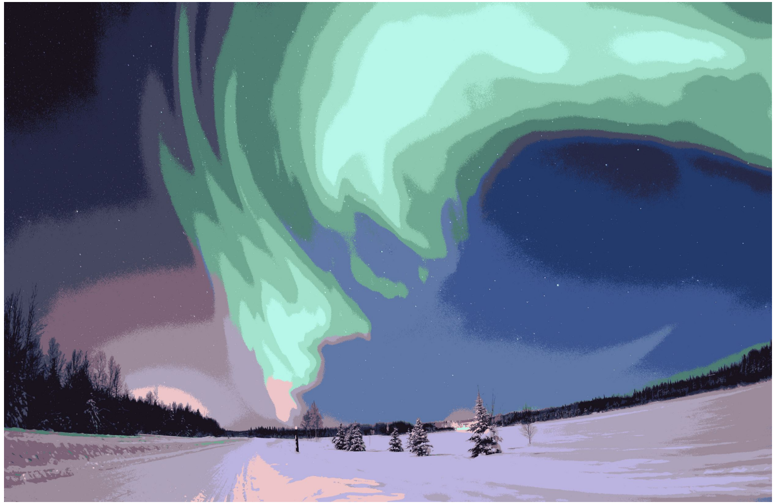
Clustering Algorithm

Presented by

Xin Yang, Zhuohang Li,
Song Yang, Yi Wu

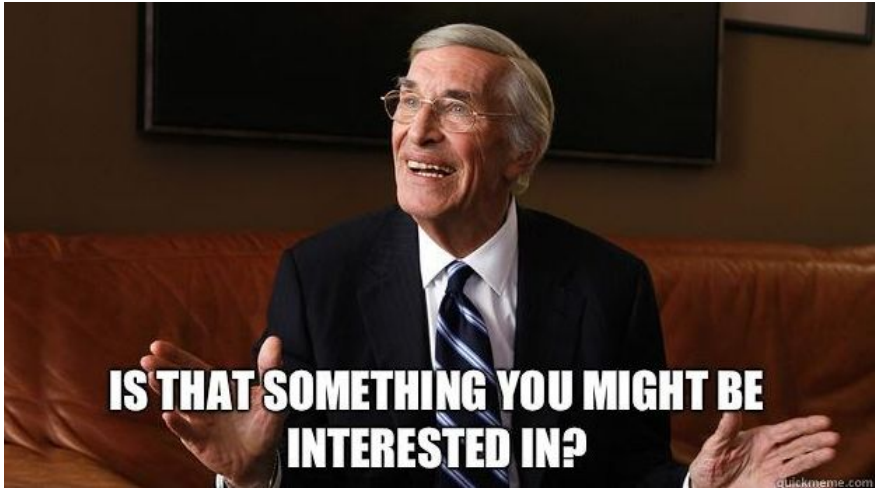
Motivation

- Widely used in data mining and machine learning area
- Image segmentation: machine vision; facial, fingerprint recognition



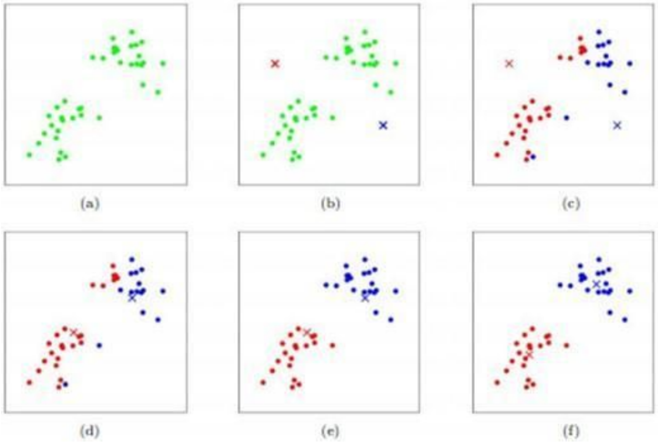
Motivation

Recommendation System:

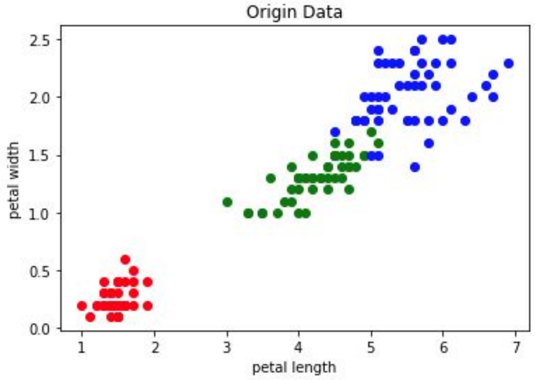


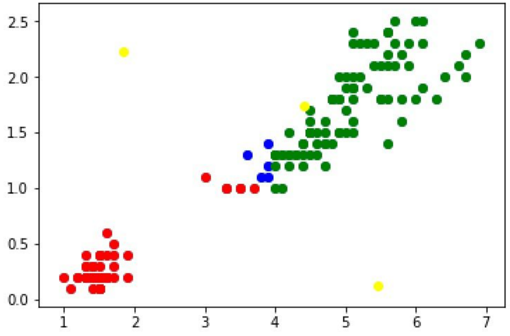
K-means

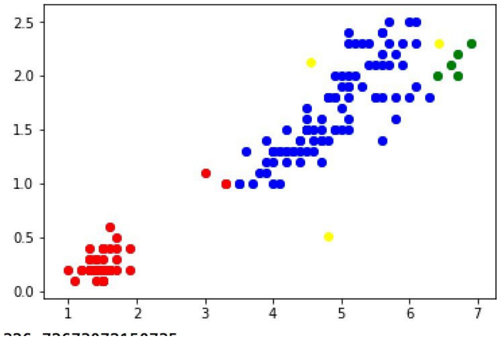
- Description:
  - Divide a group of samples into K different categories.
  - K is artificially given.
  - Time complexity is O(m*n*d*k)
- Algorithm
  - Choose k arbitrary points as centroids.
  - For each sample i, assign it to it's nearest centroids.
  - For each category j, recalculate the centroids.
  - Reassign every data point to its nearest centroid.
  - Iterate from step 2 until no point is reassigned in step 4.

