

Computational Data Analysis

Machine Learning

Yao Xie, Ph.D.

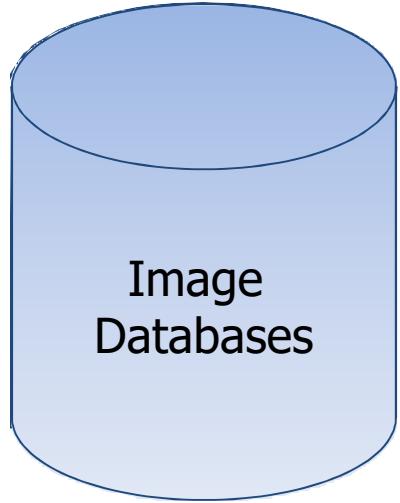
Associate Professor

Harold R. and Mary Anne Nash Early Career Professor
H. Milton Stewart School of Industrial and Systems
Engineering

Clustering

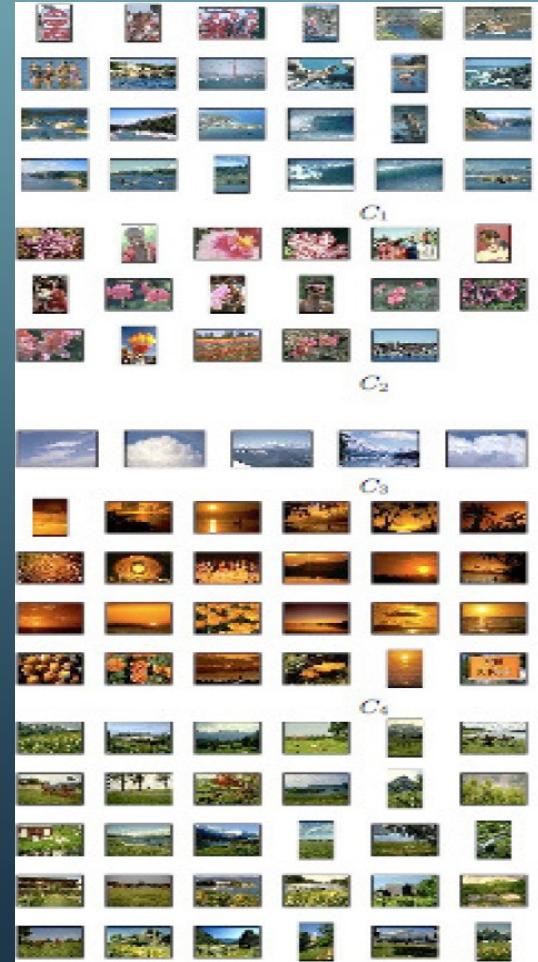


Cluster images



Goal of clustering:

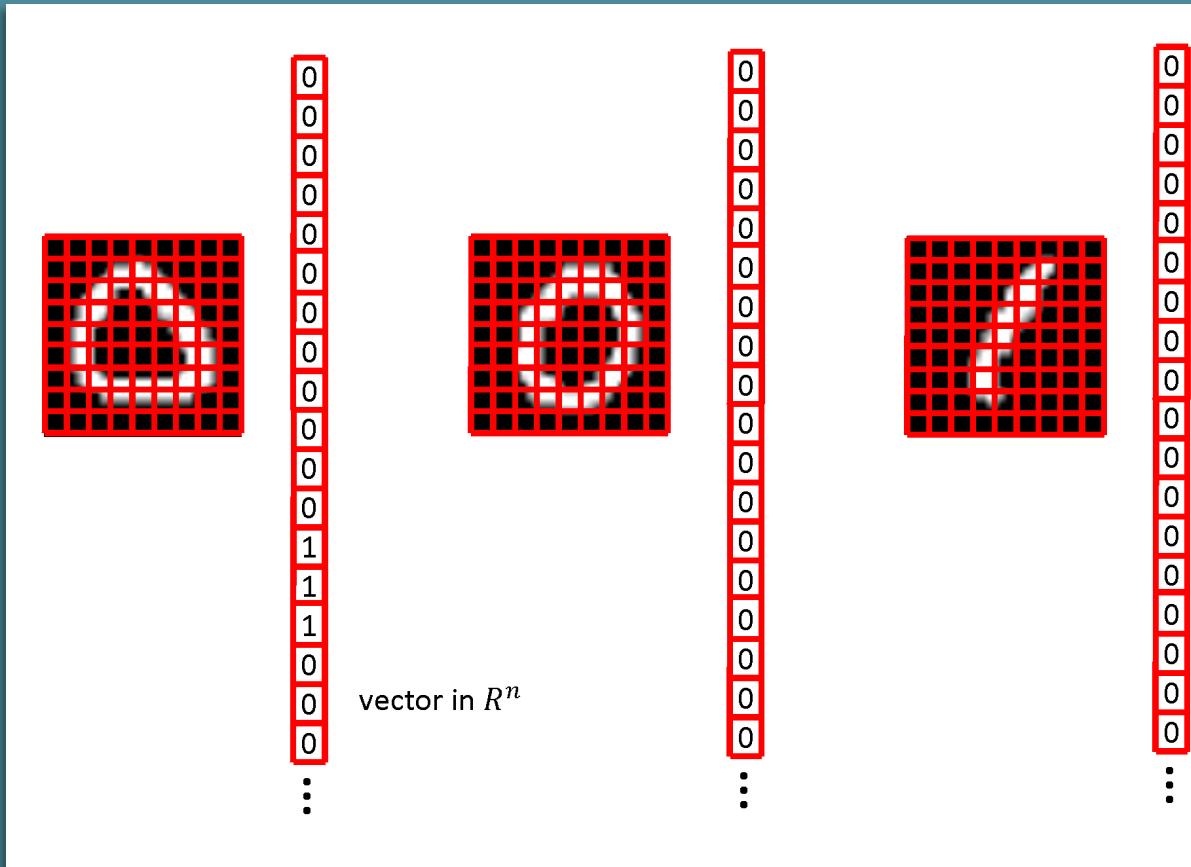
Divide object into groups, and objects within a group are more similar than those outside the group



