CLUSTERING

Today

- Reading
  - Introduction to Information Retrieval (IR) Ch. 16, 17
- Goals
  - Finish Agglomerative clustering
  - Briefly look at Divisive clustering
  - Evaluating cluster quality

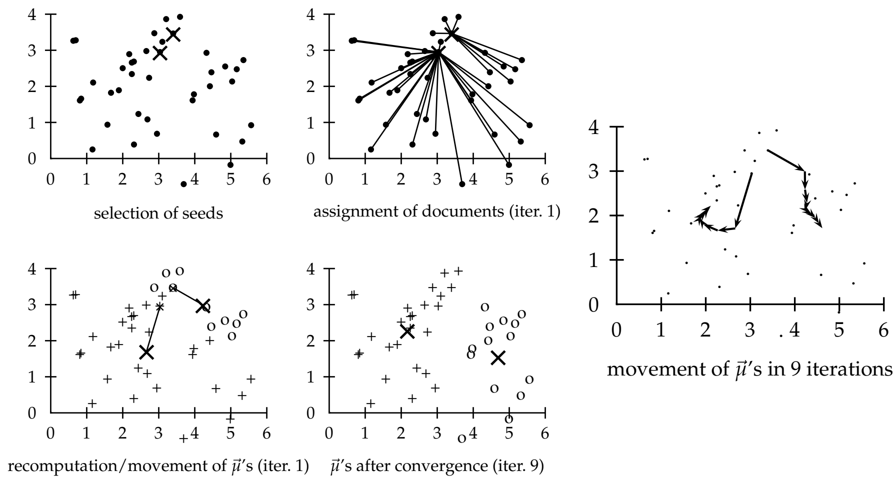
Lots of Announcements!

- If you haven't voted for a movie yet, login to Piazza to do so
- Review the list of topics for exam 2
  - Friday April 25th
  - 50 minute in-class
- Review the schedule of events for end of semester

Types of clustering algorithms

- Flat versus Hierarchical
  - Flat algorithms return an unstructured set of clusters
  - Hierarchical algorithms return a hierarchy of clusters
- Sequential (online) versus Batch
  - Sequential algorithms are typically fast
- Hard versus soft
  - Hard algorithms make a hard assignment of elements to clusters
  - Soft algorithms compute a distribution over clusters for each element

K-means Clustering



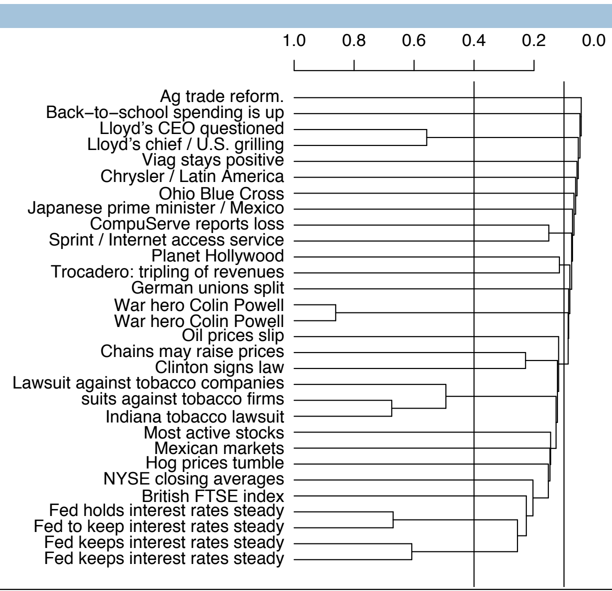
Flat versus Hierarchical

- K-means
  - Returns unstructured set of clusters
  - Requires user to determine K
  - Non-deterministic
  - Linear run time O(IKNM)
- Hierarchical (e.g. Agglomerative clustering)
  - Returns a hierarchy of clusters
  - No need to (initially) determine K
  - Deterministic
  - Quadratic run time

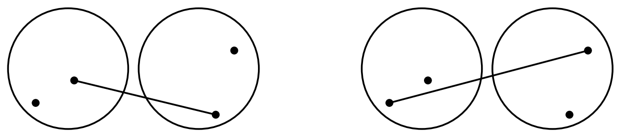
Hierarchical Clustering

- Agglomerative clustering
  - Start with *N* clusters each with one data point
  - Merge similar clusters to form larger clusters until there is only a single cluster left
- Divisive Clustering
  - Start with a single cluster containing all data points
  - Divide large clusters into smaller clusters until each cluster contains a single data point