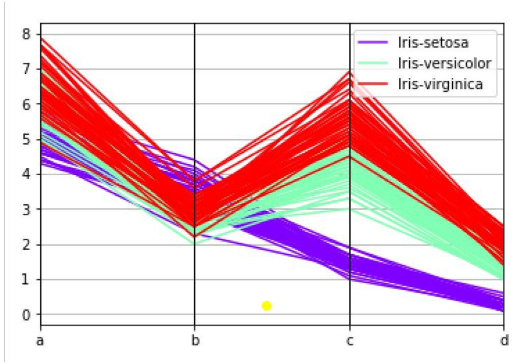


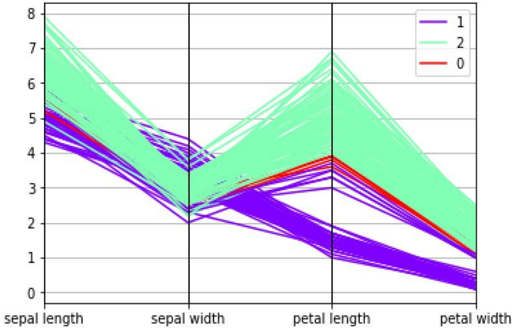
K-means Result

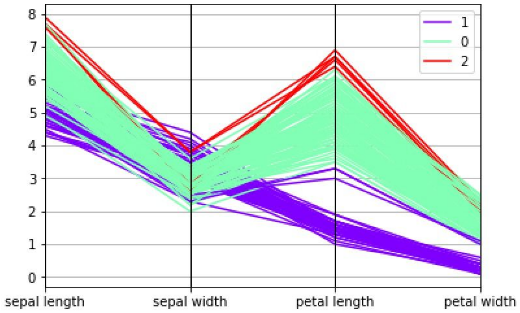






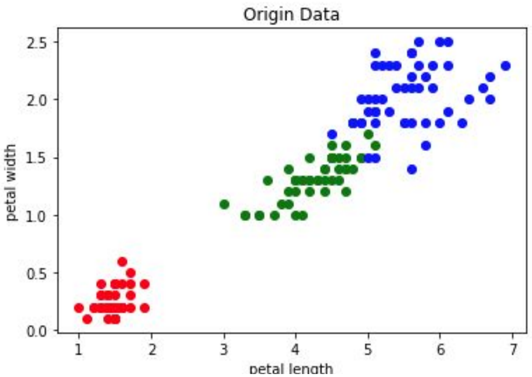


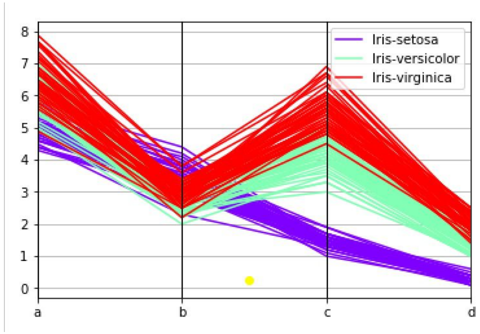


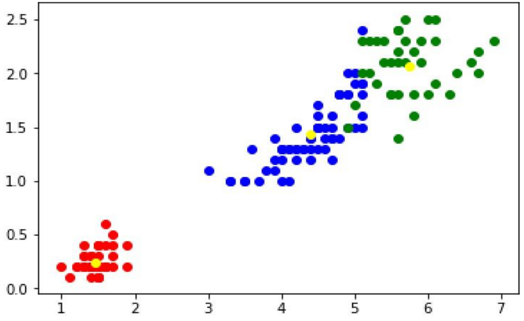


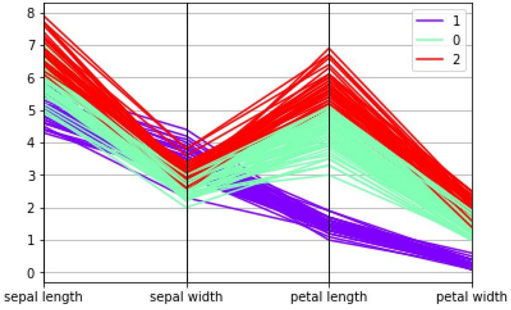
K-means++

- The main idea is to make the original centroids spread from each other.
- After determine the original centroids, implement the original K-means algorithm.

