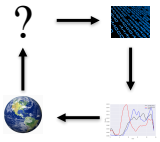
Data Science 100
Lec 25:
Clustering and K-means

DS100 Spring 2017

Slides by:
Bin Yu
binyu@stat.berkeley.edu
Joey Gonzalez
jegonzal@berkeley.edu
Thanks to Andrew Do for his assistance on data analysis



Cluster, defined by Oxford Dictionary

cluster | ˈkləstər |

noun

a group of similar things or people positioned or occurring closely together: ***clusters of*** *creamy-white flowers* | ***a cluster of*** *antique shops*.

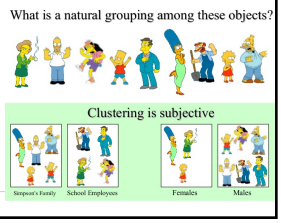
- Astronomy a group of stars or galaxies forming a relatively close association.
- Linguistics (also **consonant cluster**) a group of consonants pronounced in immediate succession, as *str* in *strong*.
- a natural subgroup of a population, used for statistical sampling or analysis.
- Chemistry a group of atoms of the same element, typically a metal, bonded closely together in a molecule.

Clustering, why bother?

Humans clustered similar things (objects and animals and people) way before statistics and machine learning…

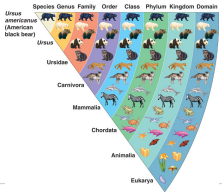
We even gave terms to the clusters: red, blue; big, small; good, bad…
Language is clustering of "reality" into words…

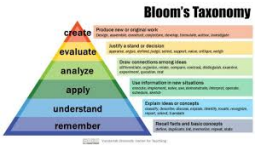
Clustering is old, vague, and subjective…



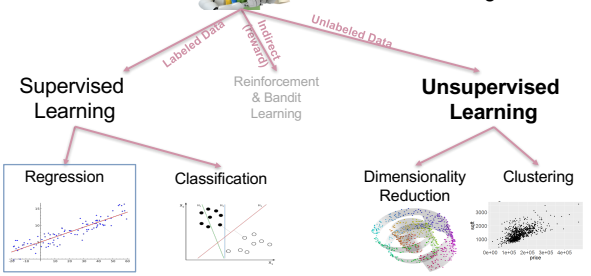
Taxonomy is clustering

Taxonomy (biology), a branch of science that encompasses the description, identification, nomenclature, and classification of organisms
-- Wikipedia





Taxonomy of Machine Learning/Statistics



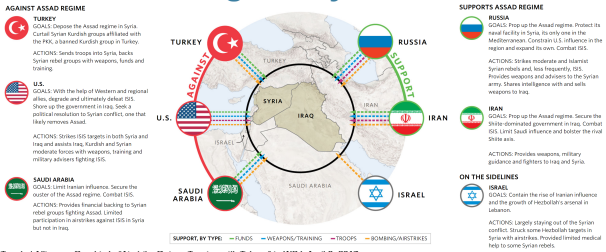
Clustering is a form of information reduction/organization

- To store in human finite memory (or computer's finite memory) and facilitate understanding

- To communicate between people (or processors) for more effective understanding between people and collective decisions

Effective decision-making is impossible based on raw big data

Understanding the Syria conflict



Tangled Alliances Graphic In "Airstrike Raises Tension with Tehran" in WSJ, April 8, 2017

Very helpful **clustered** information in terms of countries and important dimensions to consider.

One criticism: country names could be placed better, not associated with the bars.

Syria conflict

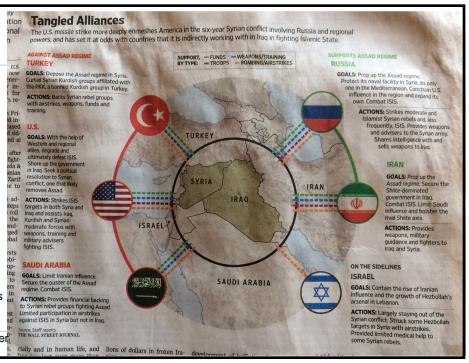
Tangled Alliances Graphic in

"Airstrike Raises Tensions with Tehran"

WSJ, April 8, 2017

Very helpful **clustered** information in terms of countries and important dimensions to consider.