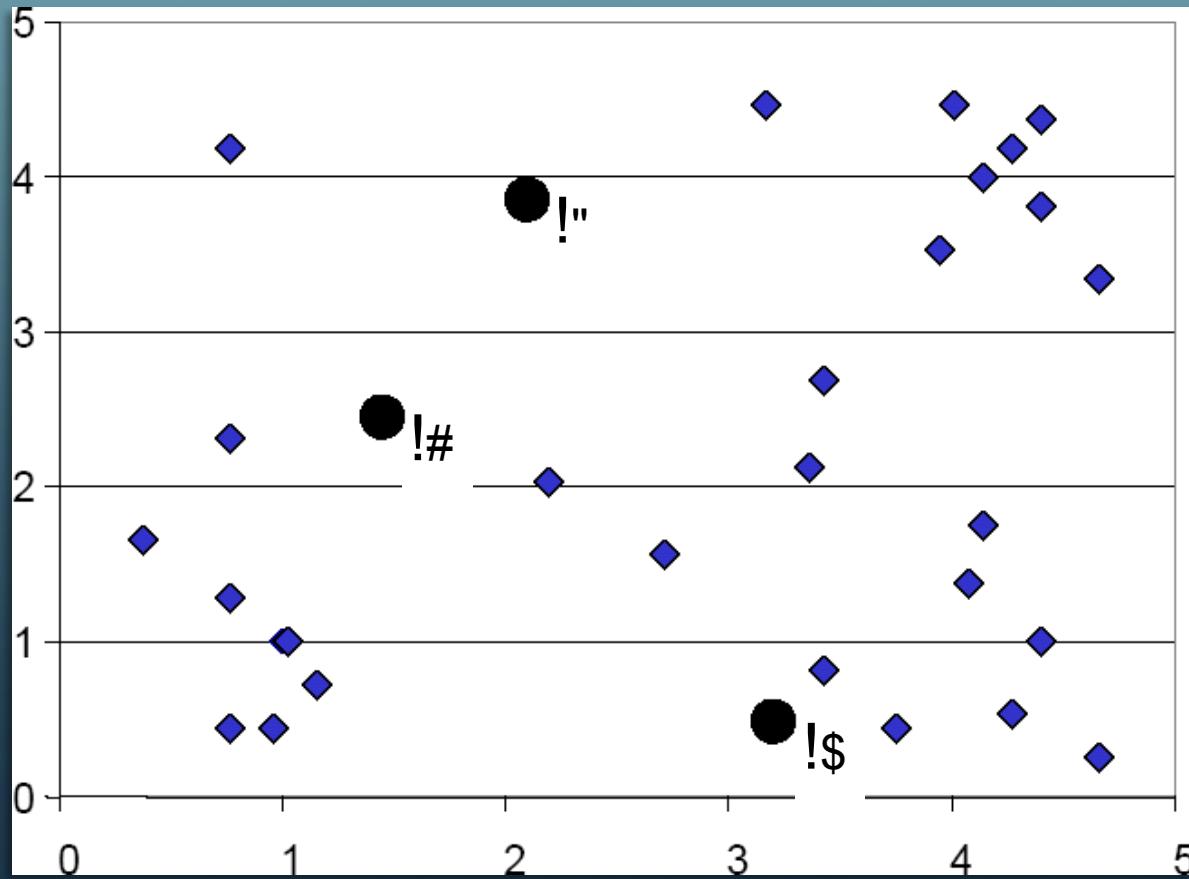
Cluster handwritten digits

72104149590690159784966540740131
34727121174235124463556041957853
74043070291732977627847361369314
17696054992194873974449254767905
85665781016467317182029955156034
465465451447232711818185089250111
09031642361113952945939036557227
12841733887922415987230442419577
28268577918180301994182129759264
15429204002847124027433003196525
12936420711215339786361381051315
56185119462250656372088541140337
61621928619525442838245031773797
19214292049148184598837600302664
85332391268056663882758961841259
19754089910523789406395213136578
22632654897130383193446421825488
40023277087447969098046063548339
33378087170654380963809968685786
02402231975108462479329822927359
18020511376712580371409186774349
19317397691378336128585114431077
07944855408210845040613326726931
46254206217341054311749948402451
16471942415538314568941538032512
83440883317358632613607217142421
79611248177480231310770355276692
83522560829288887493066321322930
05781446029147473988471212232383
91740355863267663279117564951334
78911691445406223151203812671623
90122089

