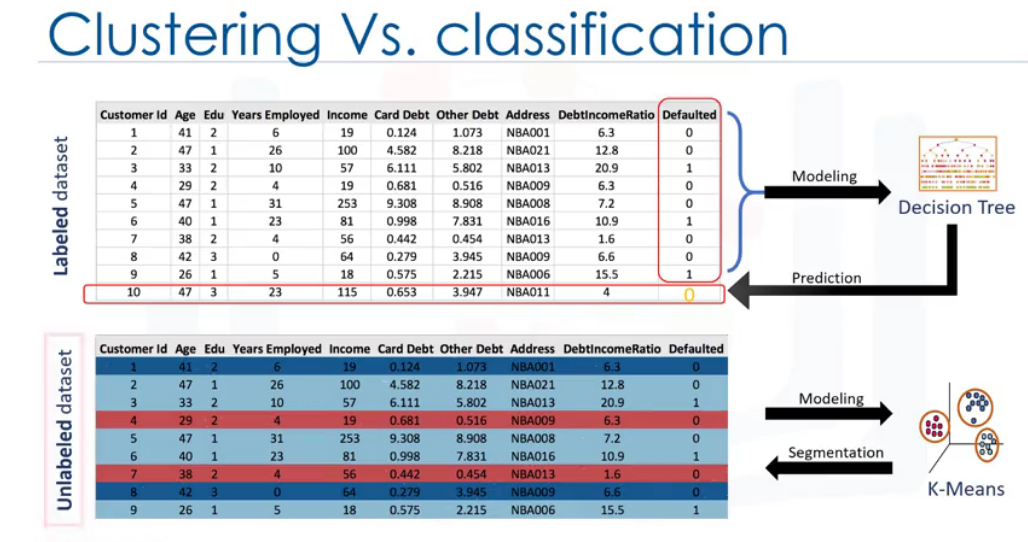
Sequence mining – predict the next occurrence eg. Click stream

Recommender systems – Recommending items eg Netflix, youtube.

**Clustering:**

* Customer segmentation to group customer based on specific category so that the market can focus on specific group.
* Clustering is the best approach to group data only unsupervised. (Group the object that are similar to other object and dissimilar to data points in another cluster).
* Its partition each group into a mutually exclusive cluster.





**Applications of Clustering:**





**Why to use Clustering:**

* Exploring data analysis
* Summary generation
* Outlier detection
* Finding duplicates, Data preprocessing.

**Types of Clustering:**

Partition based Clustering:

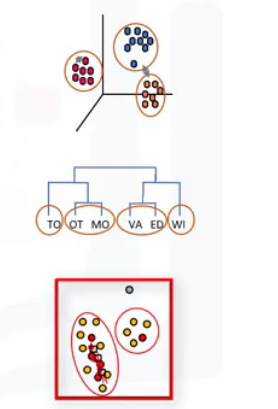
* Used for medium or large volume of data set. Relatively efficient.
* Eg. K-means, K-median , Fuzzy c-Means.

Hierarchical Clustering:

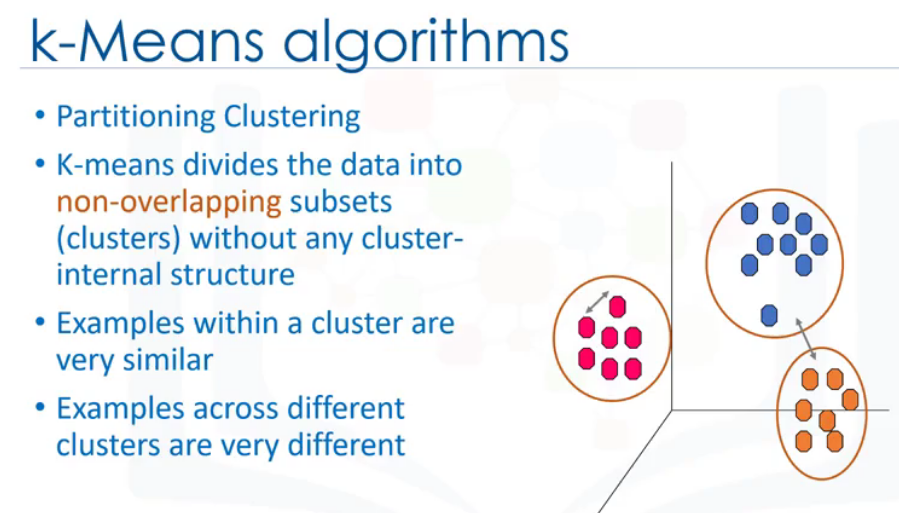
* Used in very less volume of data set.
* Produce trees of cluster.
* Eg. Agglomerative, Divisive