One criticism: country names could be placed better, not associated with the bars- more pronounced in the paper version



QPRV in the context of Ames data

- Q (not unique: translating Q in English into a Q about data…)

To understand the data better and discover heterogeneity, which is ever present in data, especially in big data; to generate hypotheses to confirm with new data

- P: all houses in tax office from 2006-2010 in Ames
- R: yes, we did simple random sampling
- -
- V: do the clusters correspond to clusters in the population?

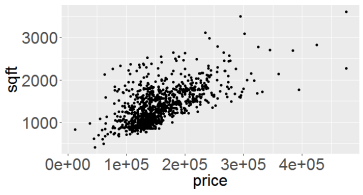
New name: PQRS (thanks to a discussion with Andrew)

- P: Population
- Q: Question
- R: Representativeness

…

- **S for Scrutinizing**

Raw and transformed (population) Ames data



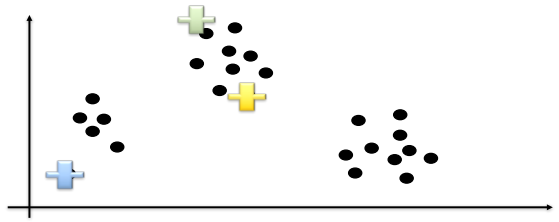
How do we Compute a Clustering?

Many different clustering models and algorithms:

- Feature based Clustering: *Points in* R^d
  - **K-Means** (aka Lloyd-Max in signal processing) alternate minimization between finding centers and cluster memberships
  - Expectation-Maximization (EM) (earliest example I know is from stat. genetics)
  - Spectral Methods: PCA (principal component analysis)+K-means on weighted or transformed data
- Hierarchical Clustering: feature based or not *clustering in a greedy fashion, widely used in biology*

K-Means Clustering: *Intuition*

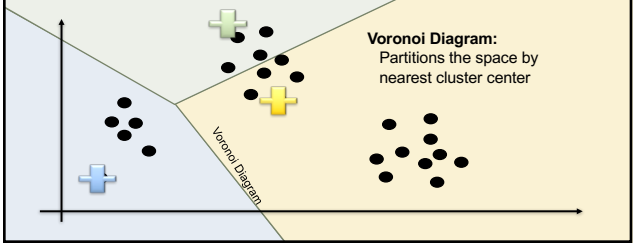
- Input K: The number of clusters to find
- Pick an initial set of points as cluster centers



K-Means Clustering: *Intuition*

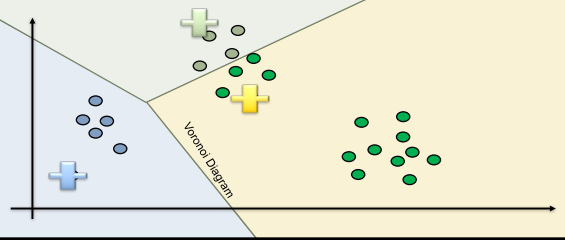
- For each data point find the cluster nearest center

Voronoi Diagram:
Partitions the space by nearest cluster center



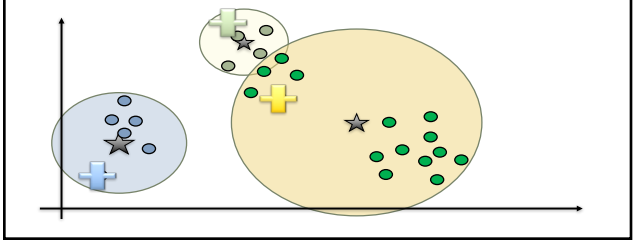
K-Means Clustering: *Intuition*

- For each data point find the cluster nearest center



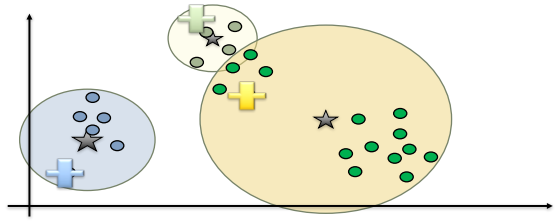
K-Means Clustering: *Intuition*

- Compute mean of points in each "cluster"



