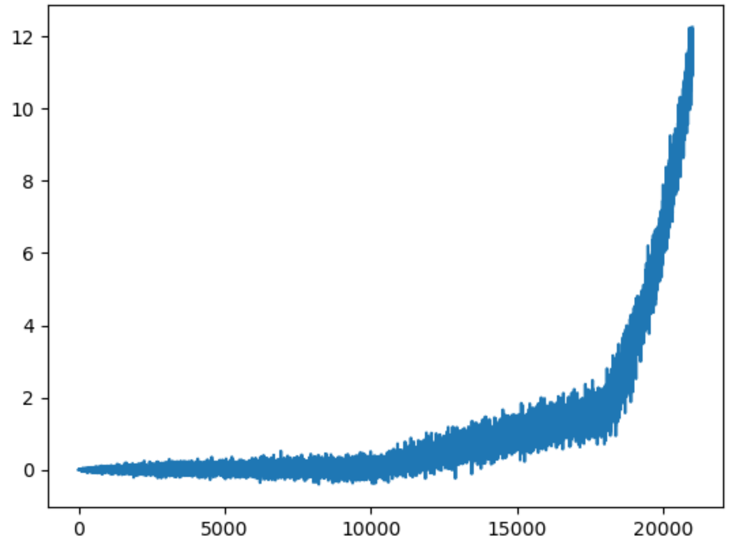
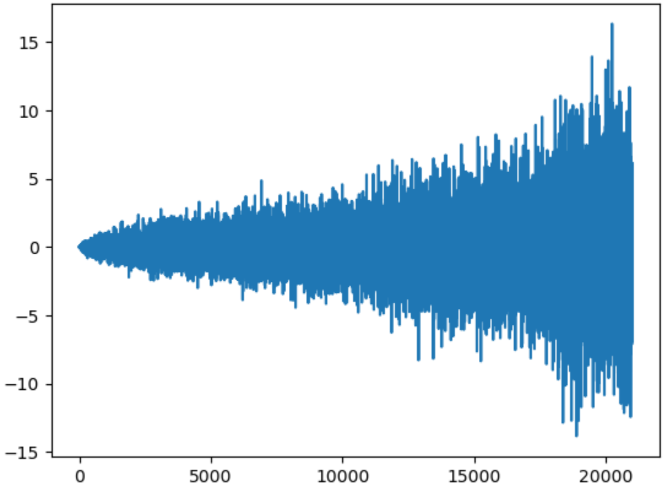
Report 5

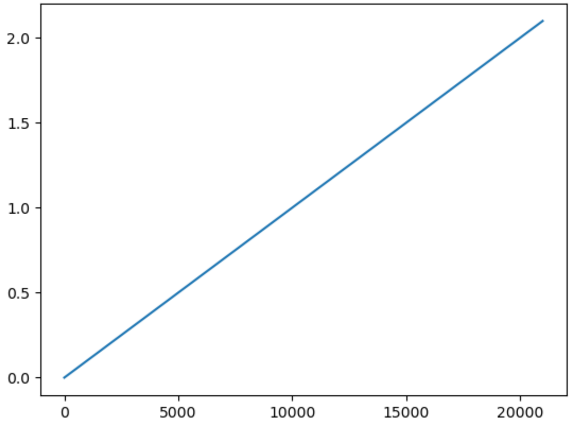
Signal

signal

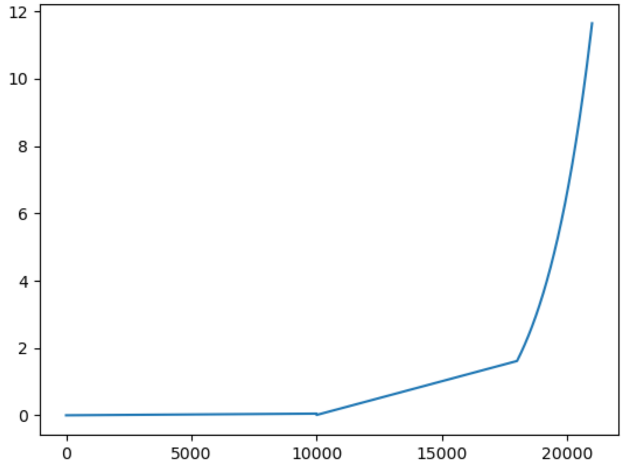




noise



time



Deterministic

Resampling (whole code)

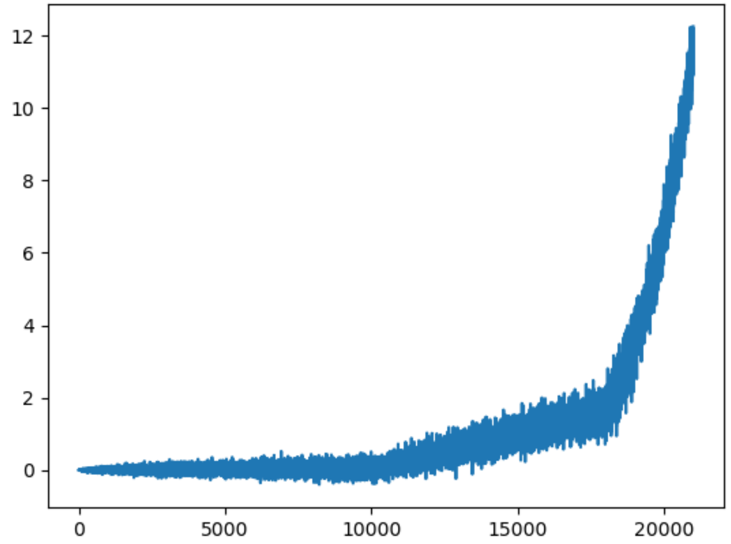
Resampler.interpolate(method='linear', axis=0, limit=None, inplace=False)

```
import numpy as np  //add some library

import pandas as pd


file = pd.read_csv('Signalnumber2.csv' ,skiprows=0) // reading Signal (excel format)

data = file.to_numpy() // convert to Array

new_series = pd.Series(data[:,0] ,index=pd.period_range('2018-01-01', freq='Q',
periods= 16383)) // convert to series and add time for indexes


upsampled = new_series.resample('M')

interpolated = upsampled.interpolate(method='linear').to_numpy() //resampling by
linear method


np.savetxt("SignalWithNoise.csv", interpolated, delimiter=",") // saving to csv file
```