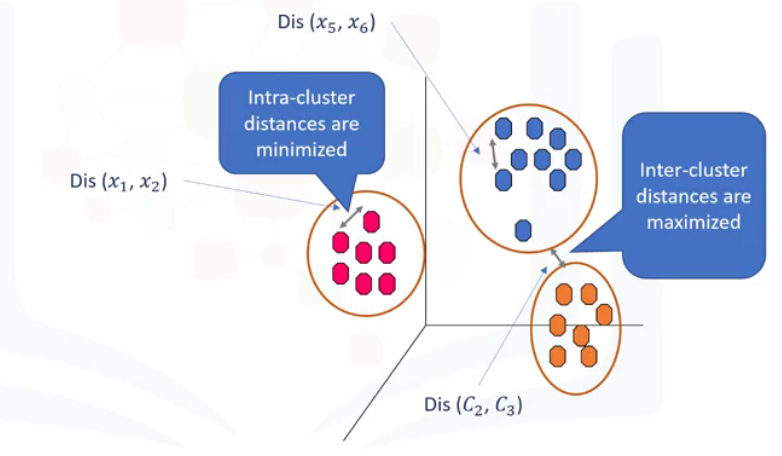
Density based clustering:

* Produce arbitrary based cluster. Good dealing with spatial data set and noisy data set.
* Eg DB scan



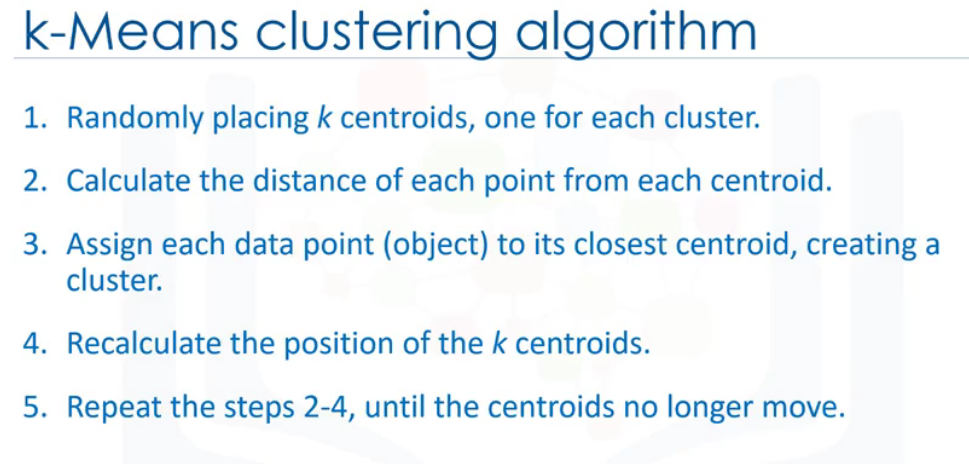
**K- Means Clustering:**



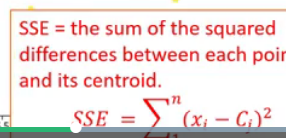


Calculate the distance using Euclidian distance formula. Here Euclidian distance is used based on the domain.

Algorithm:



**Choosing the centroid randomly causes very high error.**

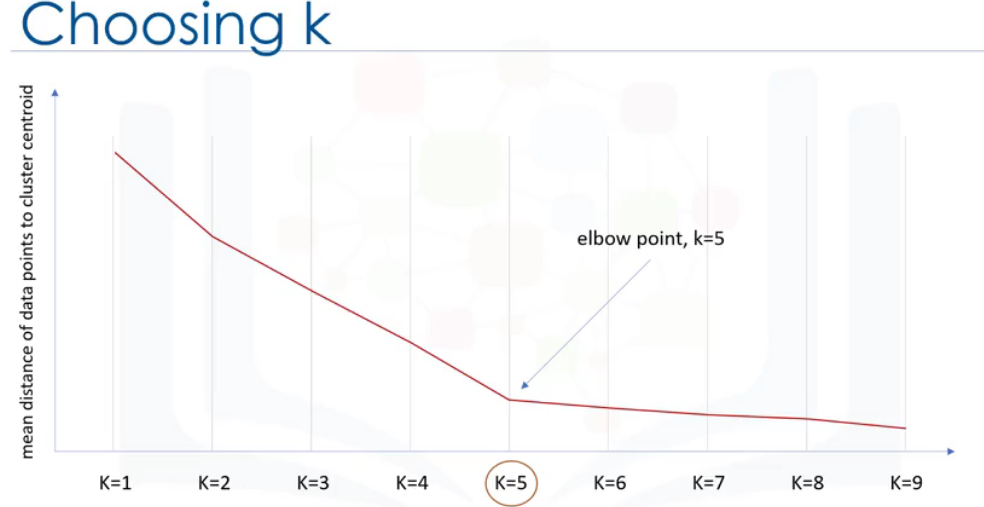


Repeat step 2 to 4 until the algorithm converges. It is a heuristic algorithm, there is no global optimum. It may result in local optimum. To resolve the problem, we must run the algorithm multiple times.