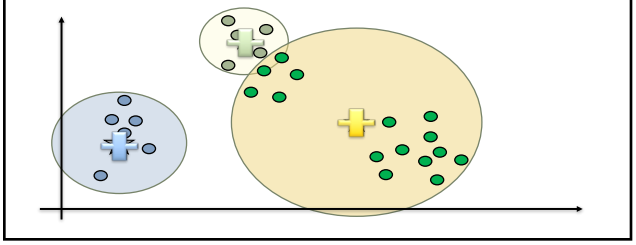
K-Means Clustering: *Intuition*

- Adjust cluster centers to be the mean of the cluster



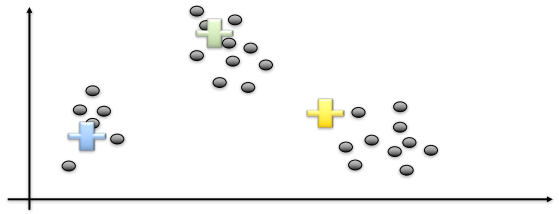
K-Means Clustering: *Intuition*

- Adjust cluster centers to be the mean of the cluster



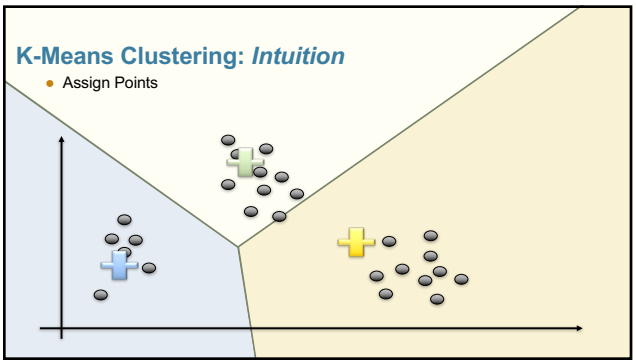
K-Means Clustering: *Intuition*

- Improved?
- Repeat



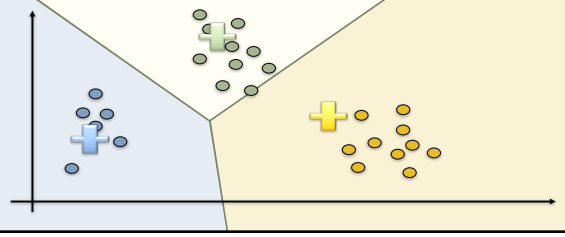
K-Means Clustering: *Intuition*

- Assign Points



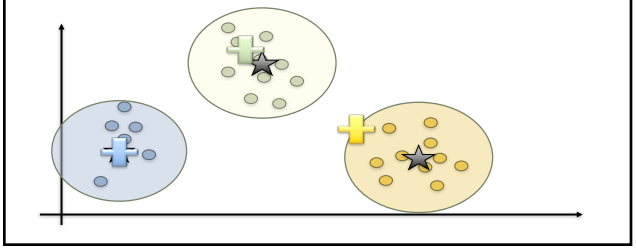
K-Means Clustering: *Intuition*

- Assign Points



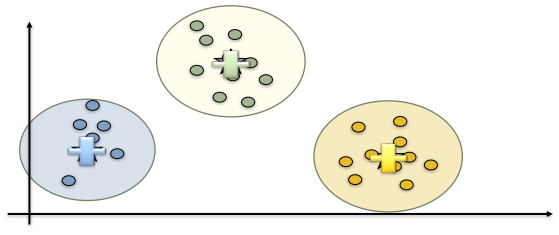
K-Means Clustering: *Intuition*

- Compute cluster means



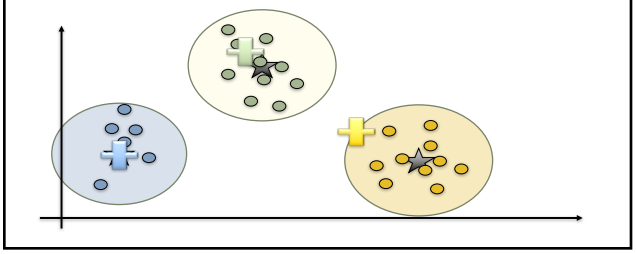
K-Means Clustering: *Intuition*

- Update cluster centers



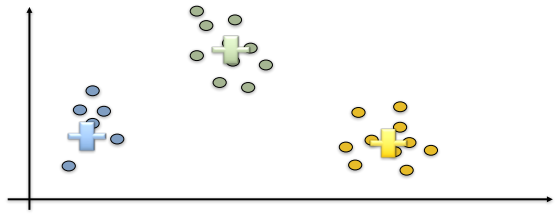
K-Means Clustering: *Intuition*

- Update cluster centers



K-Means Clustering: *Intuition*

- Repeat?
  - Yes to check that nothing changes → Converged!



K-Means Algorithm for a given k: Details

```
centers ← pick k initial Centers

while (centers are changing) {
   // Compute the assignments
   asg ← [(x, nearest(centers, x)) for x in data]
```

What do we mean by "nearest"?
A: Squared Euclidean distance

K-Means Algorithm: Details

```
centers ← pick k initial Centers

while (centers are changing) {
   // Compute the assignments
   asg ← [(x, nearest(centers, x)) for x in data]

   // Compute the new centers
   for j in range(K):
      centers[j] =
         mean([x for (x, c) in asg if c == j])
}
```

K-Means Algorithm: Details

```