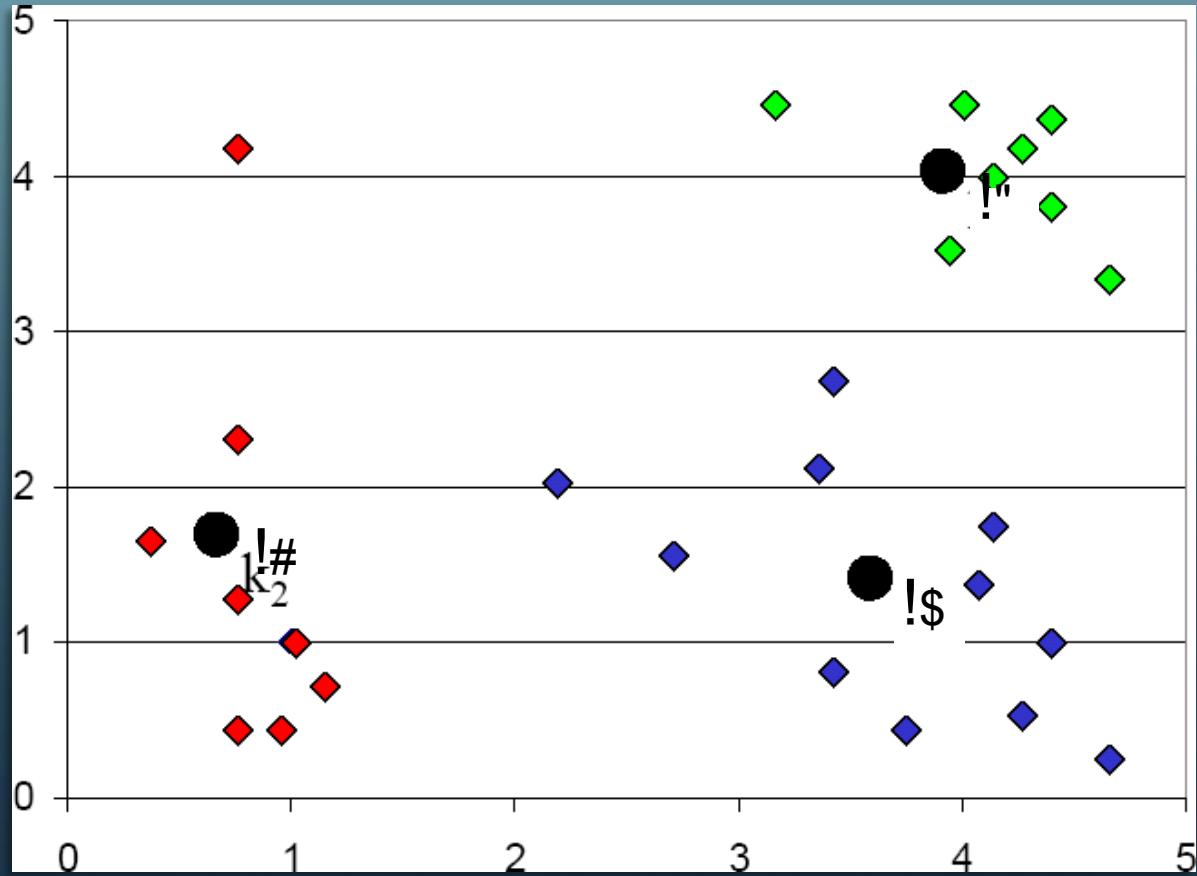
How to represent objects?



Before clustering, a bunch of vectors



After clustering, group assigned



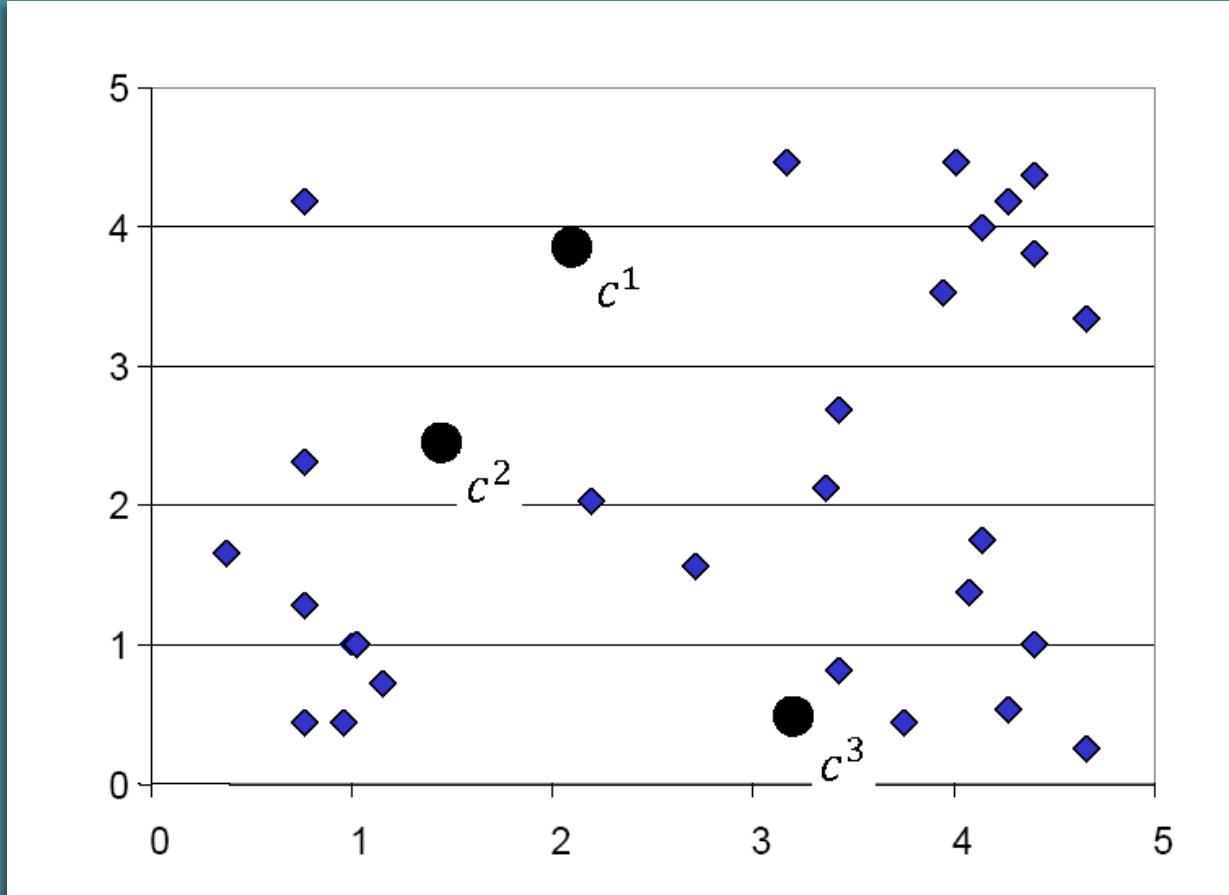
Formal statement and goal of k-means

- Given m data points, $\{x^1, x^2, \dots, x^m\} \in R^n$
- Find k cluster centers, $\{c^1, c^2, \dots, c^k\} \in R^n$
- And assign each data point i to one cluster, $\pi(i) \in \{1, \dots, k\}$

