Department of Computer Science

CS47300: Web Information Search and Management

Text Clustering: K-Means
Prof. Chris Clifton
7 October 2020
Borrows slides from Chris Manning, Ray Mooney and Soumen Chakrabarti

Indiana Center for Database Systems ™

Department of Computer Science

K-Means

- Assumes documents are real-valued vectors.
- Clusters based on *centroids* (aka the *center of gravity* or mean) 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.
  – (Or one can equivalently phrase it in terms of similarities)

33

PURDUE UNIVERSITY
Department of Computer Science

K-Means Algorithm

Let d be the distance measure between instances

Select k random instances $\{s_1, s_2, \ldots s_k\}$ as seeds

Until clustering converges or other stopping criterion:

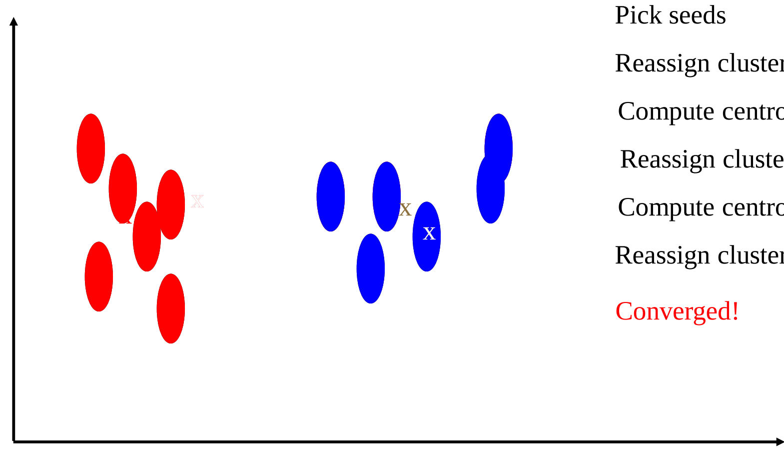
- For each instance x_i:
  - Assign x_i to the cluster c_j such that $d(x_i, s_j)$ is minimized
- *(Update the seeds to the centroid of each cluster)*
- For each cluster c_j
  - $s_j = \mu(c_j)$

34

PURDUE UNIVERSITY
Department of Computer Science

K Means Example

(K=2)



Pick seeds

Reassign clusters

Compute centroids

Reassign clusters

Compute centroids

Reassign clusters

Converged!

35

PURDUE UNIVERSITY
Department of Computer Science

Termination conditions

- Several possibilities, e.g.,
  - A fixed number of iterations.
  - Doc partition unchanged.
  - Centroid positions don't change.

Does this mean that the docs in a cluster are unchanged?

36

PURDUE UNIVERSITY
Department of Computer Science

Time Complexity

- Assume computing distance between two instances is *O(m)* where *m* is the dimensionality of the vectors.
- Reassigning clusters: *O(kn)* distance computations, or *O(knm)*.
- Computing centroids: Each instance vector gets added once to some centroid: *O(nm)*.
- Assume these two steps are each done once for *i* iterations: *O(iknm)*.
- Linear in all relevant factors, assuming a fixed number of iterations, more efficient than hierarchical agglomerative methods

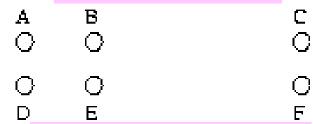
37

PURDUE UNIVERSITY
Department of Computer Science

Seed Choice

- Results can vary based on random seed selection.
- Some seeds can result in poor convergence rate, or convergence to sub-optimal clusterings.
  - Select good seeds using a heuristic (e.g., doc least similar to any existing mean)
  - Try out multiple starting points
  - Initialize with the results of another method.

Example showing sensitivity to seeds



In the above, if you start with B and E as centroids you converge to {A,B,C} and {D,E,F}
If you start with D and F you converge to {A,B,D,E} {C,F}

Exercise: find good approach for finding good starting points

38

PURDUE UNIVERSITY
Department of Computer Science

Recap

- Why cluster documents?
  - For improving recall in search applications
  - For speeding up vector space retrieval
  - Navigation
  - Presentation of search results
- *k*-means basic iteration
  - At the start of the iteration, we have *k* centroids.
  - Each doc assigned to the nearest centroid.
  - All docs assigned to the same centroid are averaged to compute a new centroid;
    - thus have *k* new centroids.

How Many Clusters?

- Number of clusters *k* is given
  - Partition *n* docs into predetermined number of clusters
- Finding the "right" number of clusters is part of the problem
  - Given docs, partition into an "appropriate" number of subsets.
  - E.g., for query results - ideal value of *k* not known up front - though UI may impose limits.
- Can usually take an algorithm for one flavor and convert to the other.

k not specified in advance

- Say, the results of a query.
- Solve an optimization problem: penalize having lots of clusters
  - application dependent, e.g., compressed summary of search results list.
- Tradeoff between having more clusters (better focus within each cluster) and having too many clusters

k not specified in advance

- Given a clustering, define the Benefit for a doc to be the cosine similarity to its centroid
- Define the Total Benefit to be the sum of the individual doc Benefits.

Why is there always a clustering of Total Benefit *n*?

Penalize lots of clusters

- For each cluster, we have a Cost *C*.
- Thus for a clustering with *k* clusters, the Total Cost is *kC*.
- Define the Value of a clustering to be =

  Total Benefit - Total Cost.
- Find the clustering of highest value, over all choices of *k*.
  - Total benefit increases with increasing K. But can stop when it doesn't increase by "much". The Cost term enforces this.

PURDUE UNIVERSITY
Department of Computer Science

Convergence

- Why should the K-means algorithm ever reach a *fixed point*?
  - A state in which clusters don't change.
- K-means is a special case of a general procedure known as the *Expectation Maximization (EM) algorithm*.
  - EM is known to converge.
  - Number of iterations could be large.

PURDUE UNIVERSITY
Department of Computer Science

Convergence of K-Means

- Define goodness measure of cluster k as sum of squared distances from cluster centroid:
  - $G_k = \Sigma_i (v_i - c_k)^2$ (sum all v_i in cluster k)
- $G = \Sigma_k G_k$
- Reassignment monotonically reduces G since each vector is assigned to the closest centroid.
- Recomputation monotonically decreases each G_k since: (m_k is number of members in cluster)
  - $\Sigma (v_{in} - a)^2$ reaches minimum for:
  - $\Sigma -2(v_{in} - a) = 0$

K-means issues, variations, etc.

- Recomputing the centroid after every assignment (rather than after all points are re-assigned) can improve speed of convergence of K-means
- Assumes clusters are spherical in vector space
  - Sensitive to coordinate changes, weighting etc.
- Disjoint and exhaustive
  - Doesn't have a notion of "outliers"

Soft Clustering

- Clustering typically assumes that each instance is given a "hard" assignment to exactly one cluster.
- Does not allow uncertainty in class membership or for an instance to belong to more than one cluster.
- *Soft clustering* gives probabilities that an instance belongs to each of a set of clusters.
- Each instance is assigned a probability distribution across a set of discovered categories (probabilities of all categories must sum to 1).