

Clustering: K-Means and Mixtures of Gaussians

CS4780/5780 – Machine Learning
Fall 2013

Thorsten Joachims
Cornell University

Reading: Manning/Raghavan/Schuetze,
Chapters 16 (not 16.3) and 17
(<http://nlp.stanford.edu/IR-book/>)

Outline

- Supervised vs. Unsupervised Learning
- Hierarchical Clustering
 - Hierarchical Agglomerative Clustering (HAC)
- Non-Hierarchical Clustering
 - K-means
 - Mixtures of Gaussians and EM-Algorithm

Non-Hierarchical Clustering

- K-means clustering (“hard”)
- Mixtures of Gaussians and training via Expectation maximization Algorithm (“soft”)

Clustering Criterion

- Evaluation function that assigns a (usually real-valued) value to a clustering
 - Clustering criterion typically function of
 - within-cluster similarity and
 - between-cluster dissimilarity
- Optimization
 - Find clustering that maximizes the criterion
 - Global optimization (often intractable)
 - Greedy search
 - Approximation algorithms

Centroid-Based Clustering

- Assumes instances are real-valued vectors.
- Clusters represented via *centroids* (i.e. average of points in a cluster) c :

$$\vec{\mu}(c) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

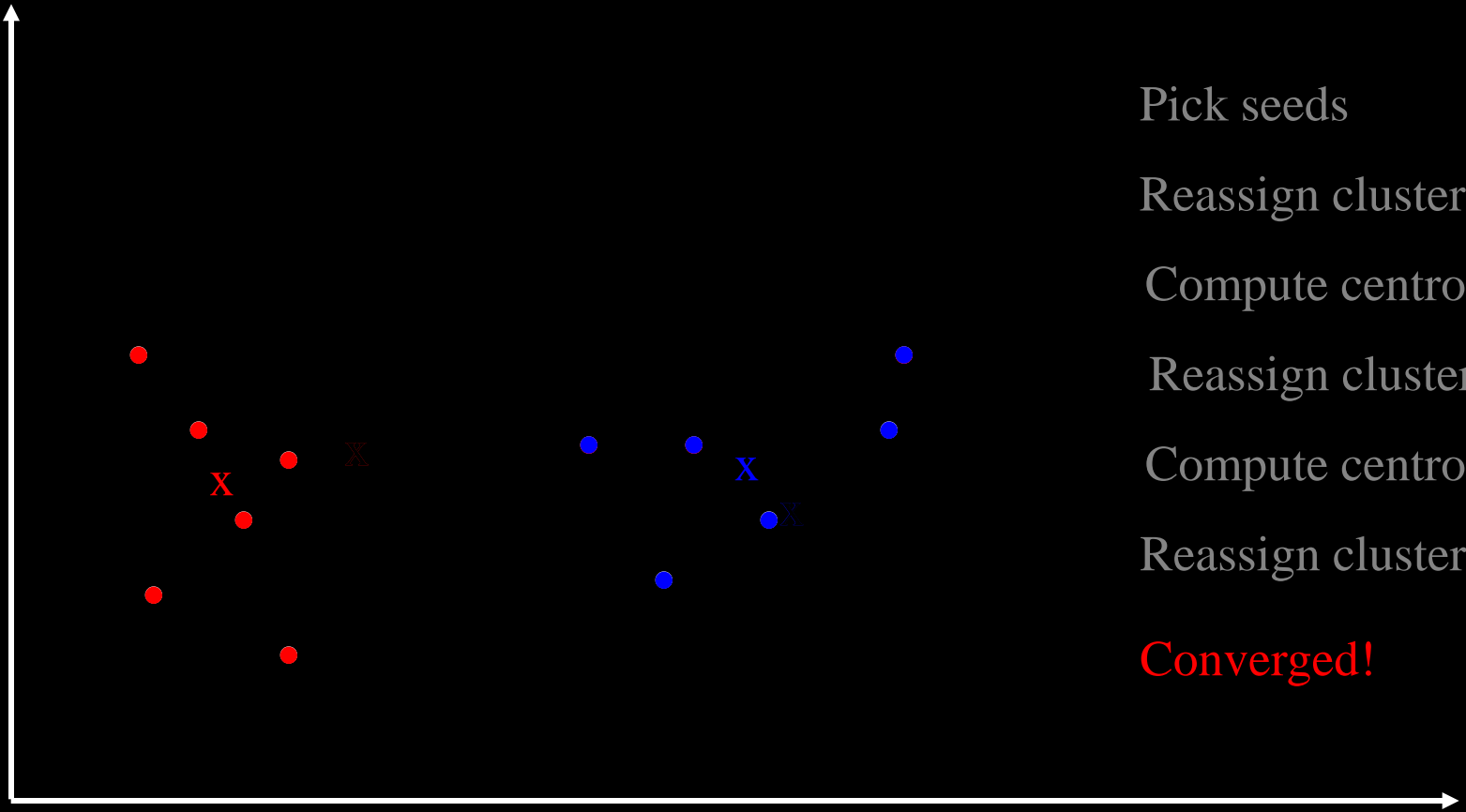
- Reassignment of instances to clusters is based on **distance** to the current cluster centroids.