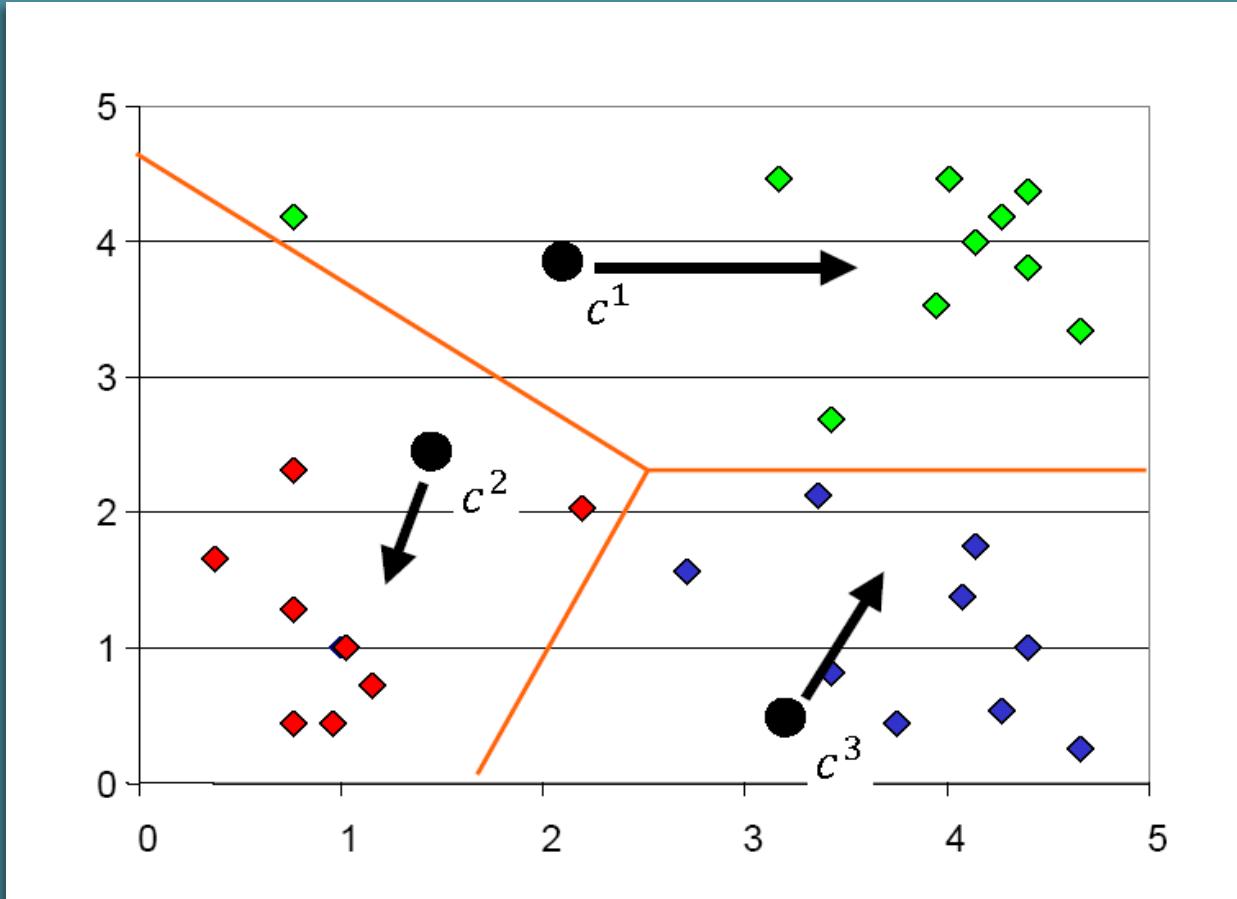
K-means algorithm

- Initialize k cluster centers, $\{c^1, c^2, \dots, c^k\}$, randomly
- Do
 - Decide the cluster memberships of each data point, x^i , by assigning it to the nearest cluster center (**cluster assignment**)
$$\pi(i) = \operatorname{argmin}_{j=1,\dots,k} \|x^i - c^j\|^2$$
 - Adjust the cluster centers (**center adjustment**)
$$c^j = \frac{1}{|\{i: \pi(i) = j\}|} \sum_{i: \pi(i)=j} x^i$$
- While any cluster center has been changed