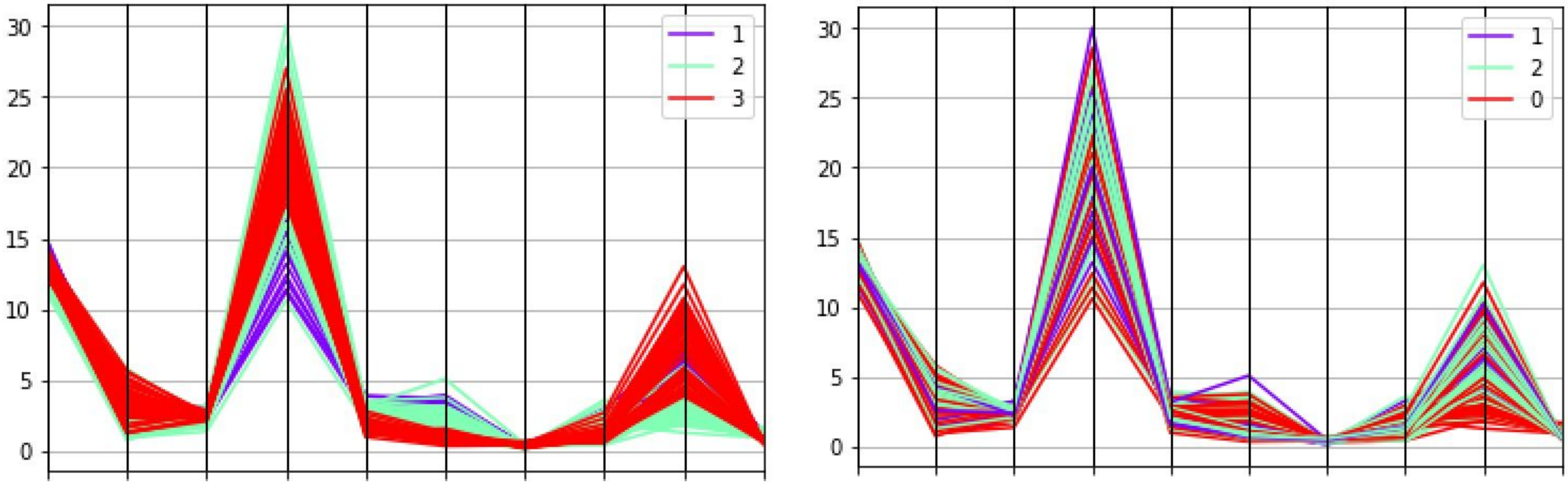






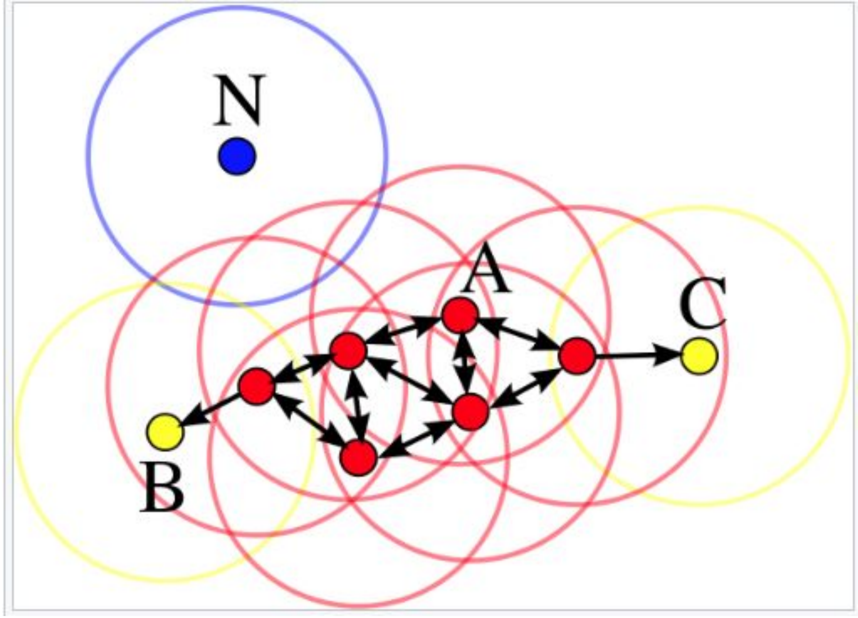


High-dimension Data



DBSCAN

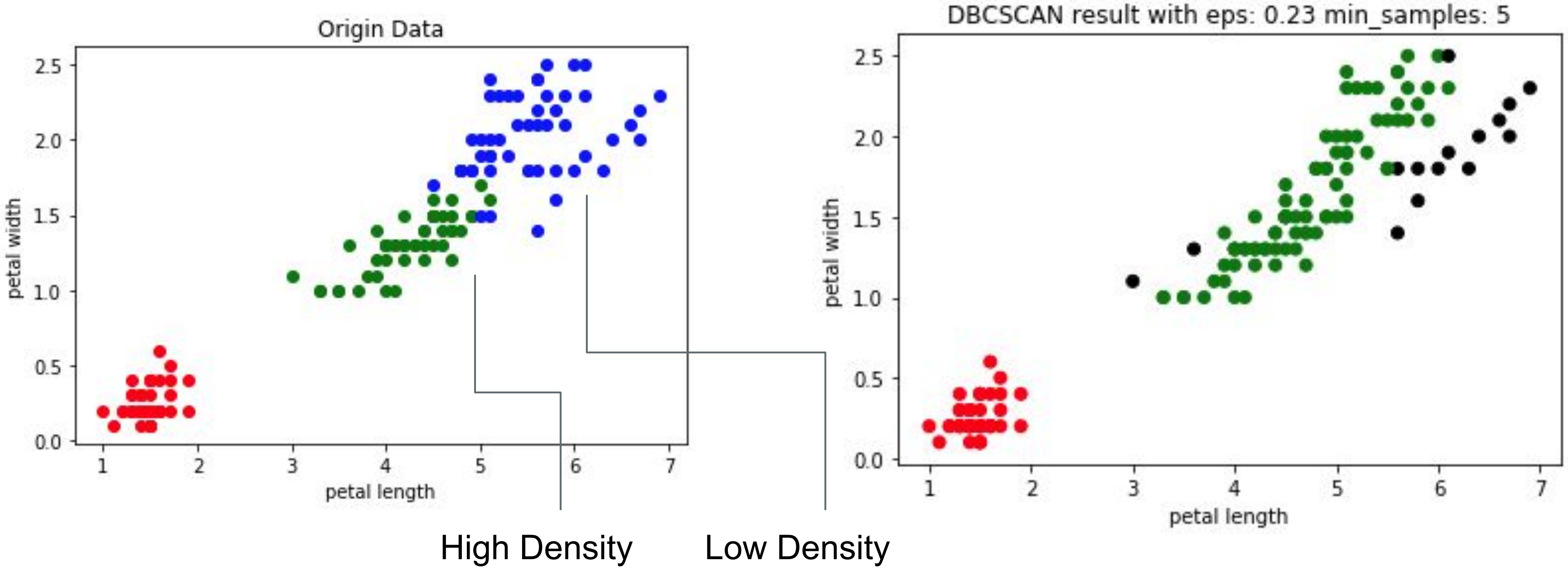
- A density-based algorithm
- Density = (Radius - R, minPts - P) which defines Neighbourhood
- Noise sensitive
- Return Stable Result
- Return an Uncertain Number of Clusters