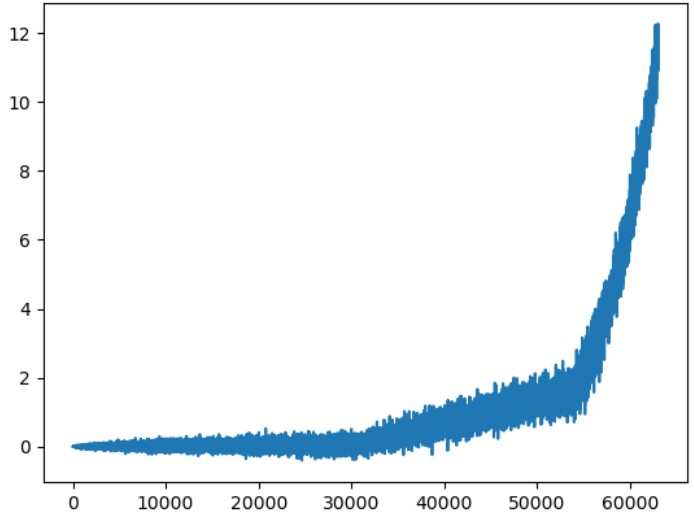
Resampling (plot)

method : ‘linear’: Ignore the index and treat the values as equally spaced. This is the only method supported on Multi Indexes.
We can create more and more data .

21000 datapoint



63000 datapoint



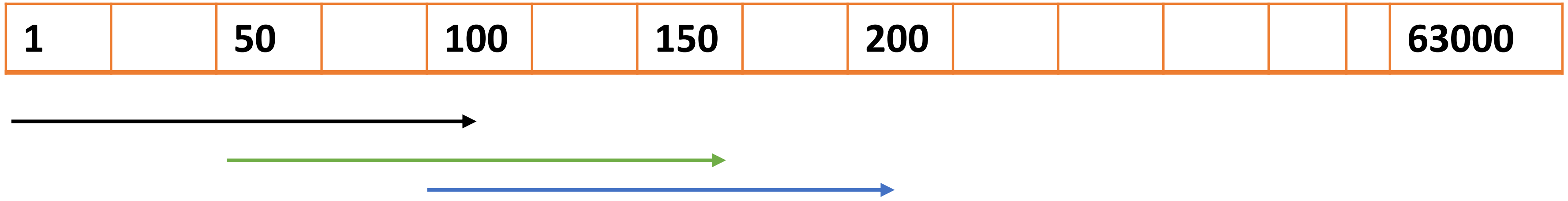
Data + feature

1	2	3	4	-	-	-	-	100						200					300	-	-	-	62998	62999	63000

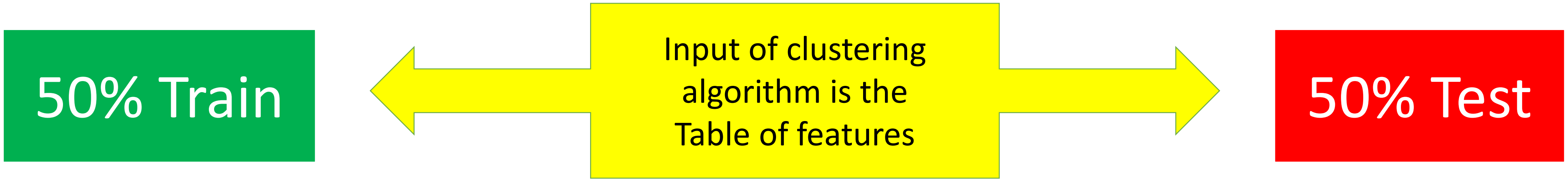
Number of segments = Data/100 +50% overlapping

	index	skew	max	rms	mean	median	Kur	std
Seg 1 →	1							
	2							
	3							
	…							
	1256							
	1257							
Seg 1258 →	1258							

Overlapping (more explain)



1		50		100		150		200						63000



50% Train

Input of clustering algorithm is the Table of features

50% Test

Score : Silhouette Score

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html

Select the optimal number of clusters

Select the optimal number of clusters based on multiple clustering validation metrics like Gap Statistic, Silhouette Coefficient, Calinski-Harabasz Index etc.

Gap Statistic

Elbow Method: https://en.wikipedia.org/wiki/Elbow_method_(clustering)

Silhouette Coefficient

Calinski-Harabasz Index

Davies-Bouldin Index

Dendrogram: https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html

Bayesian information criterion (BIC)

elbow method

It is the most popular method for determining the optimal number of clusters. The method is based on calculating the Within-Cluster-Sum of Squared Errors (WSS) for different number of clusters (k) and selecting the k for which change in WSS first starts to diminish.

The idea behind the elbow method is that the explained variation changes rapidly for a small number of clusters and then it slows down leading to an elbow formation in the curve.