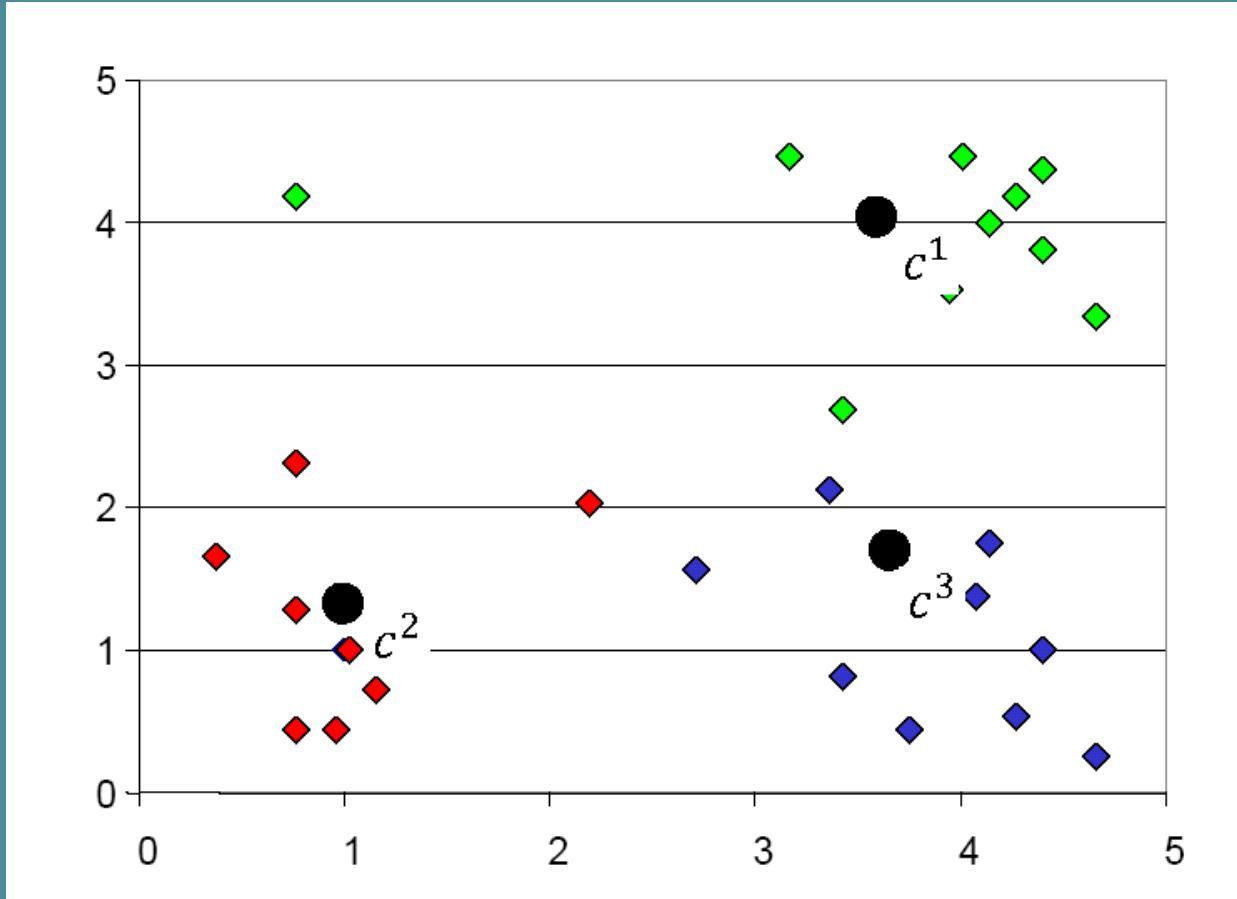
K-means: step 1



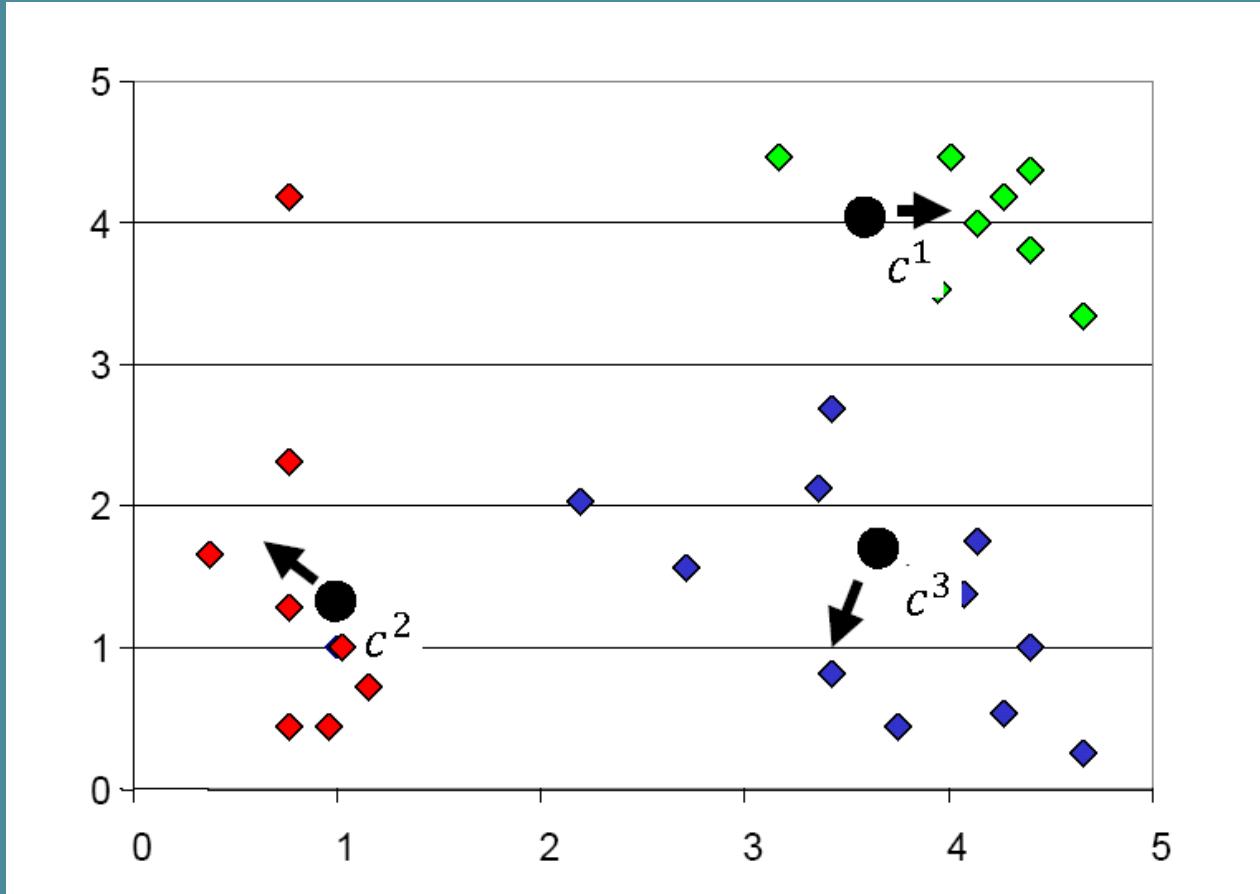
K-means: step 2



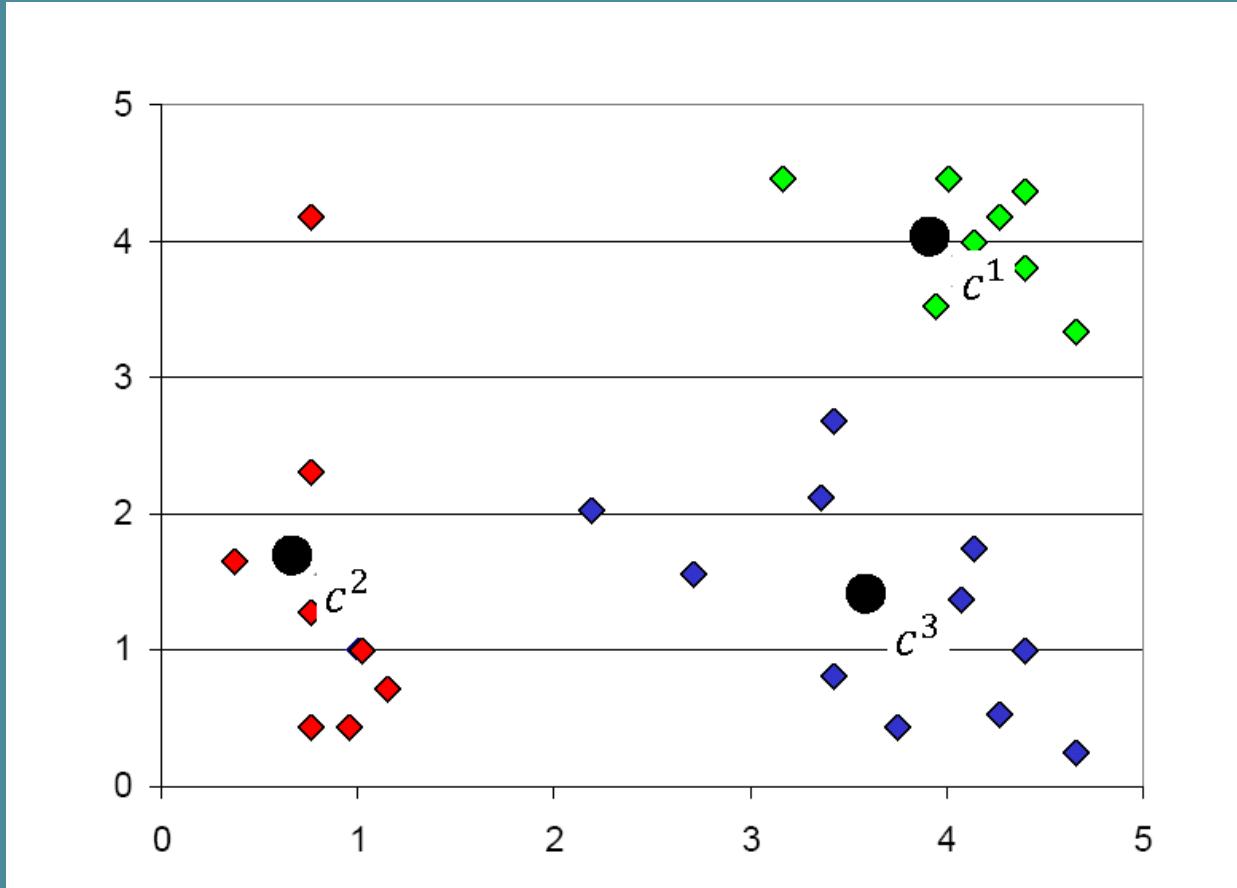
K-means: step 3



K-means: step 4



K-means: step 5



Questions

- Will different initialization lead to different results?
 - Yes ✓
 - No
 - Sometimes

- Will the algorithm always stop after some iteration?
 - Yes ✓
 - No (we have to set a maximum number of iterations)
 - Sometimes

Formal statement of k-means objective

- Given m data points, $\{x^1, x^2, \dots, x^m\} \in R^n$
- Find k cluster centers, $\{c^1, c^2, \dots, c^k\} \in R^n$
- And assign each data point i to one cluster, $\pi(i) \in \{1, \dots, k\}$
- Such that the averaged square distances from each data point to its respective cluster center is small

$$\min_{c,\pi} \frac{1}{m} \sum_{i=1}^m \|x^i - c^{\pi(i)}\|^2$$

Kmeans is an optimization problem.
Optimize this equation.
The equation shows the distortion