K-Means Algorithm

- Input: k = number of clusters, distance measure d
- Select k random instances $\{s_1, s_2, \dots, s_k\}$ as seeds.
- Until clustering converges or other stopping criterion:
 - For each instance x_i :
 - Assign x_i to the cluster c_j such that $d(x_i, s_j)$ is min.
 - For each cluster c_j //update the centroid of each cluster
 - $s_j = \mu(c_j)$

K-means Example

(k=2)



Pick seeds

Reassign clusters

Compute centroids

Reassign clusters

Compute centroids

Reassign clusters

Converged!

Time Complexity

- Assume computing distance between two instances is $O(N)$ where N is the dimensionality of the vectors.
- Reassigning clusters for n points: $O(kn)$ distance computations, or $O(knN)$.
- Computing centroids: Each instance gets added once to some centroid: $O(nN)$.
- Assume these two steps are each done once for i iterations: $O(iknN)$.
- Linear in all relevant factors, assuming a fixed number of iterations, more efficient than HAC.

Buckshot Algorithm

Problem

- Results can vary based on random seed selection, especially for high-dimensional data.
- Some seeds can result in poor convergence rate, or convergence to sub-optimal clusterings.

Idea: Combine HAC and K-means clustering.

- First randomly take a sample of instances of size $n^{1/2}$
- Run group-average HAC on this sample $n^{1/2}$
- Use the results of HAC as initial seeds for K-means.
- Overall algorithm is efficient and avoids problems of bad seed selection.

Clustering as Prediction

- Setup
 - Learning Task: $P(X)$
 - Training Sample: $S = (\vec{x}_1, \dots, \vec{x}_n)$
 - Hypothesis Space: $H = \{h_1, \dots, h_{|H|}\}$ each describes $P(X|h_i)$ where h_i are parameters
 - Goal: learn which $P(X|h_i)$ produces the data
- What to predict?
 - Predict where new points are going to fall

Gaussian Mixtures and EM

- Gaussian Mixture Models

- Assume

$$P(X = \vec{x} | h_i) = \sum_{j=1}^k P(X = \vec{x} | Y = j, h_i) P(Y = j)$$

where $P(X = \vec{x} | Y = j, h) = N(X = \vec{x} | \vec{\mu}_j, \Sigma_j)$

and $h = (\vec{\mu}_1, \dots, \vec{\mu}_k, \Sigma_1, \dots, \Sigma_k)$.

- EM Algorithm

- Assume $P(Y)$ and k known and $\sum_i = 1$.

- REPEAT

- $$\vec{\mu}_j = \frac{\sum_{i=1}^n P(Y=j|X=\vec{x}_i, \vec{\mu}_1, \dots, \vec{\mu}_k) \vec{x}_i}{\sum_{i=1}^n P(Y=j|X=\vec{x}_i, \vec{\mu}_1, \dots, \vec{\mu}_k)}$$

- $$P(Y = j | X = \vec{x}_i, \vec{\mu}_1, \dots, \vec{\mu}_k) = \frac{P(X=\vec{x}_i|Y=j, \vec{\mu}_j) P(Y=j)}{\sum_{l=1}^k P(X=\vec{x}_i|Y=l, \vec{\mu}_l) P(Y=l)} = \frac{e^{-0.5(\vec{x}_i - \vec{\mu}_j)^2} P(Y=j)}{\sum_{l=1}^k e^{-0.5(\vec{x}_i - \vec{\mu}_l)^2} P(Y=l)}$$