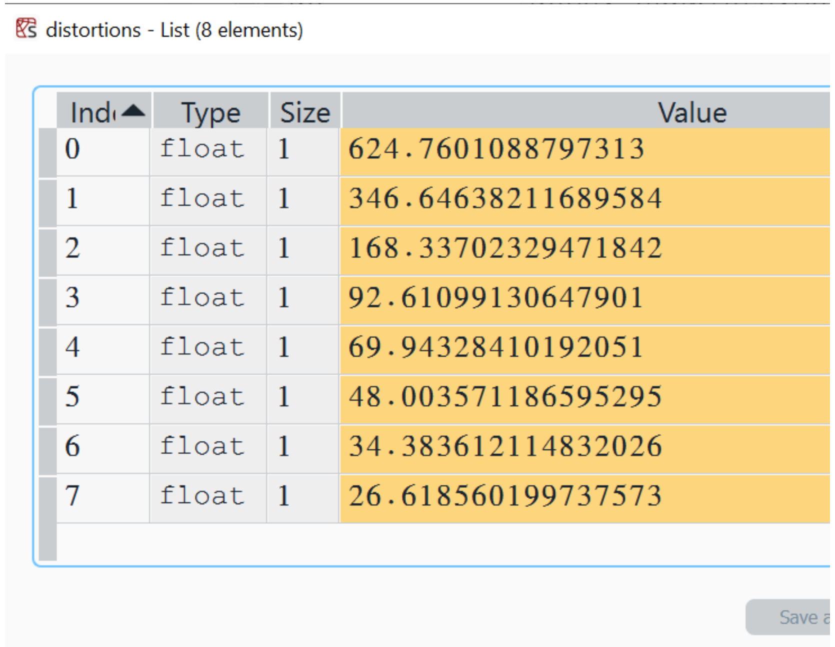
The elbow point is the number of clusters we can use for our clustering algorithm.

Further details on this method can be found in this paper by Chunhui Yuan and Haitao Yang.

using elbow method (use k-means clustering)

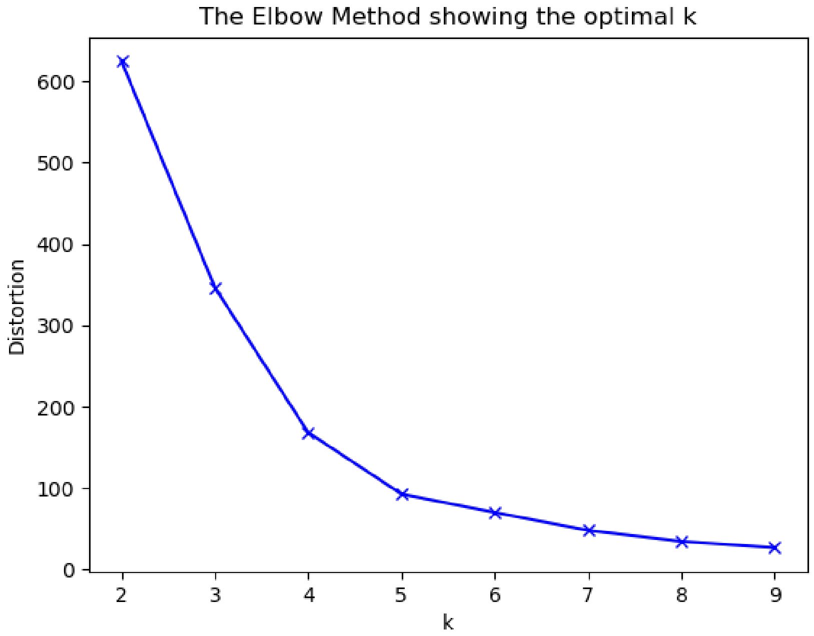
Explained variance. The "elbow" is indicated by the red circle. The number of clusters chosen should therefore be 3 or 4.
Distortion: Sum of squared distances of samples to their closest cluster center, weighted by the sample weights if provided. (the highest is better)

K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9		
Score: 0.850	Score: 0.707	Score: 0.713	Score: 0.728	Score: 0.697	Score: 0.697	Score: 0.696	Score: 0.702		



distortions - List (8 elements)

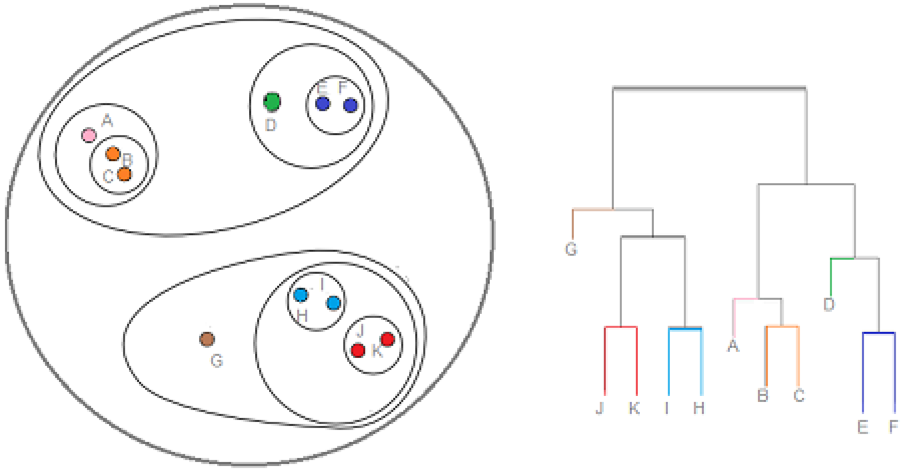
Ind	Type	Size	Value
0	float	1	624.7601088797313
1	float	1	346.64638211689584
2	float	1	168.33702329471842
3	float	1	92.61099130647901
4	float	1	69.94328410192051
5	float	1	48.003571186595295
6	float	1	34.383612114832026
7	float	1	26.618560199737573