centers ← pick k initial Centers

while (centers are changing) {
   // Compute the assignments
   asg ← [(x, nearest(centers, x)) for x in data]

   // Compute the new centers
   for j in range(k):
      centers[j] =
         mean([x for (x, c) in asg if c == j])
}
```

Guaranteed to converge! … *to what?* To a local optimum. ☹ Depends on Initial Centers

K-means global loss function for K clusters: n → K

Given n data points with feature vector x_i, we want
- a partition of the index set {1,...,n} into k subsets $I_1 = \{1, 3, 5\}, I_2, ..., I_K$
- and its associated cluster centers $c_1, c_2, ..., c_K$

K-means algorithm minimizes the following global loss function for a distance metric d (e.g. squared Euclidean distance) by alternating the minimizations over the partition and centers:

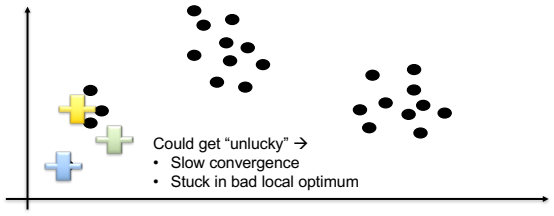
$$\sum_{j=1}^{K} \sum_{x_i : i \in I_j} d(x_i, c_j)$$

where the inner sum is over data points in a particular cluster j, and the outer sum is over the clusters

When d is absolute value loss, we have the group medians as the centers.

Picking the Initial Centers

- **Simple Strategy:** select *k* data points at random
  - What could go wrong?



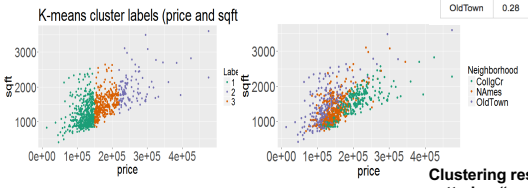
Could get "unlucky" →
- Slow convergence
- Stuck in bad local optimum

K-means with 3 clusters: random picked data points as initial centers

Clustering results with NB labels

prop. In each row category

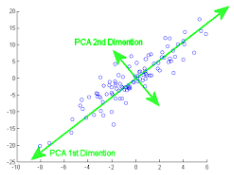
	1	2	3
NAmes	0.58	0.05	0.37
CollgCr	0.39	0.51	0.10
OldTown	0.28	0.04	0.68



