



Classification & Clustering

Hadaiq Rolis Sanabila

hadaiq@cs.ui.ac.id

Natural Language Processing and Text Mining

Pusilkom UI
22 – 26 Maret 2016





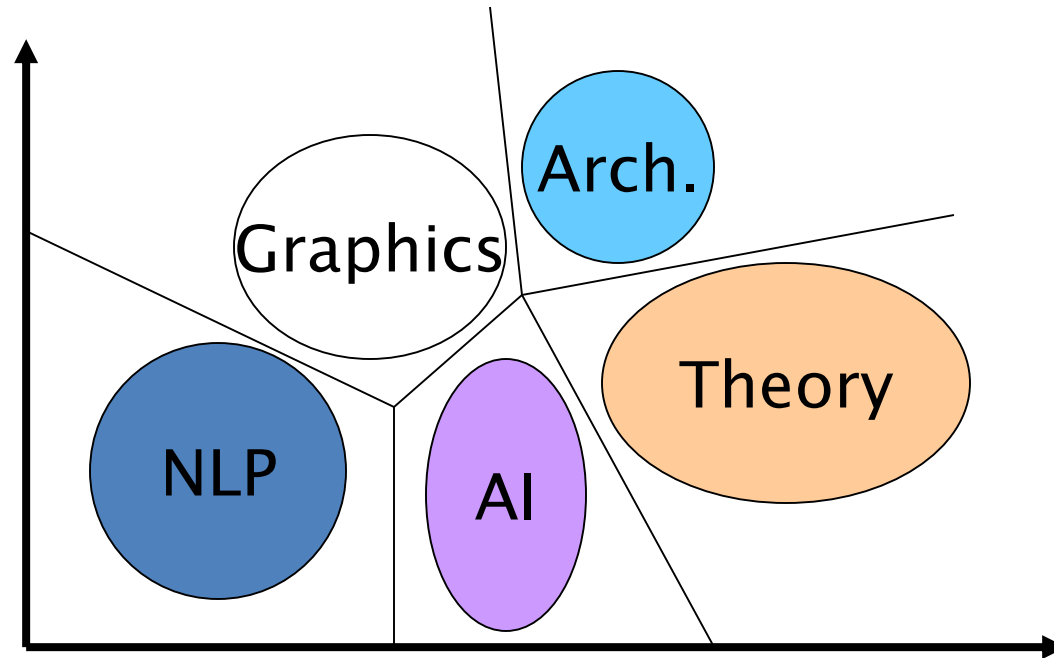
CLASSIFICATION



Categorization/Classification

- Given:
 - A description of an instance, $x \in X$, where X is the *instance language* or *instance space*.
 - Issue: how to represent text documents.
 - A fixed set of categories:
$$C = \{c_1, c_2, \dots, c_n\}$$
- Determine:
 - The category of x : $c(x) \in C$, where $c(x)$ is a *categorization function* whose domain is X and whose range is C .
 - We want to know how to build categorization functions (“classifiers”).

A GRAPHICAL VIEW OF TEXT CLASSIFICATION



EXAMPLES OF TEXT CATEGORIZATION

- LABELS=BINARY
 - “spam” / “not spam”
- LABELS=TOPICS
 - “finance” / “sports” / “asia”
- LABELS=OPINION
 - “like” / “hate” / “neutral”
- LABELS=AUTHOR
 - “Shakespeare” / “Marlowe” / “Ben Jonson”
 - The Federalist papers

Methods (1)

- Manual classification
 - Used by Yahoo!, Looksmart, about.com, ODP, Medline
 - very accurate when job is done by experts
 - consistent when the problem size and team is small
 - difficult and expensive to scale
- Automatic document classification
 - Hand-coded rule-based systems
 - Reuters, CIA, Verity, ...
 - Commercial systems have complex query languages (everything in IR query languages + *accumulators*)

Methods (2)

- Supervised learning of document-label assignment function: Autonomy, Kana, MSN, Verity, ...
 - Naive Bayes (simple, common method)
 - k-Nearest Neighbors (simple, powerful)
 - ... plus many other methods
 - No free lunch: requires hand-classified training data
 - But can be built (and refined) by amateurs



NAÏVE BAYES



Bayesian Methods

- Learning and classification methods based on probability theory (see spelling / POS)
- Bayes theorem plays a critical role
- Build a *generative model* that approximates how data is produced
- Uses *prior* probability of each category given no information about an item.
- Categorization produces a *posterior* probability distribution over the possible categories given a description of an item.