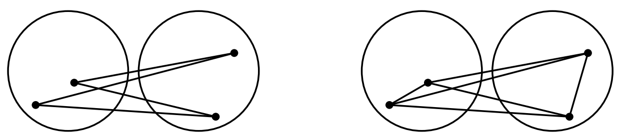
Agglomerative Clustering



Agglomerative Clustering



(a) single-link: maximum similarity (b) complete-link: minimum similarity



(c) centroid: average inter-similarity (d) group-average: average of all similarities

Cluster similarity: Single-link

- Single link
  - Similarity of c_j and c_i U c_m is the similarity of their *most similar* members
  - Can result in unwanted "long" clusters due to chaining

$$sim((c_i \cup c_m), c_j) = \max(sim(c_i, c_j), sim(c_m, c_j))$$

Cluster similarity: Complete-link

- Complete link
  - Similarity of c_j and c_i U c_m is the similarity of their least similar members
  - Makes "tighter" spherical clusters that are typically preferable.
  - Sensitive to outliers

$$sim((c_i \cup c_m), c_j) = \min(sim(c_i, c_j), sim(c_m, c_j))$$

Cluster similarity: Group-average

- Group-average (average-link)
  - Uses all vectors in clusters c_j and c_i U c_m to compute similarity
  - Average similarity between all pairs of vectors from c_j and c_i U c_m (including pairs from same cluster)
  - Efficient computing of the group-average can be done if using cosine similarity

$$\mathrm{sim}(c_i, c_j) = \frac{1}{\Big(|c_i| + |c_j|\Big)\Big(|c_i| + |c_j| - 1\Big)} \sum_{\substack{\vec{x}_n, \vec{x}_m \in c_i \cup c_j \\ \vec{x}_n \neq \vec{x}_m}} d(\vec{x}_n, \vec{x}_m)$$

Total number of pairs

All pairs of distinct vectors from c_i U c_j

Cluster similarity: Centroid

- Centroid clustering
  - Similarity of cluster c_j and cluster c_i U c_m is the similarity of their centroids

$$\begin{aligned}\text{SIM-CENT}(\omega_i,\omega_j) &= \vec{\mu}(\omega_i)\cdot\vec{\mu}(\omega_j)\\ &= (\frac{1}{N_i}\sum_{d_m\in\omega_i}\vec{d}_m)\cdot(\frac{1}{N_j}\sum_{d_n\in\omega_j}\vec{d}_n)\\ &= \frac{1}{N_iN_j}\sum_{d_m\in\omega_i}\sum_{d_n\in\omega_j}\vec{d}_m\cdot\vec{d}_n\end{aligned}$$

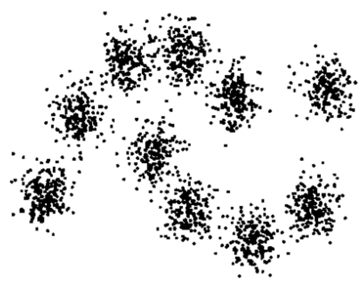
  - Equivalent to the average similarity of all pairs of documents from different clusters
  - Similarity between clusters can increase as we merge clusters (known as inversions)
    - Horizontal merge lines can be lower than the previous merge line

Divisive Clustering

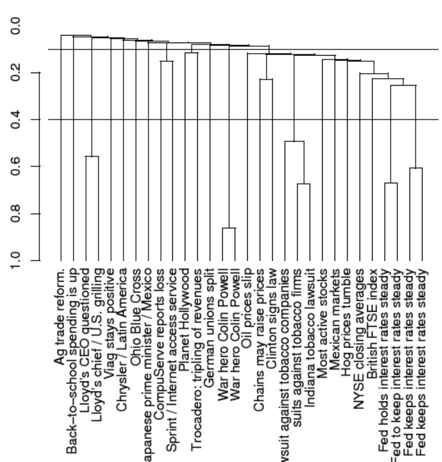
- Top-down clustering
- Divisive clustering algorithm uses a flat clustering algorithm as a subroutine
  - Start with all data points in one cluster
  - Split using a flat clustering algorithm
  - Apply recursively until each data point is in its own cluster
- Can be more efficient than agglomerative
- Benefits from complete information about the entire data set