minPts = 4

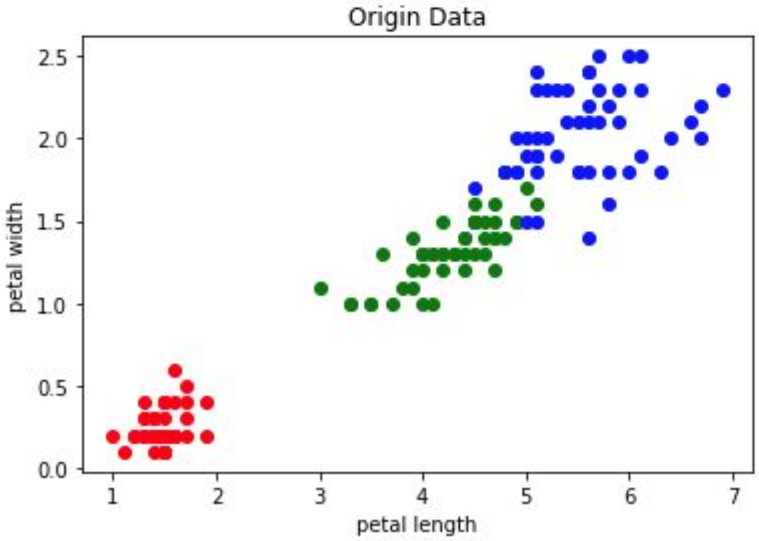
DBSCAN - Distribution Conjecture

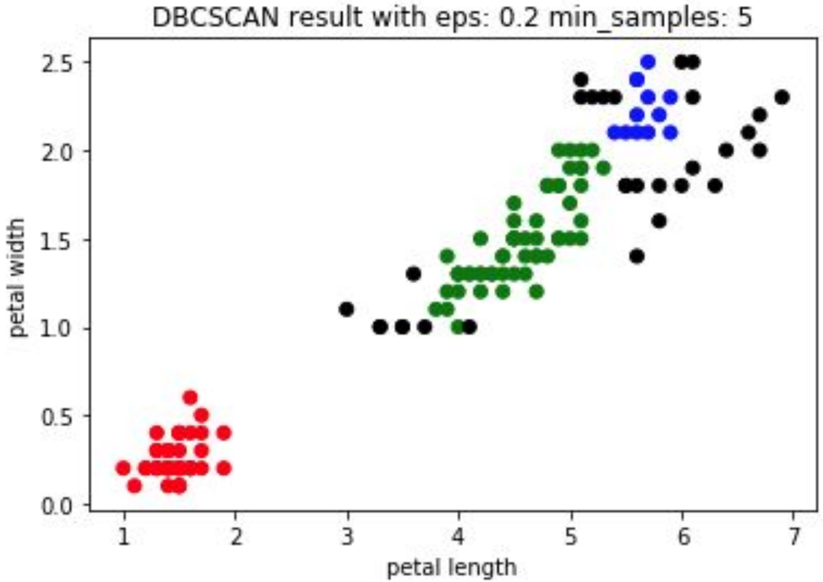
Performance when meeting a set of uneven density data