Summary – Naïve Bayes

- Bayes theorem

$$p(C = c_i | D = d_j) = \frac{P(C = c_i \cap D = d_j)}{P(D = d_j)}$$
$$= \frac{p(D = d_j | C = c_i) \times p(C = c_i)}{p(D = d_j)}$$

- *Bag of words*

$$p(C = c_i | D = d_j) = \frac{\prod_k p(w_{kj} | C = c_i) \times p(C = c_i)}{p(w_1, w_2, w_3, \dots, w_n)}$$

- Classification

$$c^* = \arg \max_{c_i \in C} \prod_k p(w_{kj} | c_i) \times p(c_i)$$

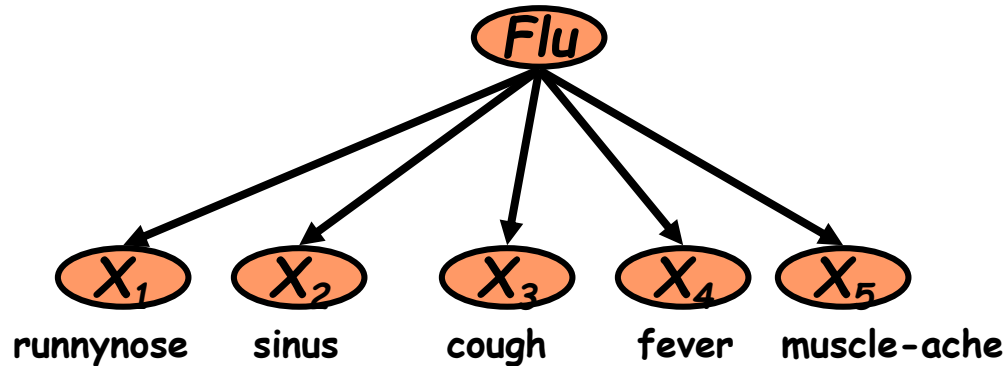
Naïve Bayes Classifier: Assumptions

- $P(c_j)$
 - Can be estimated from the frequency of classes in the training examples.
 - $P(x_1, x_2, \dots, x_n / c_j)$
 - Need very, very large number of training examples
- ⇒

Conditional Independence Assumption:

Assume that the probability of observing the conjunction of attributes is equal to the product of the individual probabilities.

The Naïve Bayes Classifier



- **Conditional Independence Assumption:**
features are independent of each other given the class:

$$P(X_1, \dots, X_5 | C) = P(X_1 | C) \bullet P(X_2 | C) \bullet \dots \bullet P(X_5 | C)$$

Text Classification Algorithms: Learning

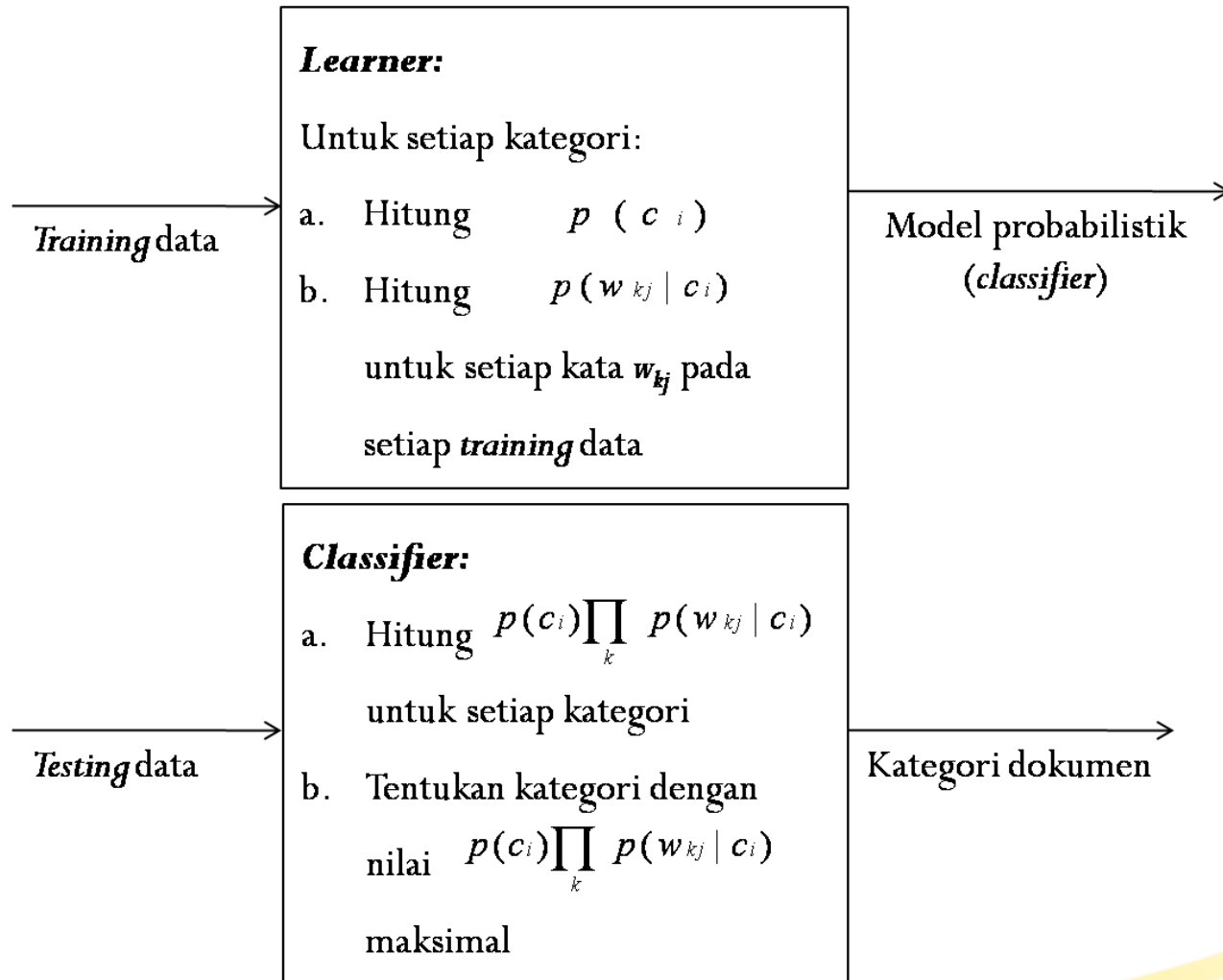
- From training corpus, extract *Vocabulary*
- Calculate required $P(c_j)$ and $P(x_k / c_j)$ terms
 - For each c_j in C do
 - $docs_j \leftarrow$ subset of documents for which the target class is c_j
 - $$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$
 - $Text_j \leftarrow$ single document containing all $docs_j$
 - for each word x_k in *Vocabulary*
 - $n_k \leftarrow$ number of occurrences of x_k in $Text_j$
 - $$P(x_k | c_j) \leftarrow \frac{n_k + 1}{n + |Vocabulary|}$$