(distance square from assigned center)

Total distortion

Clustering is NP-hard in general

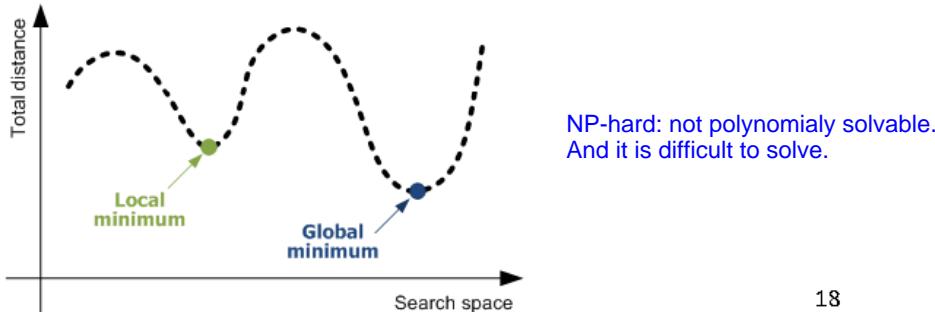
So the k means basically is an approximate way to solve this cos function.

- Find k cluster centers, $\{c^1, c^2, \dots, c^k\} \in R^n$, and assign each data point i to one cluster, $\pi(i) \in \{1, \dots, k\}$, to minimize

$$\min_{c,\pi} \frac{1}{m} \sum_{i=1}^m \|x^i - c^{\pi(i)}\|^2$$



- A search problem over the space of discrete assignments
 - For all m data point together, there are k^m possibility
 - The cluster assignment determines cluster centers, and vice versa



The reason is it has a nature, in order to find the optimal assignment, we have to enumerate through all the possible combinations of the assignment. And you have n samples, you have k clusters and then foreach data point, you have k different ways to assign them. So in total, there will be N raised to power k , number of possible ways to design is different samples.