**K- means accuracy:**

**Internal Approach:** Average the distance between the datapoints in the cluster.

**Choosing K:**



**Packages:**

import random

import numpy as np

import matplotlib.pyplot as plt

from sklearn.cluster import KMeans

from sklearn.datasets.samples\_generator import make\_blobs

**Hierarchical clustering:**

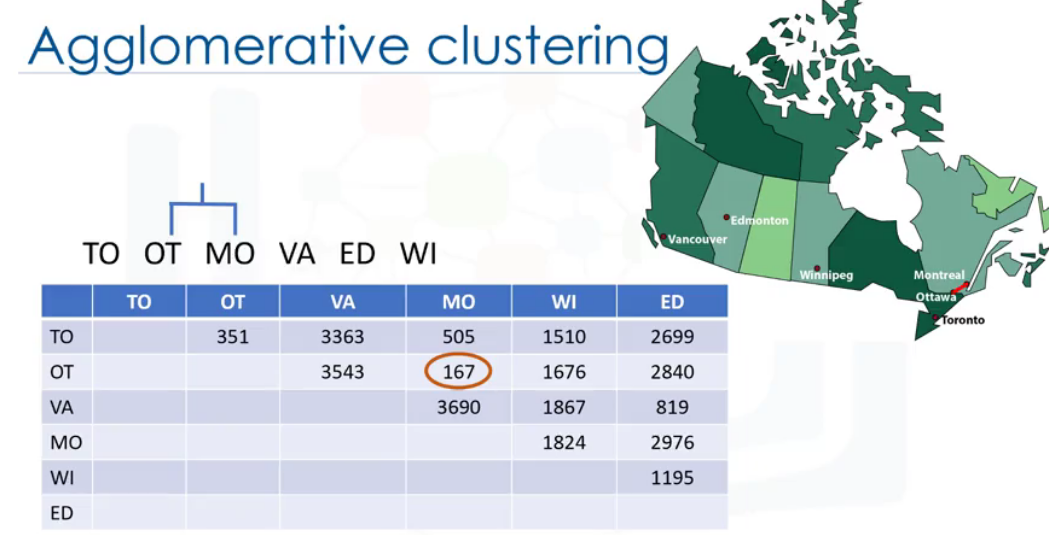
Builds hierarchy of clusters where each node is a cluster consisting of clusters of its daughter node.

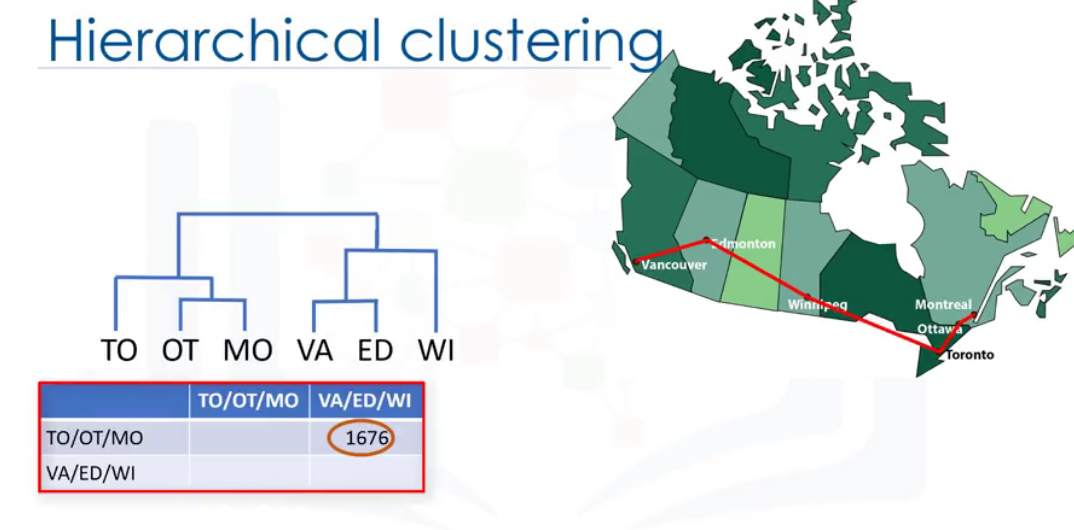
Two types of strategy:

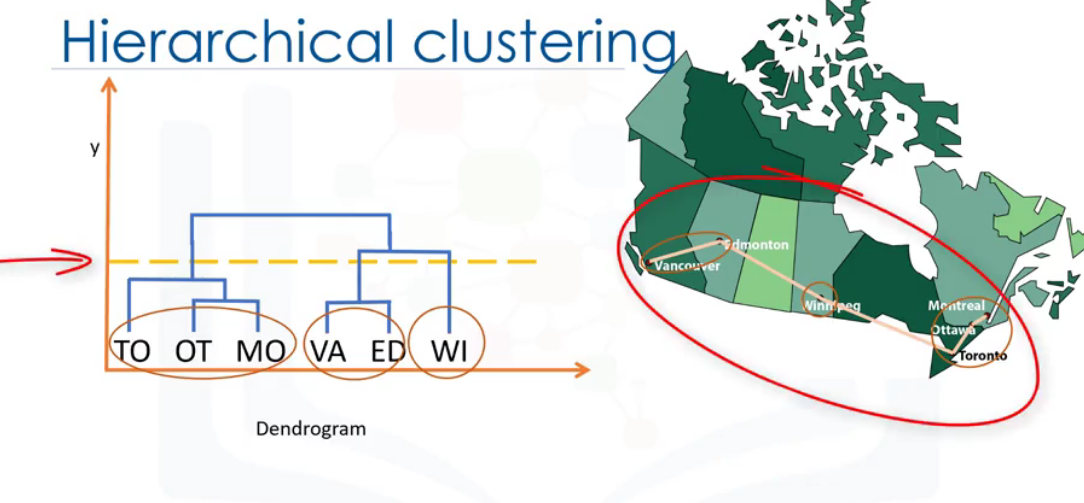
Divisive – top down approach

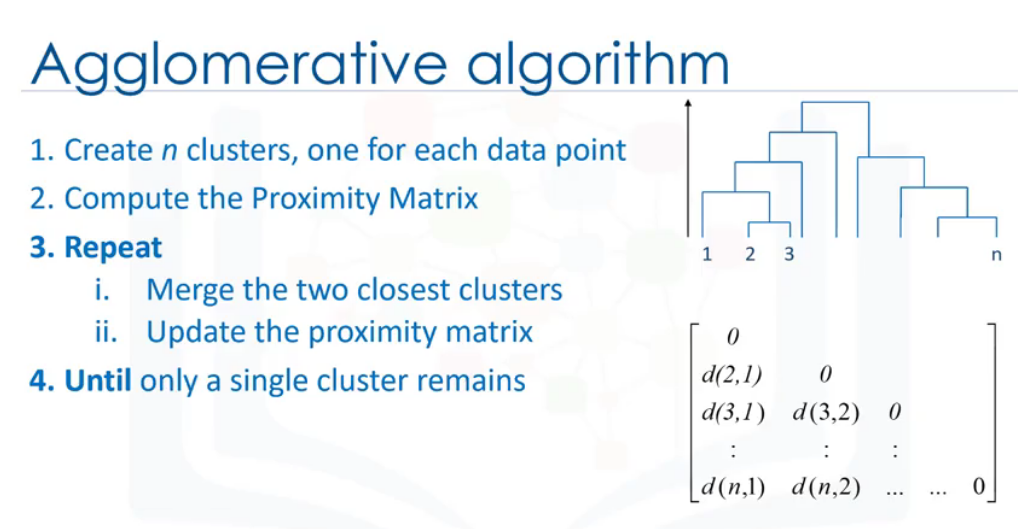
**Agglomerative** (Collect things) - bottom up approach (each cluster is started and pairs of clusters are combined as we move up the hierarchy.

