

DATASCI 207

Nedelina Teneva, PhD
nteneva@berkeley.edu

School of Information, UC Berkeley

Announcements

- Finalize datasets by end of this week: enter dataset info in the [Logistics Sheet](#)
- No live session on July 4th
- No homework 7!

8	6/27/2023	8: Unsupervised Learning: k-Means and PCA	-Daume Chap 15 - Gene Expression Clustering - Document Clustering -ESLII, Chap 13.2, 14.3, 14. 5	W8 Live Session Material No HW7 :) Decide on a project dataset
9	7/4/2023	9: Embeddings For Text	Eigenfaces Paper	No class on July 4th HW8 due...]
10	7/11/2023	10: Convolutional Neural Networks		HW9 due...
11	7/18/2023	11: Network Architecture Design		HW10 due...
12	7/25/2023	12: Fairness		Project work
13	8/1/2023	13: Advanced Topics		Project work
14	8/8/2023	-		Project work

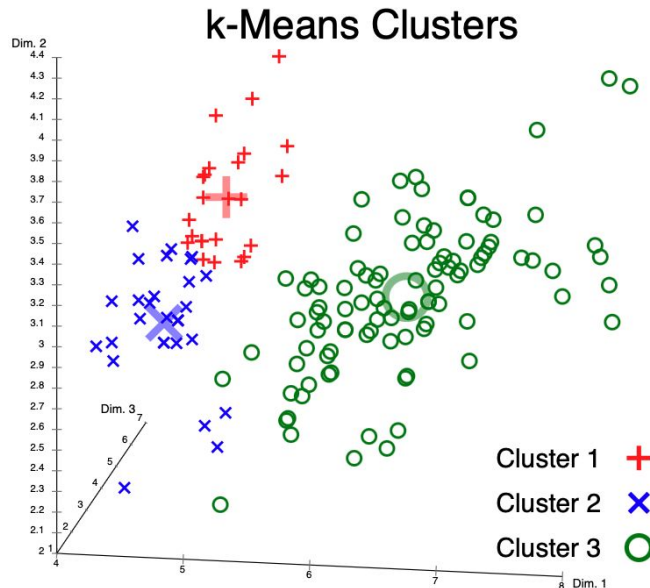
Unsupervised Methods

- What is unsupervised learning?
- Discuss two groups of unsupervised techniques
 - Clustering & Dimensionality reduction
 - GMMs

Unsupervised Learning

Recap

- What is unsupervised learning?
 - Learning without a teacher/labels
 - Typically based on distances between data points
- Examples?
 - Genre classification (given a collection of music tracks predict their genres)
 - <https://dc.uwm.edu/cgi/viewcontent.cgi?article=3844&context=etd>
 - Gene identification (given a collection of microarray data, identify functionally related genes)
 - <https://www.gene-quantification.de/haeseleer-bioinf-2005.pdf>
- How is “learning” performed in unsupervised learning?
- How is model performance measured?