DBSCAN - Distribution Conjecture

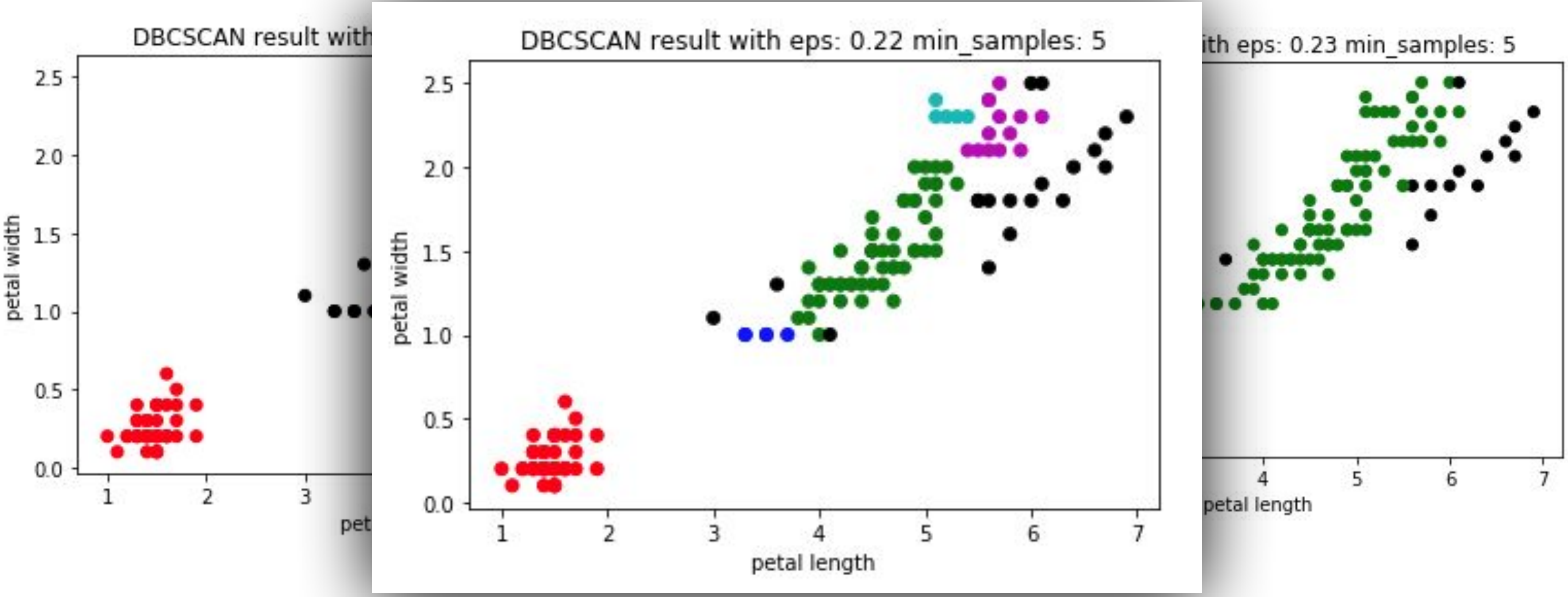
Performance when meeting a set of uneven density data





DBSCAN - Distribution Conjecture

Result



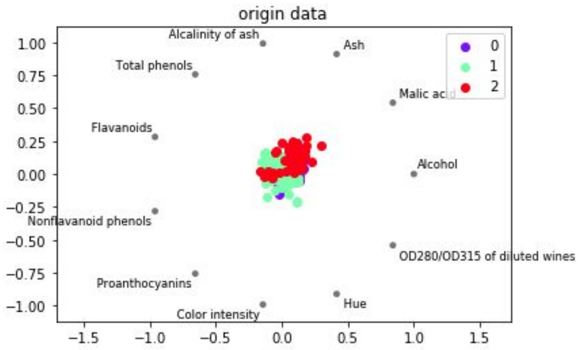
DBSCAN - Curse of Dimensionality

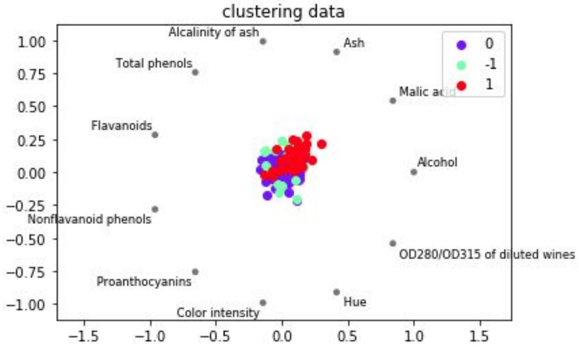
- Using high dimension data
- Euclidean distance
- Neighbourhood Sample(density)
- The Consequence is:
  - Model Overfitting or Do not work

Euclidean distance in Cartesian coordinates

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$





DBSCAN - Curse of Dimensionality

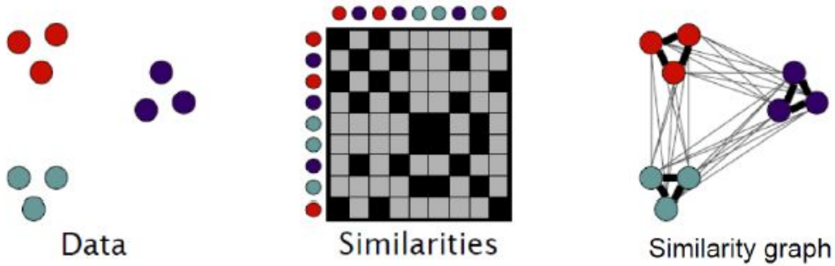
- The Consequence is:
  - Model Overfitting or Do not work

Solution: Dimensionality reduction

Spectral Clustering

- Algorithm Description:
  - Based on graph theory
  - Similarity matrix: mapping similarity into similarity coordinate system
  - Dimension deduction: Laplacian matrix
  - Run standard clustering on relevant eigenvectors of Laplacian matrix
  - Segmentation using Ncut