The Elbow Method showing the optimal k

Dendrogram

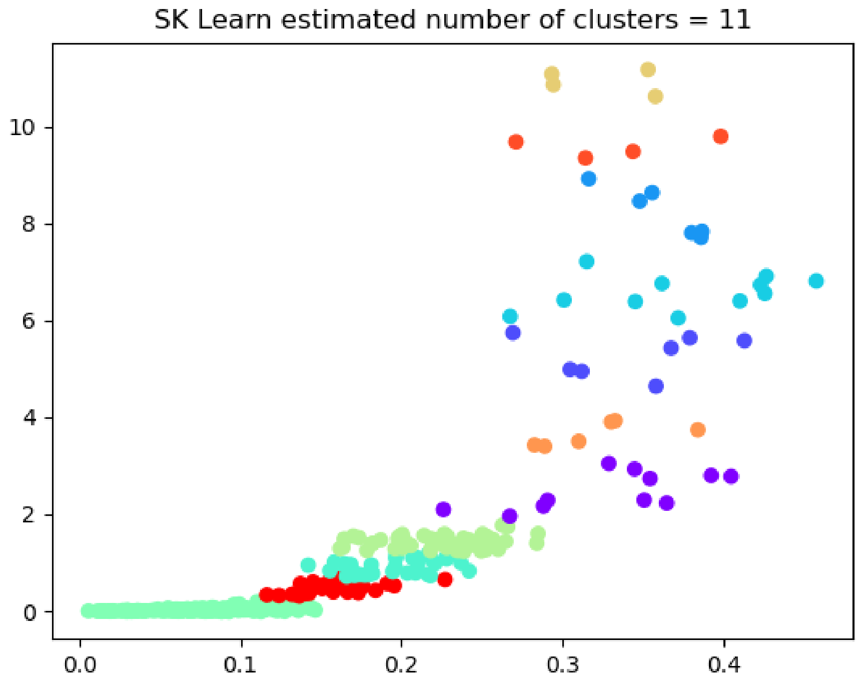
- A dendrogram is a type of tree diagram showing hierarchical clustering — relationships between similar sets of data
- This technique is specific to the agglomerative hierarchical method of clustering.
- The agglomerative hierarchical method of clustering starts by considering each point as a separate cluster and starts joining points to clusters in a hierarchical fashion based on their distances.
- To get the optimal number of clusters for hierarchical clustering, we make use a dendrogram which is tree-like chart that shows the sequences of merges or splits of clusters.

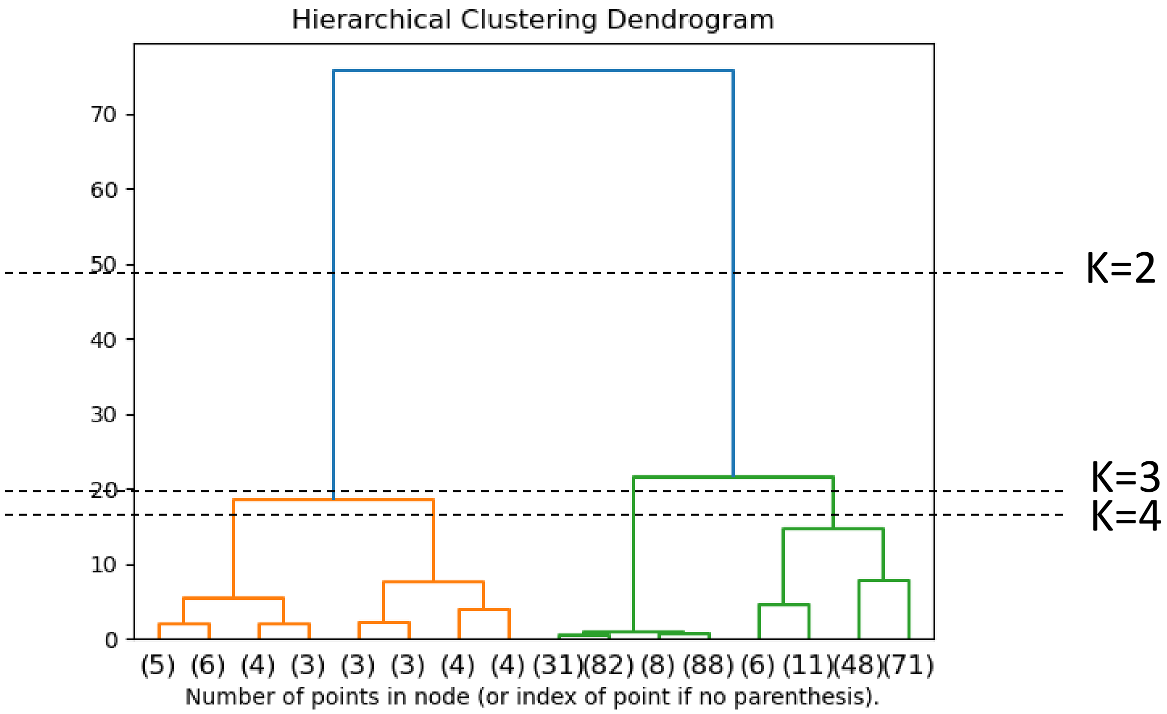


Dendrogram (use Agglomerative Clustering)

But we can use 3 or 4 cluster by seeing dendrogram diagram.

The program said : Optimal k is 11!





Feature Selection

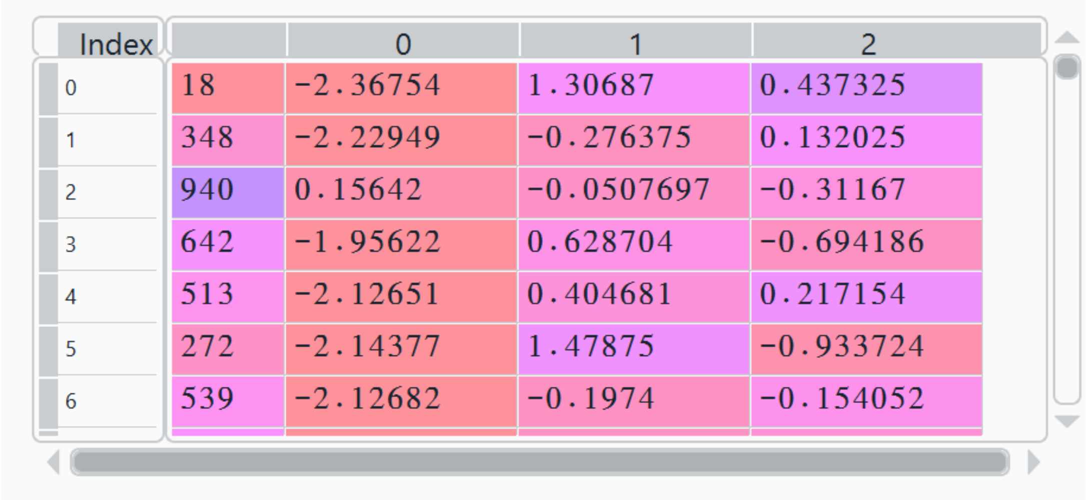
2. Principal Component Analysis

Percentage of variance explained by each of the selected components.