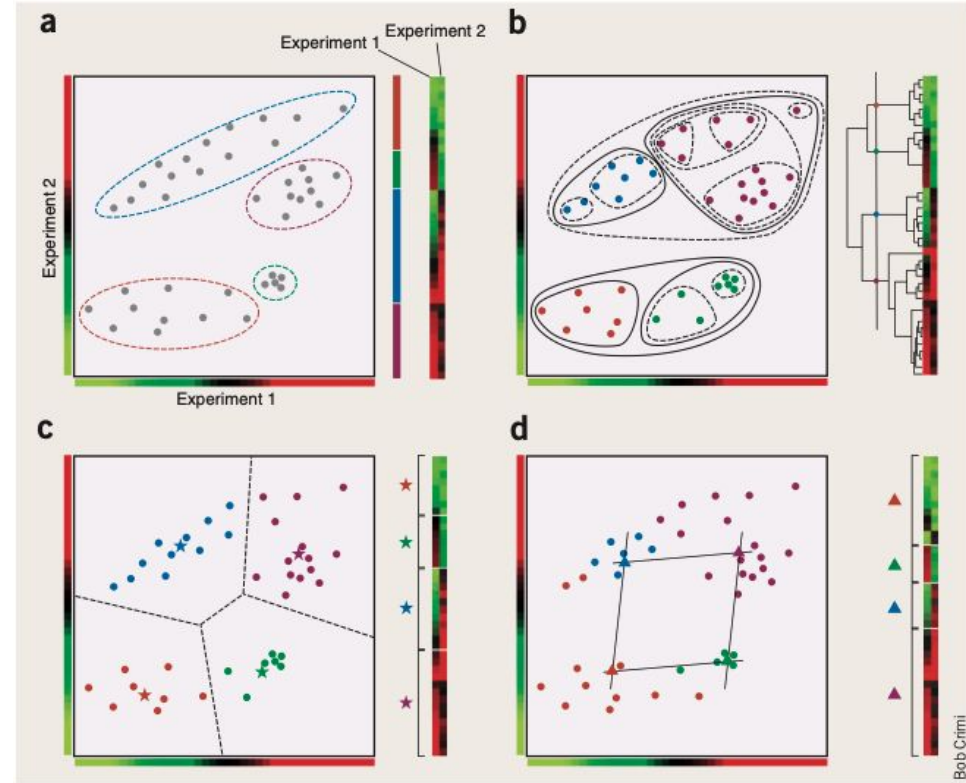


Clustering

Clustering

Idea: Grouping data points together

- Centroid-based
 - Based on the use of archetypes (e.g. K-means)
- Hierarchical clustering: constructs a hierarchy of clusters
 - Agglomerative (bottom up)
 - Divisive (top down)
- Spectral clustering
 - Relies on the spectrum (eigenvalues) of a similarity matrix of the data to reduce the data dimensions before clustering.



K-means

See Daume [Chap 15](#)

Algorithm 35 K-MEANS(D, K)

```

1: for  $k = 1$  to  $K$  do
2:    $\mu_k \leftarrow$  some random location // randomly initialize mean for  $k$ th cluster
3: end for
4: repeat
5:   for  $n = 1$  to  $N$  do
6:      $z_n \leftarrow \operatorname{argmin}_k ||\mu_k - x_n||$  // assign example  $n$  to closest center
7:   end for
8:   for  $k = 1$  to  $K$  do
9:      $\mu_k \leftarrow \operatorname{MEAN}(\{x_n : z_n = k\})$  // re-estimate mean of cluster  $k$ 
10:   end for
11: until converged
12: return  $z$  // return cluster assignments

```


K-means

Objective: *sum of squared distances from any data point to its assigned center.*

1. does it converge (and if so, how quickly);
2. how sensitive it is to initialization?

K-means

Objective: *sum of squared distances from any data point to its assigned center.*

1. does it converge (and if so, how quickly);
2. how sensitive it is to initialization?

Answers:

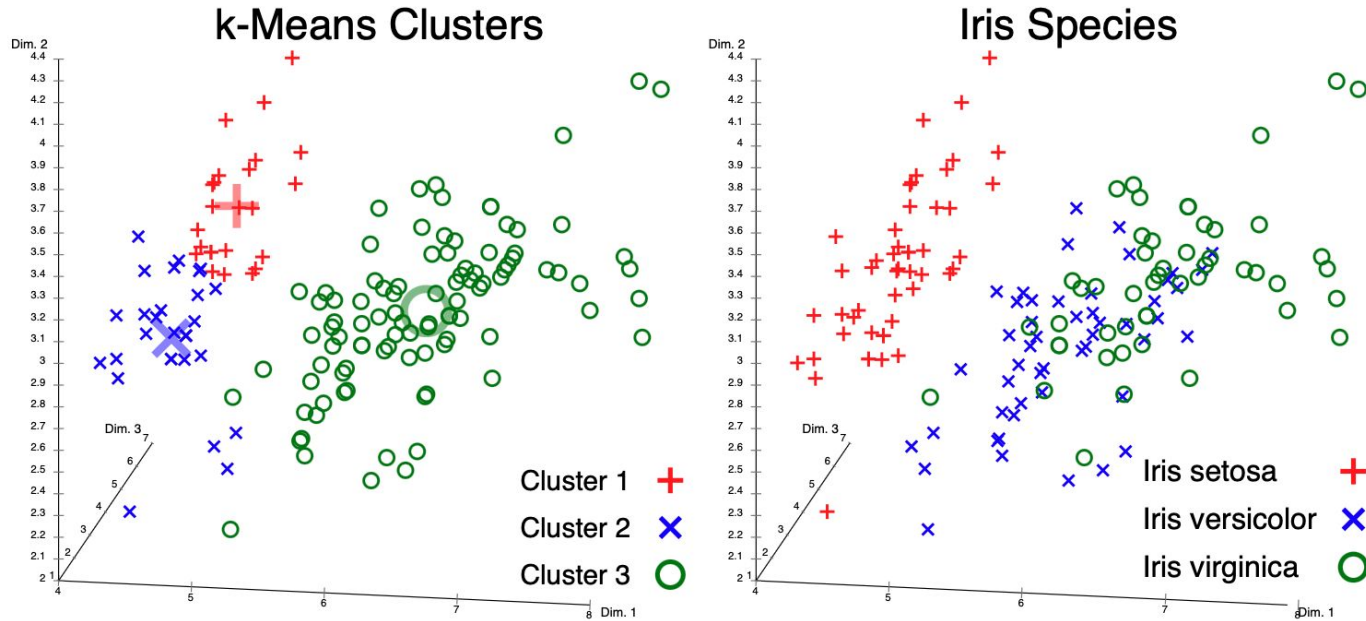
(1) yes it converges, and it converges very quickly in practice (though slowly in theory)

- KNN converges only to local minima.

(2) yes it is sensitive to initialization, but there are good ways to initialize it. e.g. you want means to be as far from each other as possible, see KNN++ in Figure 36 in Daume, Chap 15)

K-means

https://en.wikipedia.org/wiki/K-means_clustering#/media/File:Iris_Flowers_Clustering_kMeans.svg

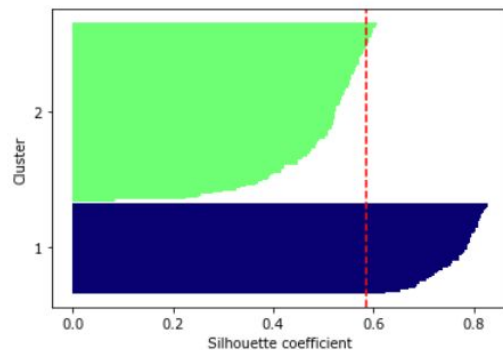
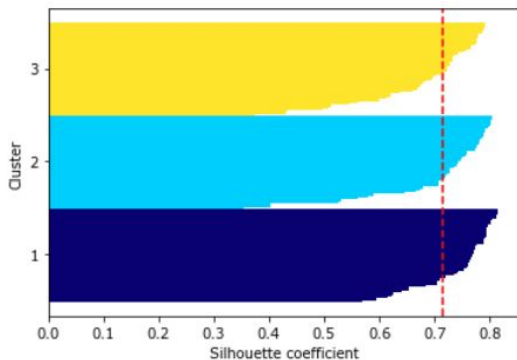


Best K in K-means?

Best K?

To calculate the silhouette coefficient of a single example, apply 3 steps:

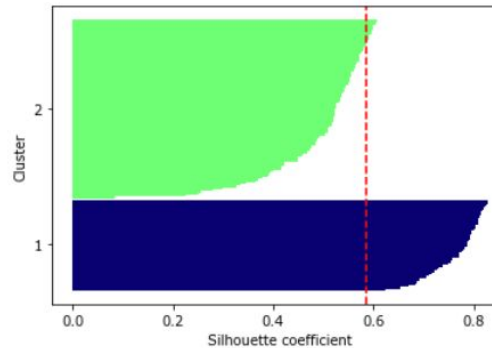
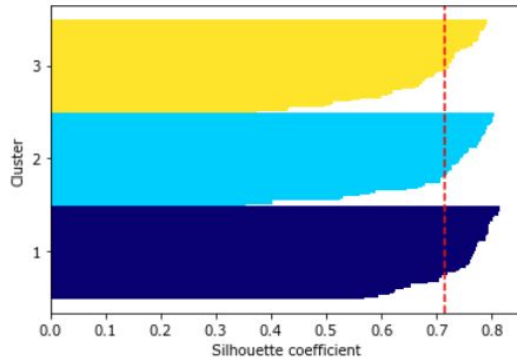
1. calculate **cluster cohesion**, a_i = average distance between an example x_i and all other points in the same cluster.
2. calculate **cluster separation** b_i from the next closest cluster = average distance between an example x_i and all examples in the nearest cluster.
3. calculate **the silhouette**, $s_i = (b_i - a_i) / \max(b_i, a_i)$



Best K?