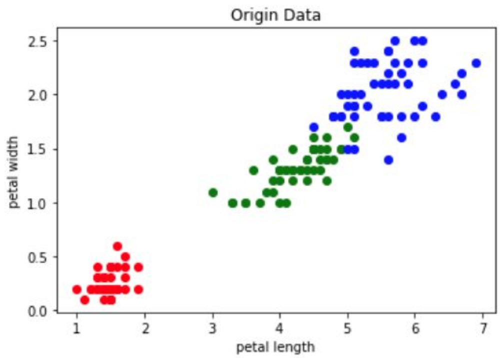
Clustering -> Graph Cutting



Spectral Clustering

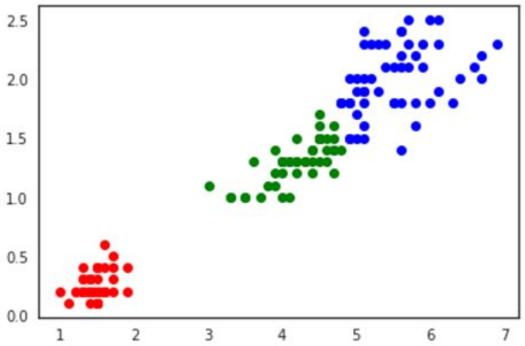
- Implementation:
  - Similarity matrix(Kernel Function):
    - Nearest Neighbors (default k=10)
    - Polynomial (default γ=1, d=3 ,r=1)
    - Radial Basis Function (default γ=1)
  - Test cases:
    - 2 Dimension
    - 4 Dimension
    - 13 Dimension

Two Dimensional Data

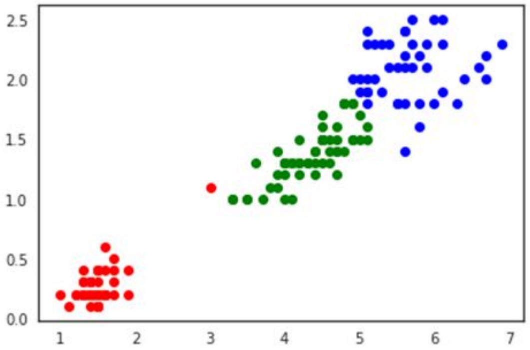


Indicators	KNN 2D	POLY 2D $\gamma = 0.18$	RBF 2D
Adjusted Rand Index	0.8857	0.8498	0.8856
Mutual Information Scores	0.8680	0.8401	0.8622
V-measure	0.8705	0.8443	0.8641
Calinski-Harabaz Index	1192.7951	1160.5767	1215.9292

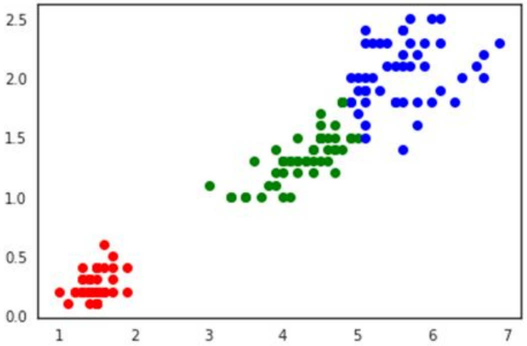
Table 4: Evaluation Indicators of Spectral Clustering 2D