Convergence of kmeans algorithm

- Will kmeans objective oscillate?

$$\frac{1}{m} \sum_{i=1}^m \|x^i - c^{\pi(i)}\|^2$$

- The minimum value of the objective is finite
- Each iteration of kmeans algorithm decrease the objective
 - Cluster assignment step decreases objective
 - $\pi(i) = \operatorname{argmin}_{j=1,\dots,k} \|x^i - c^j\|^2$ for each data point i
 - Center adjustment step decreases objective
 - $c^j = \frac{1}{|\{i:\pi(i)=j\}|} \sum_{i:\pi(i)=j} x^i = \operatorname{argmin}_c \sum_{i:\pi(i)=j} \|x^i - c\|^2$

So the kmeans our objective will not oscillate.

The reason is, in every iteration we always decrease the cost function.

We always improve our assignment.

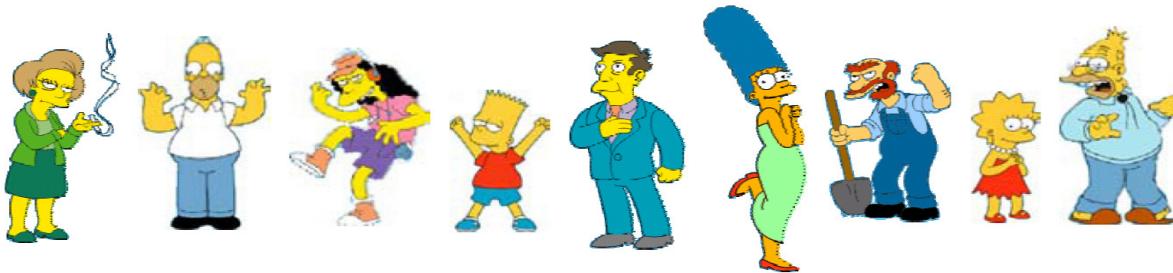
And we always improve our centers based on the new assignment.

So the kmeans objective function will always be decreasing.

So, now we can draw the conclusion that, Kmeans algorithm will always converge. Because it always methodically decreases the objective function in each iteration, and the smallest possible value you can have zero.

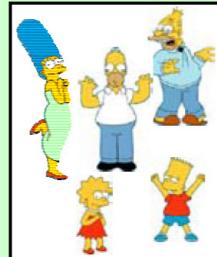
So it will converge after certain number of steps.

What else needs to be considered?



What is consider similar/dissimilar?

Clustering is subjective



Simpson's Family



School Employees



Females



Males

You pick your similarity/dissimilarity



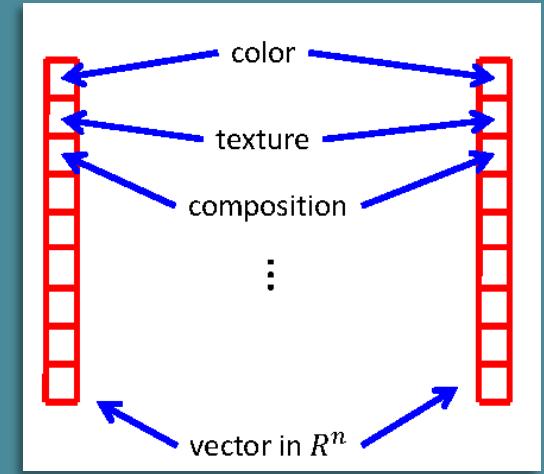
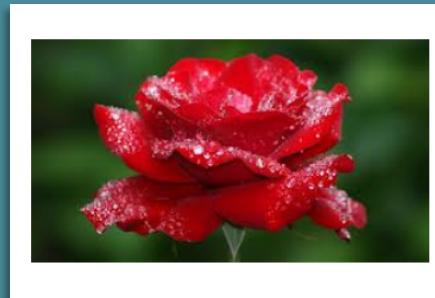
<https://www.paddingtonpups.com.au/tag/similarities-between-owners-and-dogs>

So what is clustering in general?

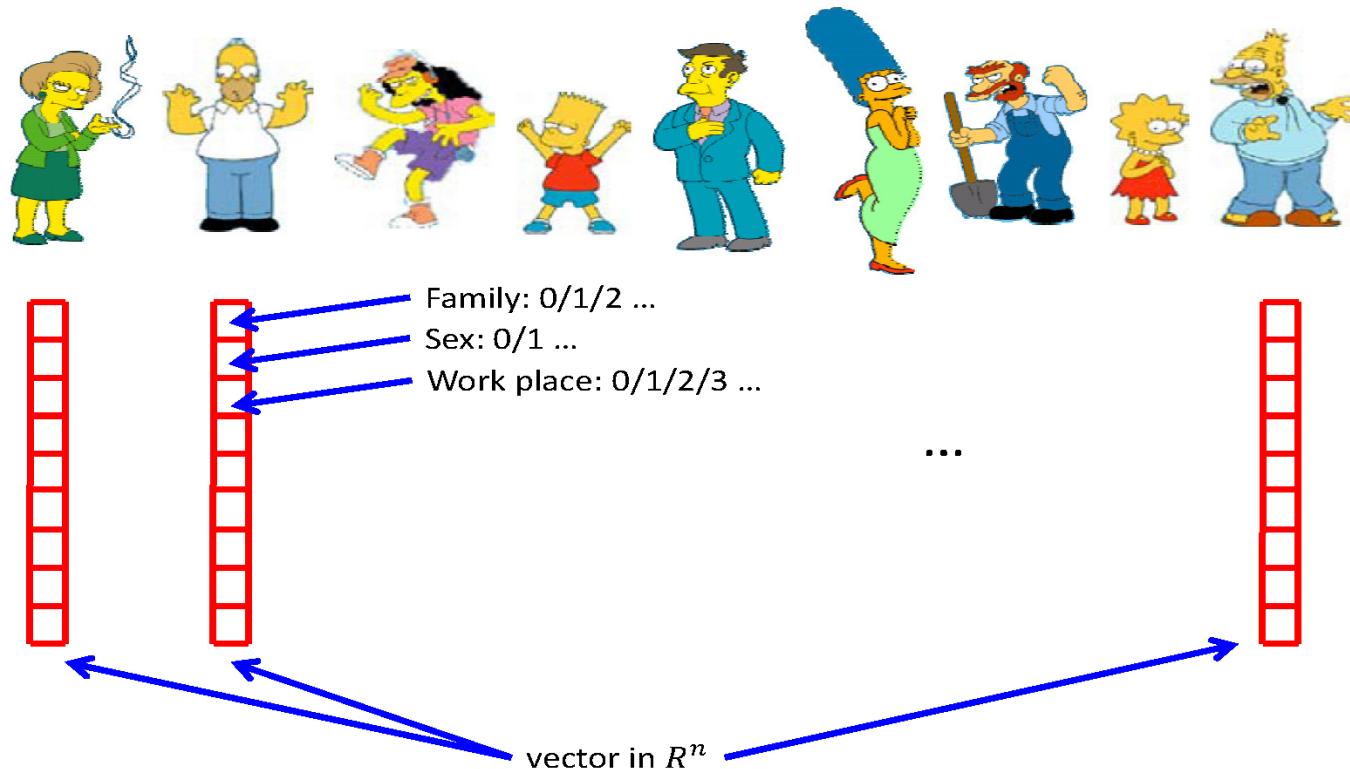
Similarity/Dissimilarity function.