To calculate the silhouette coefficient of a single example, apply 3 steps:


1. calculate **cluster cohesion**, a_i = average distance between an example x_i and all other points in the same cluster.
2. calculate **cluster separation** b_i from the next closest cluster = average distance between an example x_i and all examples in the nearest cluster.
3. calculate **the silhouette**, $s_i = (b_i - a_i) / \max(b_i, a_i)$



- Ideal values of s_i is 1
- Clustering models with a high silhouette coefficient are said to be **dense**, where samples in the same cluster are similar to each other, and well **separated**, where samples in different clusters are not very similar to each other.


Dimensionality Reduction

SVD

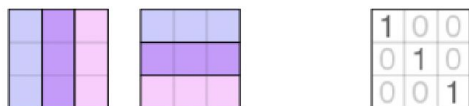


$$\mathbf{M} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^*$$

$m \times n$ $m \times m$ $m \times n$ $n \times n$



$$\mathbf{U} \mathbf{U}^* = \mathbf{I}_m$$



$$\mathbf{V} \mathbf{V}^* = \mathbf{I}_n$$

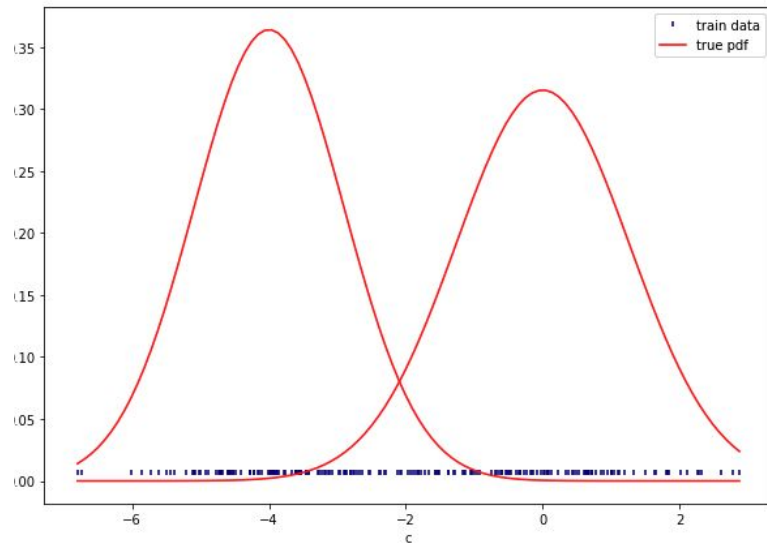
- Project data points on the singular vectors
- Effectively allows us to obtain a data point embeddings
 - n dimensional data points gets mapped to k dimensions ($\mathbb{R}^n \rightarrow \mathbb{R}^k$) where k is the number of the top k singular vectors selected.

Advanced: Numerical Linear Algebra [book](#) by Trefethen and Bao

Gaussian Mixture Models (GMMs)

GMMs

- Generative probabilistic model describing the distribution of the data
- Technically not a clustering method but serves the same purpose.
- Idea:
 - Based on the assumption that all data points come from a mixture of Gaussian distributions with unknown parameters.
 - a model that tries to learn the data generating process (i.e., the true data distribution (mean, variance and scale parameters).
 - learns these parameters by maximizing the maximum likelihood criterion using the Expectation Maximization (EM) algorithm.



Demo: Fitting a GMM model on 1-D synthetic dataset using EM for training

https://github.com/MIDS-W207/nteneva/blob/main/live_sessions_current/week8/GMM_EM.ipynb