knn

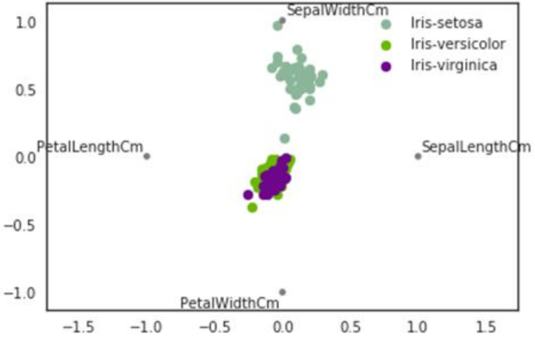


poly

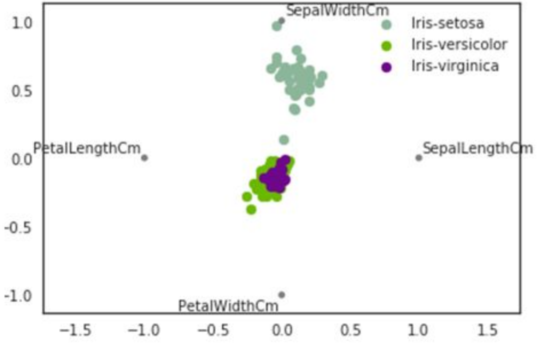


rbf

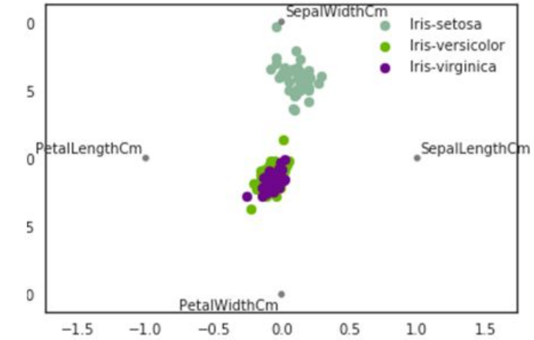
Four Dimensional Data



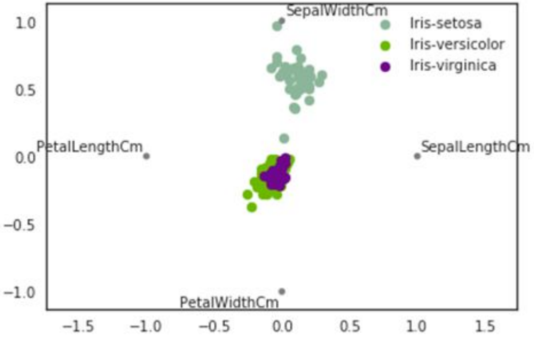
original



knn



poly



rbf

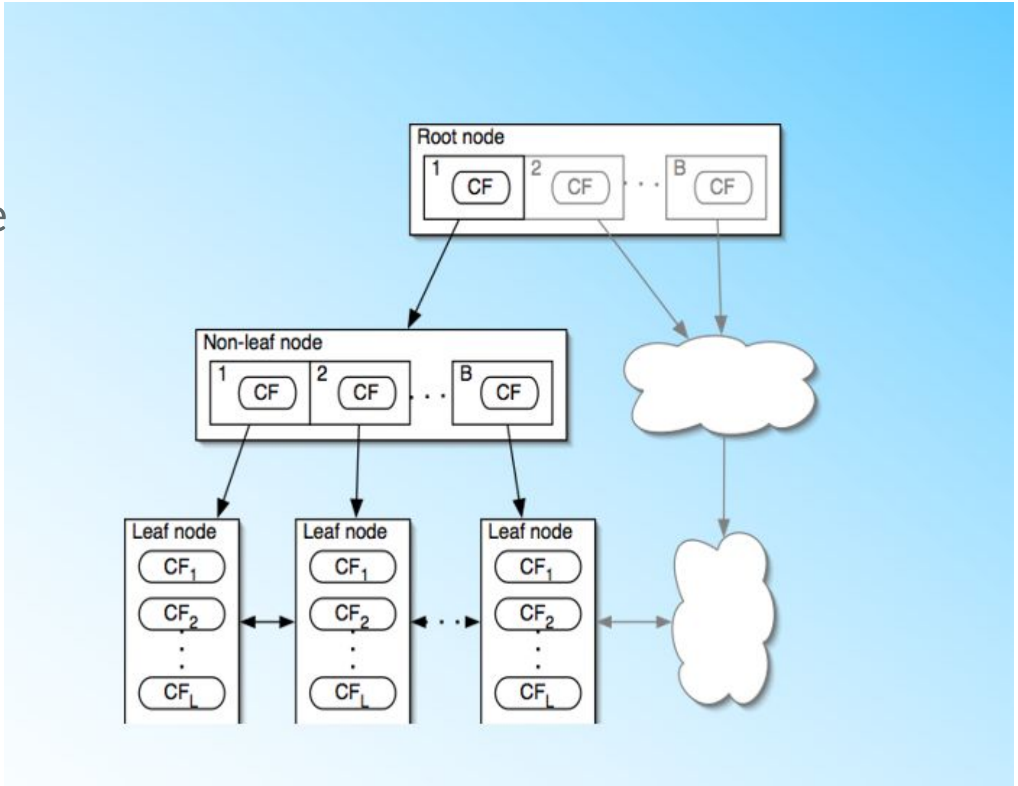
Indicators	KNN	POLY	RBF γ=0.4
ARI	0.7951	0.8838	0.7302
MIS	0.7934	0.8484	0.7483
Vmeasure	0.8056	0.8503	0.7581
CHI	555.6662	436.2717	560.3999

Thirteen Dimensional Data

Indicators	K-means	DBSCAN	KNN k=10	POLY γ=0.0000955 d=2	RBF γ=0.00005
Adjusted Rand Index	0.1184	0.2927	0.3590	0.3747	0.3308
Mutual Information Scores	0.1001	0.3730	0.4132	0.4374	0.3730
V-measure	0.1139	0.3917	0.4199	0.4455	0.4029
Calinski-Harabaz Index	241.1358	239.2542	533.8577	529.7771	491.5899

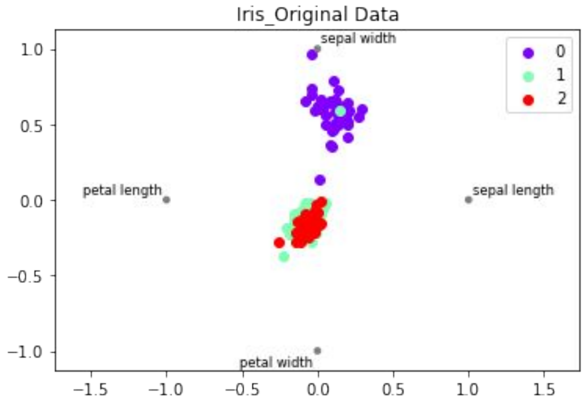
BIRCH(Balanced Iterative Reducing and Clustering Using Hierarchies)

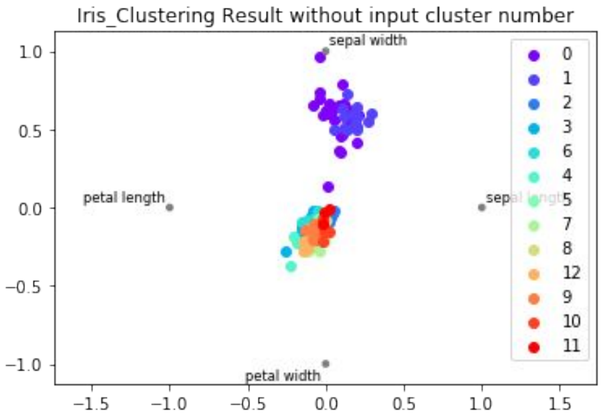
- A hierarchical clustering algorithm
- Use CF(clustering feature) tree to implement multilayer clustering
- CF is a statistic summary of a subcluster CF=(N, LS, SS)