Explained Variance: [0.96876454 0.02256079 0.0074725]

Principal axes in feature space , representing the directions of maximum variance in the data. Equivalently, the right singular vectors of the centered input data, parallel to its eigenvectors.

PCA1 PCA2 PCA3



Index		0	1	2
0	18	-2.36754	1.30687	0.437325
1	348	-2.22949	-0.276375	0.132025
2	940	0.15642	-0.0507697	-0.31167
3	642	-1.95622	0.628704	-0.694186
4	513	-2.12651	0.404681	0.217154
5	272	-2.14377	1.47875	-0.933724
6	539	-2.12682	-0.1974	-0.154052

10 kind of clustering

List of Clustering + Reference

- ✓ spectral clustering (Spectral clustering is **a technique with roots in graph theory**, where the approach is used to identify communities of nodes in a graph based on the edges connecting them. The method is flexible and allows us to cluster non graph data as well)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.SpectralClustering.html

https://en.wikipedia.org/wiki/Spectral_clustering

- ✓ birch clustering (**Balanced Iterative Reducing and Clustering** using Hierarchies (BIRCH) is a clustering algorithm that can cluster large datasets by first generating a small and compact summary of the large dataset that retains as much information as possible.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.Birch.html

https://en.wikipedia.org/wiki/BIRCH

- ✓ agglomerative clustering (The agglomerative clustering is **the most common type of hierarchical clustering used to group objects in clusters based on their similarity**. ... Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

https://en.wikipedia.org/wiki/Hierarchical_clustering

List of Clustering + Reference

- ✓ k-means clustering (K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

https://en.wikipedia.org/wiki/K-means_clustering

- ✓ mini-batch k-means clustering (The Mini-batch K-means clustering algorithm is **a version of the standard K-means algorithm in machine learning**. It uses small, random, fixed-size batches of data to store in memory, and then with each iteration, a random sample of the data is collected and used to update the clusters.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.MiniBatchKMeans.html

https://www.geeksforgeeks.org/ml-mini-batch-k-means-clustering-algorithm/

- ✓ affinity propagation clustering (Affinity propagation (AP) is **a graph based clustering algorithm** similar to k Means or K medoids, which does not require the estimation of the number of clusters before running the algorithm. Affinity propagation finds "exemplars" i.e. members of the input set that are representative of clusters.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AffinityPropagation.html

https://en.wikipedia.org/wiki/Affinity_propagation

- ✓ DB scan clustering (The principle of DBSCAN is **to find the neighborhoods of data points exceeds certain density threshold**. The density threshold is defined by two parameters: the radius of the neighborhood (eps) and the minimum number of neighbors/data points (minPts) within the radius of the neighborhood.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

https://en.wikipedia.org/wiki/DBSCAN

List of Clustering + Reference

- ✓ optics clustering (Ordering points to identify the clustering structure (OPTICS) is an **algorithm for finding density-based clusters in spatial data**. ... Its basic idea is similar to DBSCAN, but it addresses one of DBSCAN's major weaknesses: the problem of detecting meaningful clusters in data of varying density.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.OPTICS.html

https://en.wikipedia.org/wiki/OPTICS_algorithm

- ✓ mean shift clustering (Mean shift clustering using a flat kernel. Mean shift clustering aims to discover "blobs" in a smooth density of samples. It is a **centroid-based algorithm**, which works by updating candidates for centroids to be the mean of the points within a given region. ... If not set, the seeds are calculated by clustering.)

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.MeanShift.html

https://en.wikipedia.org/wiki/Mean_shift

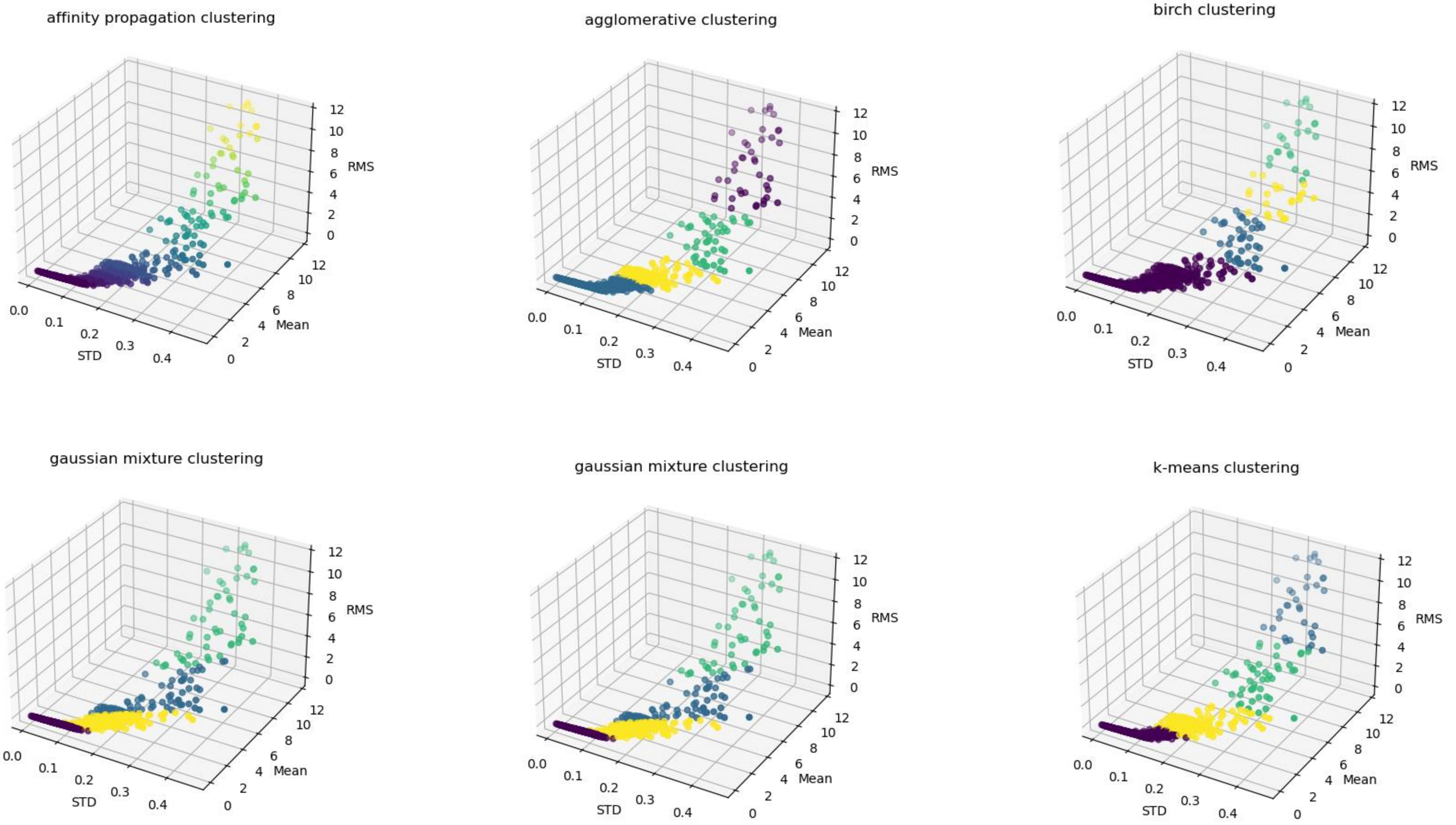
- ✓ gaussian mixture clustering (Gaussian mixture models (GMMs) are **often used for data clustering**. You can use GMMs to perform either hard clustering or soft clustering on query data. To perform hard clustering, the GMM assigns query data points to the multivariate normal components that maximize the component posterior probability, given the data.)