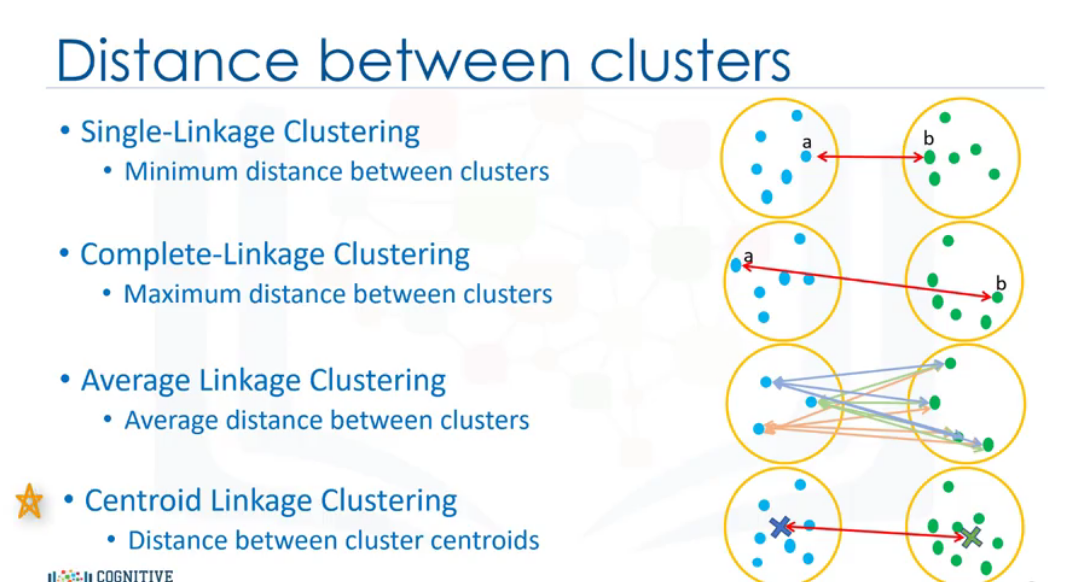
It moves up the ladder by progressively merging cluster.

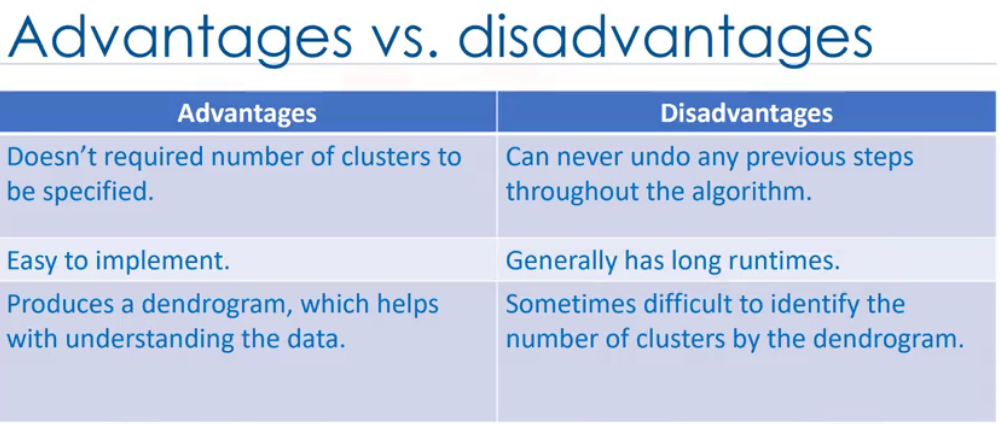


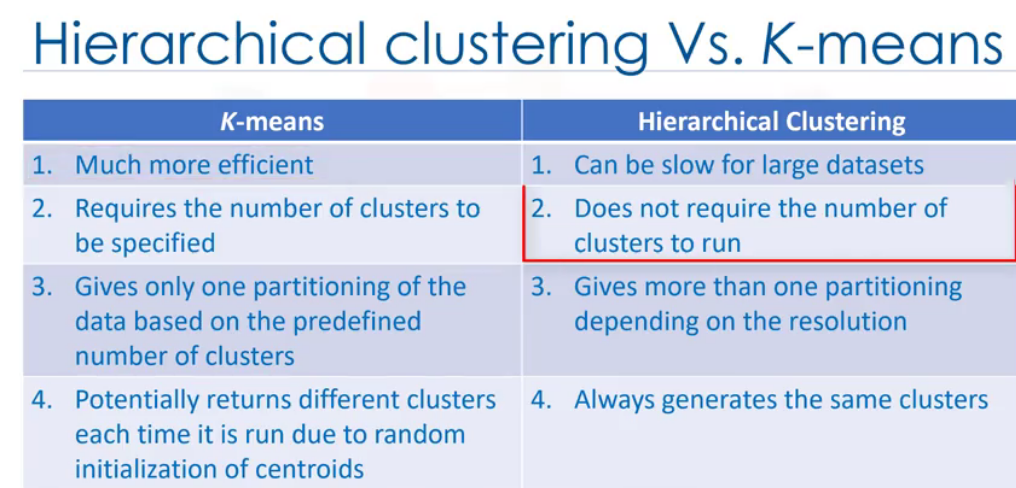












# References

Aghabozorgi, S. (n.d.). *coursera.* Retrieved from www.coursera.org: https://www.coursera.org/learn/machine-learning-with-python