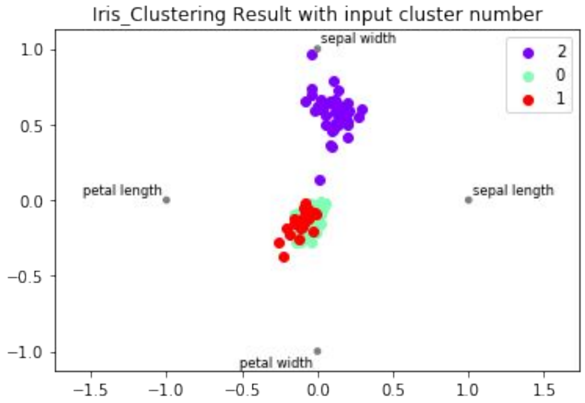


Number of clusters

- Optional
- If not given, equals to the number of nodes in CF tree
- If given, merge subclusters to fit

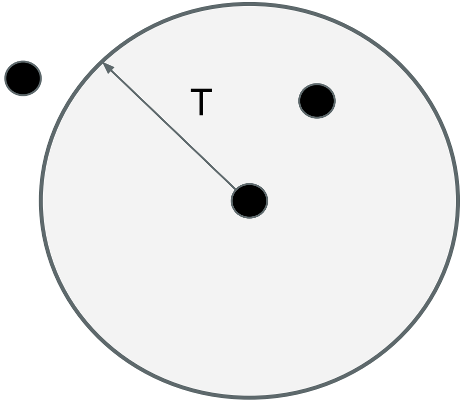






Threshold

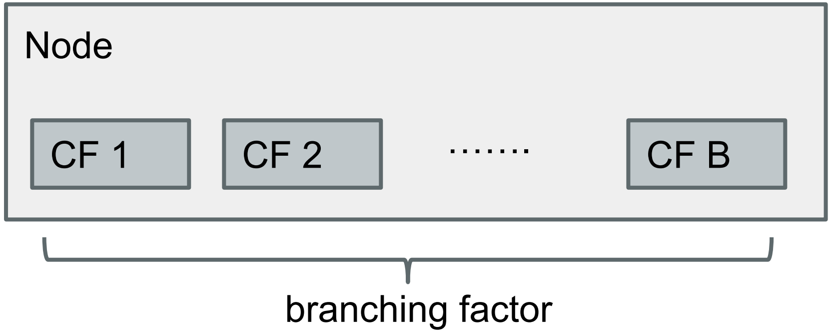
the threshold value of the radius of the leaf nodes, i.e. the maximum radius for all hypersphere formed by data points.



threshold	**0.1**	**0.3**	**0.5**
Adjusted Rand Index	0.6517	0.5436	0.5823
Mutual Information Scores	0.6764	0.5984	0.6237
V-measure	0.6943	0.6698	0.6566
Calinski-Harabaz Index	503.1305	399.9507	457.5418

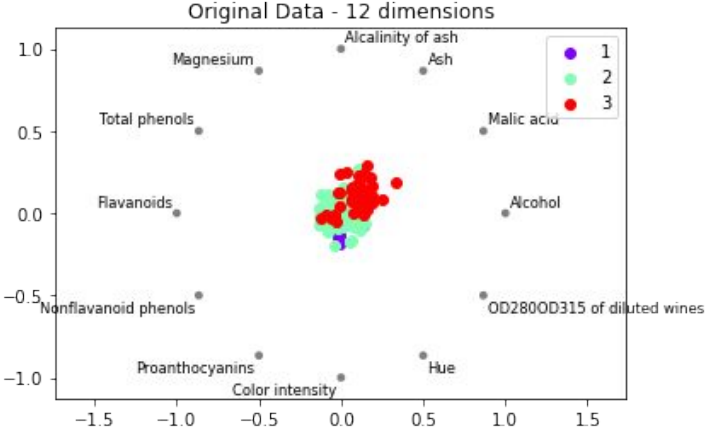
Branching factor

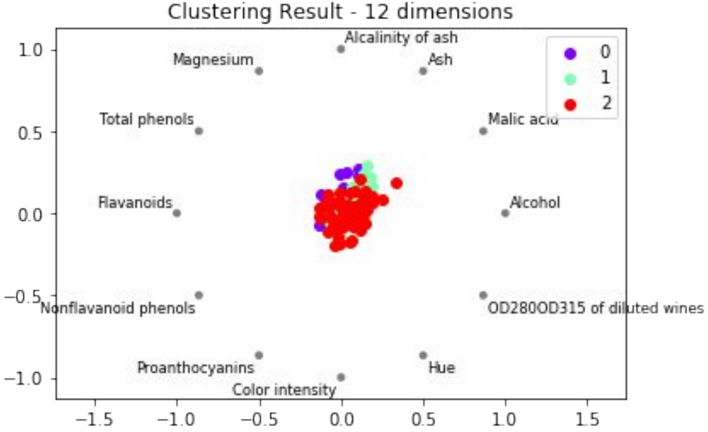
the maximum number of clustering features in each node



branching factor	10	30	50
Adjusted Rand Index	0.7122	0.6517	0.6517
Mutual Information Scores	0.7470	0.6764	0.6764
V-measure	0.7606	0.6943	0.6943
Calinski-Harabaz Index	554.9067	503.1305	503.1305

High Dimensional data





dimension	**4**	**8**	**12**
Adjusted Rand Index	0.1219	0.1189	0.1090
Mutual Information Scores	0.1429	0.1264	0.1090
V-measure	0.1572	0.1369	0.1238
Calinski-Harabaz Index	186.6881	244.9339	230.3928

Thanks for listening !!!!