- You pick your similarity/dissimilarity function
- The algorithm figures out the grouping of objects based on the chosen similarity/dissimilarity function
 - Points within a cluster is similar
 - Points across clusters are not so similar

Images of different sizes



Objects in real life



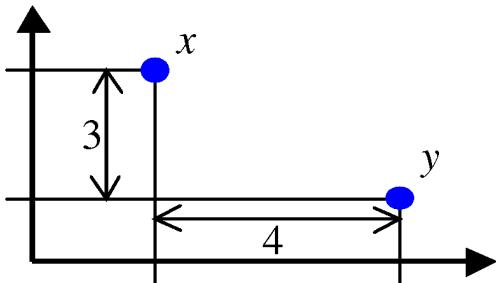
What similarity/dissimilarity function?

- Desired properties of dissimilarity function
 - Symmetry: $d(x, y) = d(y, x)$
 - Otherwise you could claim "Alex looks like Bob, but Bob looks nothing like Alex"
 - Positive separability: $d(x, y) = 0$, if and only if $x = y$
 - Otherwise there are objects that are different, but you cannot tell apart
 - Triangular inequality: $d(x, y) \leq d(x, z) + d(z, y)$
 - Otherwise you could claim "Alex is very like Bob, and Alex is very like Carl, but Bob is very unlike Carl"

Distance functions for vectors

- Suppose two data points, both in R^n
 - $x = (x_1, x_2, \dots, x_n)^\top$
 - $y = (y_1, y_2, \dots, y_n)^\top$
- Euclidian distance: $d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$
- Minkowski distance: $d(x, y) = \sqrt[p]{\sum_{i=1}^n (x_i - y_i)^p}$
 - Euclidian distance: $p = 2$
 - Manhattan distance: $p = 1, d(x, y) = \sum_{i=1}^n |x_i - y_i|$
 - “inf”-distance: $p = \infty, d(x, y) = \max_{i=1}^n |x_i - y_i|$

Distance example



- Euclidian distance: $\sqrt{4^2 + 3^2} = 5$
- Manhattan distance: $4 + 3 = 7$
- “inf”-distance: $\max\{4,3\} = 4$

Hamming distance

- Manhattan distance is also called *Hamming distance* when all features are binary
 - Count the number of difference between two binary vectors
 - Example, $x, y \in \{0,1\}^{17}$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
x	0	1	1	0	0	1	0	0	1	0	0	1	1	1	0	0	1
y	0	1	1	1	0	0	0	0	1	1	1	1	1	1	0	1	1

$$d(x, y) = 5$$

Edit distance

- Transform one of the objects into the other, and measure how much effort it takes

x	I	N	T	E	*	N	T	I	O	N
y	*	E	X	E	C	U	T	I	O	N
	d	s	s		i	s				

d: deletion (cost 5)

s: substitution (cost 1)

i: insertion (cost 2)

$$d(x, y) = 5 \times 1 + 3 \times 1 + 1 \times 2 = 10$$

Generalization of K-means

- Given m data points, $\{x^1, x^2, \dots, x^m\} \in R^n$
- Find k cluster centers, $\{c^1, c^2, \dots, c^k\} \in R^n$
- And assign each data point i to one cluster, $\pi(i) \in \{1, \dots, k\}$
- Such that the sum of the squared distances from each data point to its respective cluster center is minimized

d function is a general function. We should be able to design based on our requirements and the type of problem we are solving.

$$\min_{c,\pi} \sum_{i=1}^m d(x^i, c^{\pi(i)})^2$$



NP-hard!

Generalized K-means algorithm

- Initialize k cluster centers, $\{c^1, c^2, \dots, c^k\}$, randomly
- Do

- Decide the cluster memberships of each data point, x^i , by assigning it to the nearest cluster center

$$\pi(i) = \operatorname{argmin}_{j=1, \dots, k} d(x^i, c^j)$$

squared Euclidean distance:

$$c^j = \frac{1}{\#\{\pi(i) = j\}} \sum_{i:\pi(i)=j} x^i$$

- Adjust the cluster centers

$$c^j = \operatorname{argmin}_{v \in R^n} \sum_{i:\pi(i)=j} d(x^i, v)^2$$

- While any cluster center has been changed



See demo kmeans_digit.m



Use matlab code for demo

See demo kmeans_animation.m



Use matlab code for demo