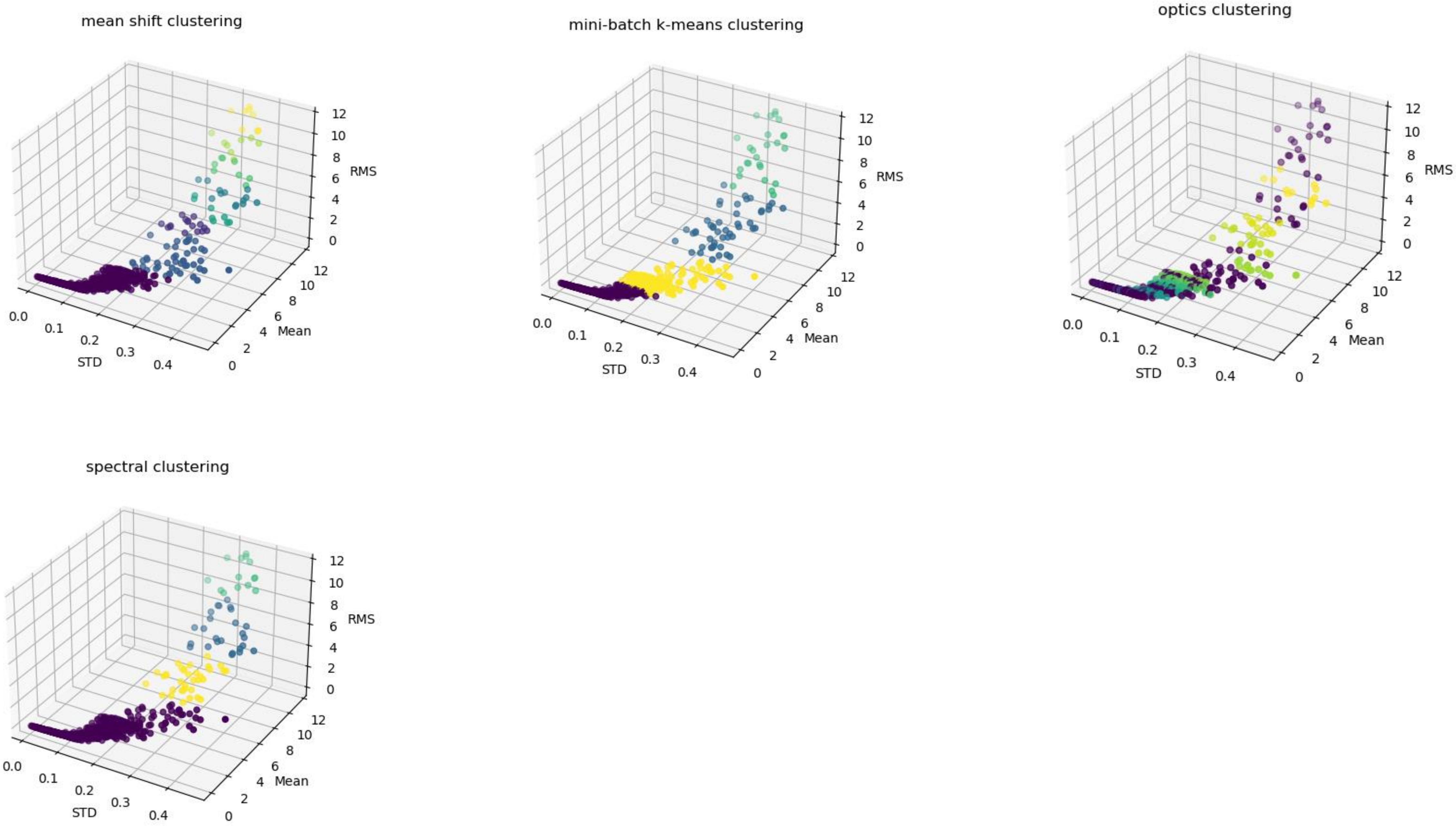
https://en.wikipedia.org/wiki/EM_algorithm_and_GMM_model