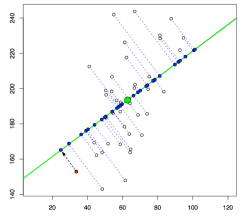
Clustering results are vetted or "scrutinized" by neighborhood labels

Principal Component Analysis (PCA)

PCA in a graph



Projecting to first PCA direction to reduce data to 1-dim



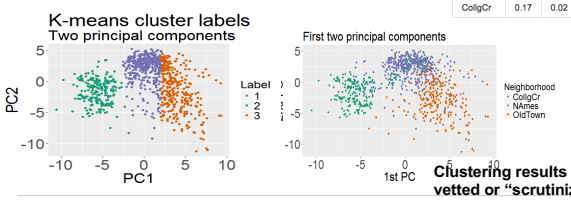
Data projected to first two PCAs give much better results

using all 81 features (untransformed)

Clustering results with NB labels

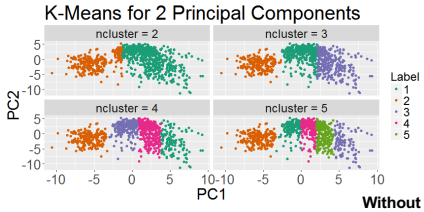
Prop. In each row category

	1	2	3
NAmes	0.83	0.14	0.03
OldTown	0.05	0.95	0.00
CollgCr	0.17	0.02	0.81



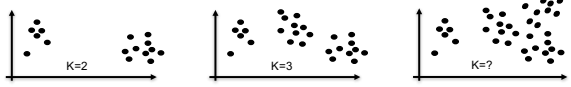
Clustering results are vetted or "scrutinized" by neighborhood labels

K-means results for K=2, 3, 4, 5: relying on first PC heavily



Without NB info, hard to know which K to use.

How do we choose K?



- Basic Elbow Method (you may try this out in your HW if you like)
- Try range of K values and plot average distance to centers
- Silhouette (graphical method, popular in stats)
- Cross-Validation (Better)
  - Repeatedly split the data into training and validation datasets
  - Cluster the training dataset
  - Measure avg. dist. To centers on validation data

Silhouette (Peter J. Rousseeuw, 1986): graphic method for K selection

Given k and k clusters, given any data point i , let a_i be the average distance or dissimilarity of i with all other points in the same cluster. For Euclidean k-means, use Euclidean distance for dissimilarity

a_i measures how well i fits into its cluster.

b_i is the smallest average distance of i to other clusters

$$s_i = \frac{b_i - a_i}{max(b_i, a_i)}$$ which is between -1 and 1.

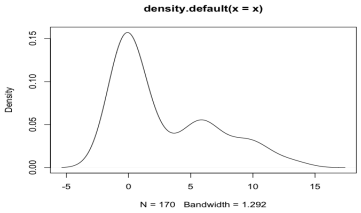
s_i is close to 1 if point i is in a tight cluster and far away from other clusters; close to -1, if it is in a loose cluster and close to other clusters.

Maximize $\frac{1}{n}\sum_{i=1}^{n} s_i$ over k.

Trying out Silhouette with simulated data

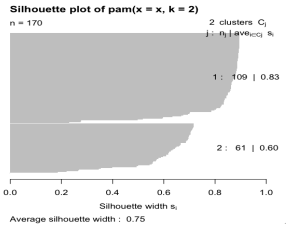
Example 1: simulated data from mixture of 3 Gaussians of n=170
N(0, 1), N(5, 4), and N(8, 9), with proportions 0.58, 0.17, and 0.23

The third Gaussian centered at 8 is not very visible.



Trying out Silhouette with 2-cluster data

Example 1: Silhouette for the results of two-clusters using pam, which is Euclidean K-means with the K-means centers as the data points closest to centers



Summary

- Clustering is an old activity and is used for information organization
- New name: PQRS
- One clustering method -- K-means algorithm: initial values, choice of K
  Euclidean distance in K-means corresponds to taking means – sensitive to outliers because of the squared Euclidean distance; using median corresponds to absolute loss function, robust.
- We do not always have labels to compare – other "S" investigation is needed to back up why the clustering results are meaningful in context
- PCA – dimensionality reduction (details to come)