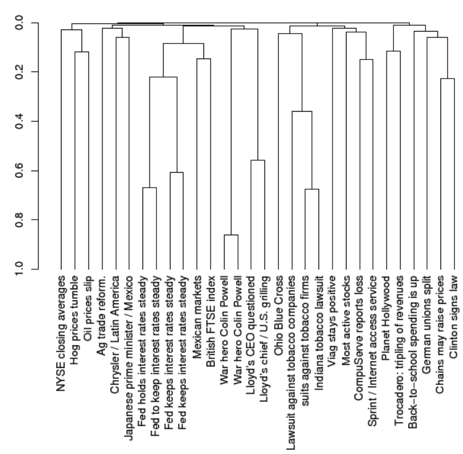
Which clustering is correct?

- Different techniques cluster the same data set differently.
- Who is right? Is there a "right" clustering?



Which clustering is correct?



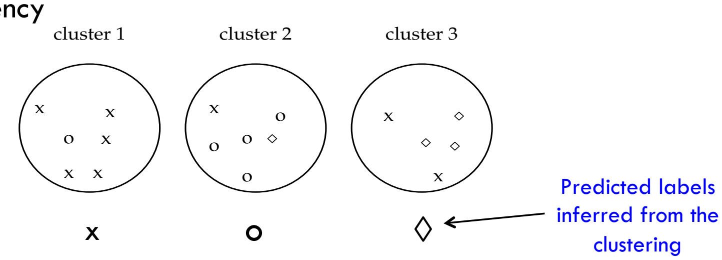


Which clustering is correct?

- Internal criteria
  - A good clustering has high intra-cluster similarity and low inter-cluster similarity
- External criteria
  - Use an external task (e.g. search, document classification) to validate the clustering
  - Requires labeled data

External Criteria

- Purity
  - Set aside labels from labeled data
  - Cluster data
  - Predicted label for each cluster is label with highest frequency



  - Compute accuracy: $\frac{5+4+3}{17} = 0.71$

External Criteria

- Normalized Mutual Information
  - Mutual Information is an information theoretic quantity similar to entropy and information gain

$$\mathrm{I}(X,Y) = \sum_y \sum_x p(x,y) \log \frac{p(x,y)}{p(x)p(y)} = \mathrm{H}(X) - \mathrm{H}(X|Y)$$

  - How much information does the clustering contain about the class labels?

External Criteria

- Normalized Mutual Information
  - Define random variables for the clustering and for the class label:

$$\begin{aligned} I(\Omega;\mathbb{C}) &= \sum_k \sum_j P(\omega_k \cap c_j) \log \frac{P(\omega_k \cap c_j)}{P(\omega_k)P(c_j)} \\ &= \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} \log \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|} \end{aligned}$$

External Criteria

- Normalized Mutual Information
  - Given by the equation:

$$\mathrm{NMI}(\Omega, \mathbb{C}) = \frac{I(\Omega; \mathbb{C})}{[H(\Omega) + H(\mathbb{C})]/2}$$

  - Why are we normalizing by the entropy?

Rand Index

- Two data points should be in the same cluster if and only if they have the same label
- Define contingency table:

	same cluster	different clusters
same class	true positives (TP)	false negatives (FN)
different classes	false positives (FP)	true negatives (TN)

- Once we have a contingency table, we can compute the Rand Index which is just the accuracy

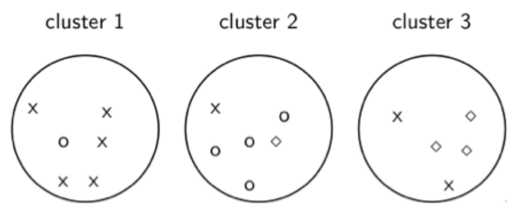
$$\mathrm{RI} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}+\mathrm{TN}}$$

Rand Index Example

- There are $\binom{17}{2} = 136$ pairs of data points

	same cluster	diff. cluster
same class	20	24
diff class	20	72

RI = (20+72)/136 = 0.68



23

F-measure

- Given the contingency table, we can compute the precision, recall, and F-measure

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN}$$

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}$$

- The parameter β controls the weighting between precision and recall

Clustering Evaluation

	purity	NMI	RI	F_5
lower bound	0.0	0.0	0.0	0.0
maximum	1.0	1.0	1.0	1.0
value for example	0.71	0.36	0.68	0.46

All four measures range from 0 (really bad clustering) to 1 (perfect clustering).