https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html

Clustering K=4

Result One	Result Two
✓ spectral clustering for k: 4 Score=**79.345 %**	❖ spectral clustering for k: 4 Score=**77.832 %**
✓ birch clustering for k: 4 Score=**70.532 %**	❖ birch clustering for k: 4 Score=**71.178 %**
✓ agglomerative clustering for k:4 **Score=71.3%**	❖ agglomerative clustering for k: 4 Score=**71.3%**
✓ affinity propagation clustering Score=**67.867 %**	❖ affinity propagation clustering Score=**68.480 %**
✓ DB scan clustering Score=-**10.525 %**	❖ DB scan clustering Score=-**7.415 %**
✓ k-means clustering for k: 4 Score=**72.055 %**	❖ k-means clustering for k: 4 Score=**72.684 %**
✓ mini-batch k-means for k: 4 Score=**72.388 %**	❖ mini-batch k-means for k: 4 Score=**70.438 %**
✓ optics clustering Score=-**15.781 %**	❖ optics clustering Score=-**24.821 %**
✓ mean shift clustering Score=**69.290 %**	❖ mean shift clustering Score=**62.534 %**
✓ gaussian mixture clustering Score=**62.302 %**	❖ gaussian mixture clustering Score=**61.972 %**


affinity propagation clustering
agglomerative clustering
birch clustering
gaussian mixture clustering
gaussian mixture clustering
k-means clustering
STD
0.0
0.1
0.2
0.3
0.4
Mean
0
2
4
6
8
10
12
RMS
0
2
4
6
8
10
12



mean shift clustering

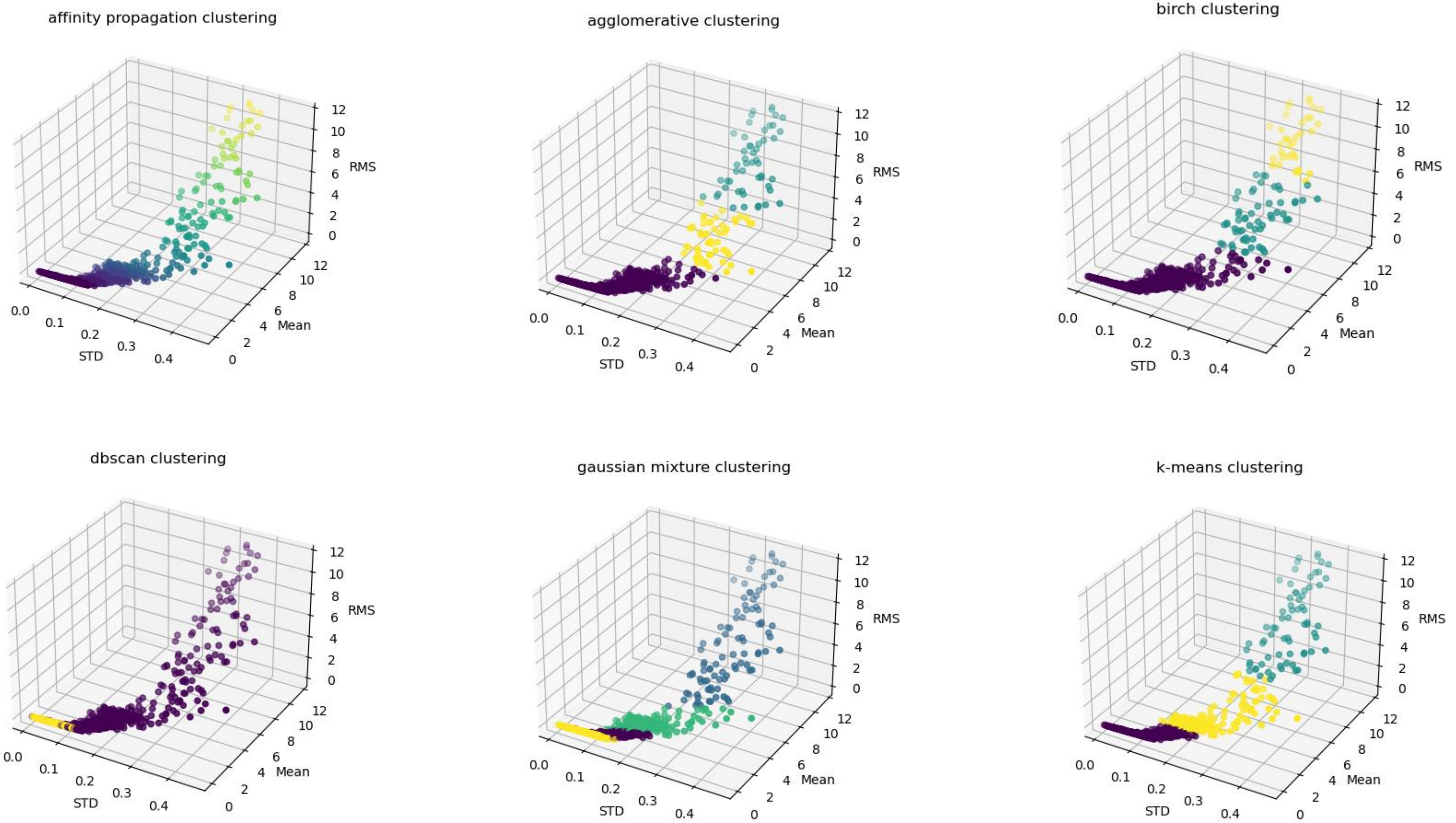
mini-batch k-means clustering

optics clustering

spectral clustering

Clustering K=3

Result One	Result Two
✓ spectral clustering for k: 3 Score=**82.491 %**	❖ spectral clustering for k: 3 Score=**82.778 %**
✓ birch clustering for k: 3 Score=**79.684 %**	❖ birch clustering for k: 3 Score=**78.850 %**
✓ agglomerative clustering for k:3 Score=**77.95%**	❖ agglomerative clustering for k:3 Score=**79.5 %**
✓ affinity propagation clustering Score=**68.153 %**	❖ affinity propagation clustering Score=**64.168 %**
✓ DB scan clustering Score=**26.314 %**	❖ DB scan clustering Score=**19.954 %**
✓ k-means clustering for k: 3 Score=**80.470 %**	❖ k-means clustering for k: 3 Score=**79.331 %**
✓ mini-batch k-means for k: 3 Score=**79.716**	❖ mini-batch k-means for k: 3 Score=**79.428 %**
✓ optics clustering Score=**-17.675 %**	❖ optics clustering Score=-**33.553 %**
✓ mean shift clustering Score=**61.524 %**	❖ mean shift clustering Score=**77.467 %**
✓ gaussian mixture clustering Score=**62.355 %**	❖ gaussian mixture clustering Score=**54.917 %**



affinity propagation clustering

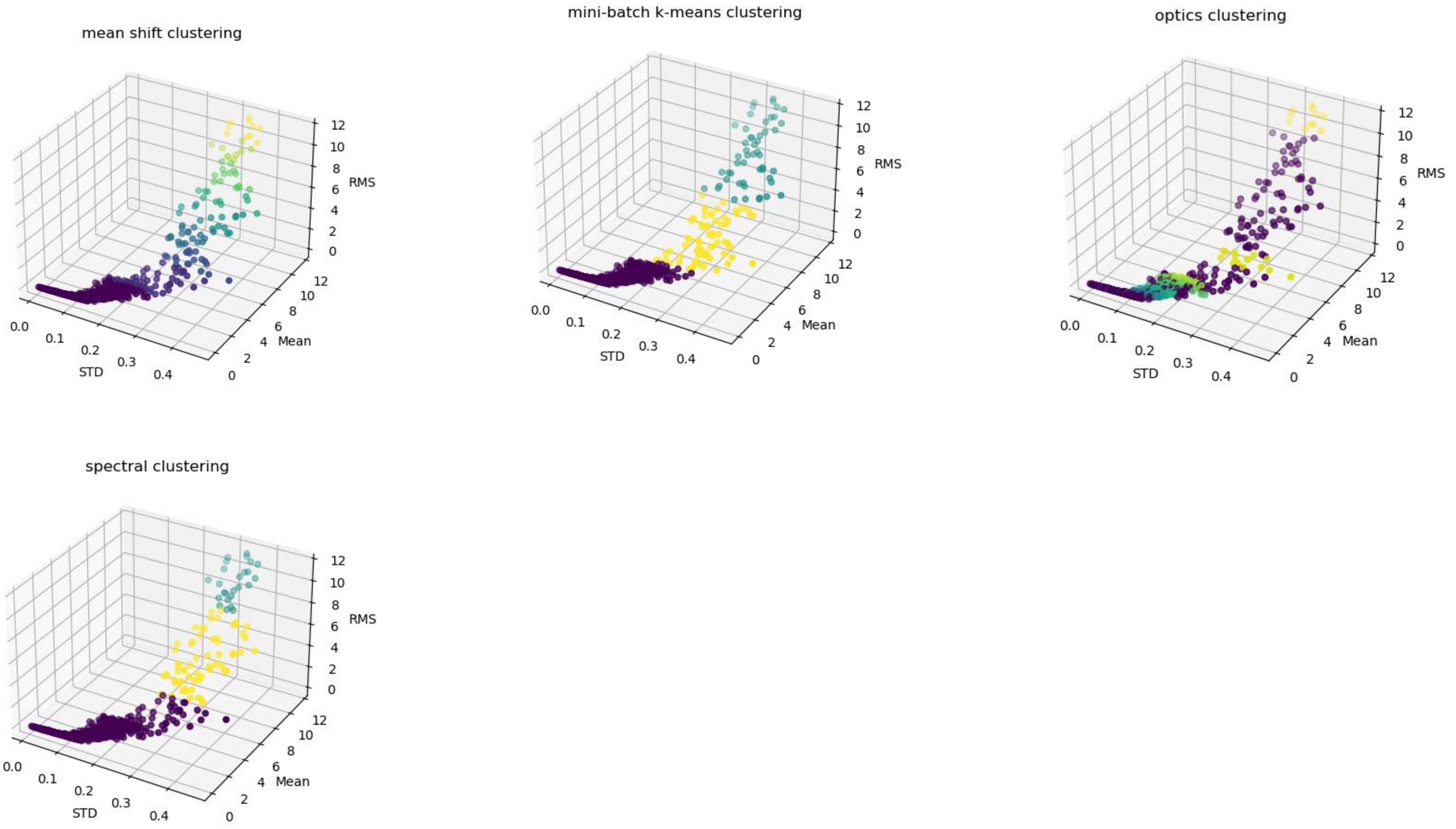
agglomerative clustering

birch clustering

dbscan clustering

gaussian mixture clustering

k-means clustering



mean shift clustering

mini-batch k-means clustering

optics clustering

spectral clustering

PCA feature => Clustering K=3,4

Clustering K=3	Clustering K=4
✓ spectral clustering for k: 3 Score=76.817 %	❖ spectral clustering for k: 4 Score=74.768 %
✓ birch clustering for k: 3 Score=68.093 %	❖ birch clustering for k: 4 Score=46.112 %
✓ agglomerative clustering for k:3 Score=73.2%	❖ agglomerative clustering for k:4 Score=52.6%
✓ affinity propagation clustering Score=28.841 %	❖ affinity propagation clustering Score=28.468 %
✓ DB scan clustering Score=-38.831 %	❖ DB scan clustering Score=-41.601 %
✓ k-means clustering for k: 3 Score=70.515 %	❖ k-means clustering for k: 4 Score=54.547 %
✓ mini-batch k-means for k: 3 Score=58.023 %	❖ mini-batch k-means for k: 4 Score=54.429 %
✓ optics clustering Score=-47.884 %	❖ optics clustering Score=-41.394 %
✓ mean shift clustering Score=61.090 %	❖ mean shift clustering Score=70.571 %
✓ gaussian mixture clustering Score=42.772 %	❖ gaussian mixture clustering Score=37.334 %