Text Classification Algorithms: Classifying

- $positions \leftarrow$ all word positions in current document which contain tokens found in *Vocabulary*
- Return c_{NB} , where

$$c_{NB} = \operatorname{argmax}_{c_j \in C} P(c_j) \prod_{i \in positions} P(x_i | c_j)$$

Summary – Naïve Bayes



Term Document Matrix

Term Document Matrix. Representasi kumpulan dokumen yang akan digunakan untuk melakukan proses klasifikasi dokumen teks

	w_1	w_2			w_i
d_1	w_{11}	w_{12}	.	.	w_{i1}
d_2	w_{21}	w_{22}	.	.	w_{i2}
	.	.			.
	.	.			.
	.	.			.
d_j	w_{1j}	w_{2j}	.	.	w_{ij}

Term Document Matrix

Dokumen	Fitur (Kemunculan)
dokumen1	pajak (3), cukai (9), uang (2), sistem (1)
dokumen2	java (4), linux (2), sistem (6)
dokumen3	catur (2), menang (1), kalah (1), uang(1)

catur cukai java kalah linux menang pajak sistem uang

Pilihan:
Frequency
Presence
TF-IDF

dokumen 1	0	9	0	0	0	0	3	1	2
dokumen 2	0	0	4	0	2	0	0	6	0
dokumen 3	2	0	0	1	0	1	0	0	1

Contoh – Naïve Bayes

Dokumen	Kategori	Fitur (Kemunculan)
dokumen1	olahraga	menang (2), bola (3), gol (2)
dokumen2	politik	partai (3), pemilu (2), capres (4)
dokumen3	?	partai (2), menang (1), tandang (2)

	bola	capres	gol	menang	partai	pemilu	tandang
dokumen 1	3	0	2	2	0	0	0
dokumen 2	0	4	0	0	3	2	0
dokumen 3	0	0	0	1	2	0	2

Contoh – Naïve Bayes (2)

Using smoothing

$$p(w_{kj} | c_i) = \frac{f(w_{kj}, c_i) + 1}{f(c_i) + |W|} \quad p(c_i) = \frac{f_d(c_i)}{|D|}$$

Kategori	$p(c_i)$	$p(w_{kj} c_i)$						
		bola	capres	gol	menang	partai	pemilu	tandang
olahraga	$1/2$	$4/14$	$1/14$	$3/14$	$3/14$	$1/14$	$1/14$	$1/14$
politik	$1/2$	$1/16$	$5/16$	$1/16$	$1/16$	$4/16$	$3/16$	$1/16$

Contoh – Naïve Bayes (3)

$$c^* = \arg \max_{c \in C} \prod_k p(w_{kj} | c_i) \times p(c_i)$$

$$p(\text{"olahraga"} | \text{"dokumen3"}) = p(\text{"olahraga"}) \times p(\text{"partai"} | \text{"olahraga"}) \times p(\text{"menang"} | \text{"olahraga"}) \times p(\text{"tandang"} | \text{"olahraga"})$$

$$= 1/2 \times 1/14 \times 3/14 \times 1/14$$

$$= 3/5488 \approx 0,0000594$$

$$p(\text{"politik"} | \text{"dokumen3"}) = p(\text{"politik"}) \times p(\text{"partai"} | \text{"politik"}) \times p(\text{"menang"} | \text{"politik"}) \times p(\text{"tandang"} | \text{"politik"})$$

$$= 1/2 \times 4/16 \times 1/16 \times 1/16$$

$$= 1/2048 \approx 0,0004882$$

karena $p(\text{"politik"} | \text{"dokumen3"}) > p(\text{"olahraga"} | \text{"dokumen3"})$, maka kategori dari dokumen3 adalah **politik**.



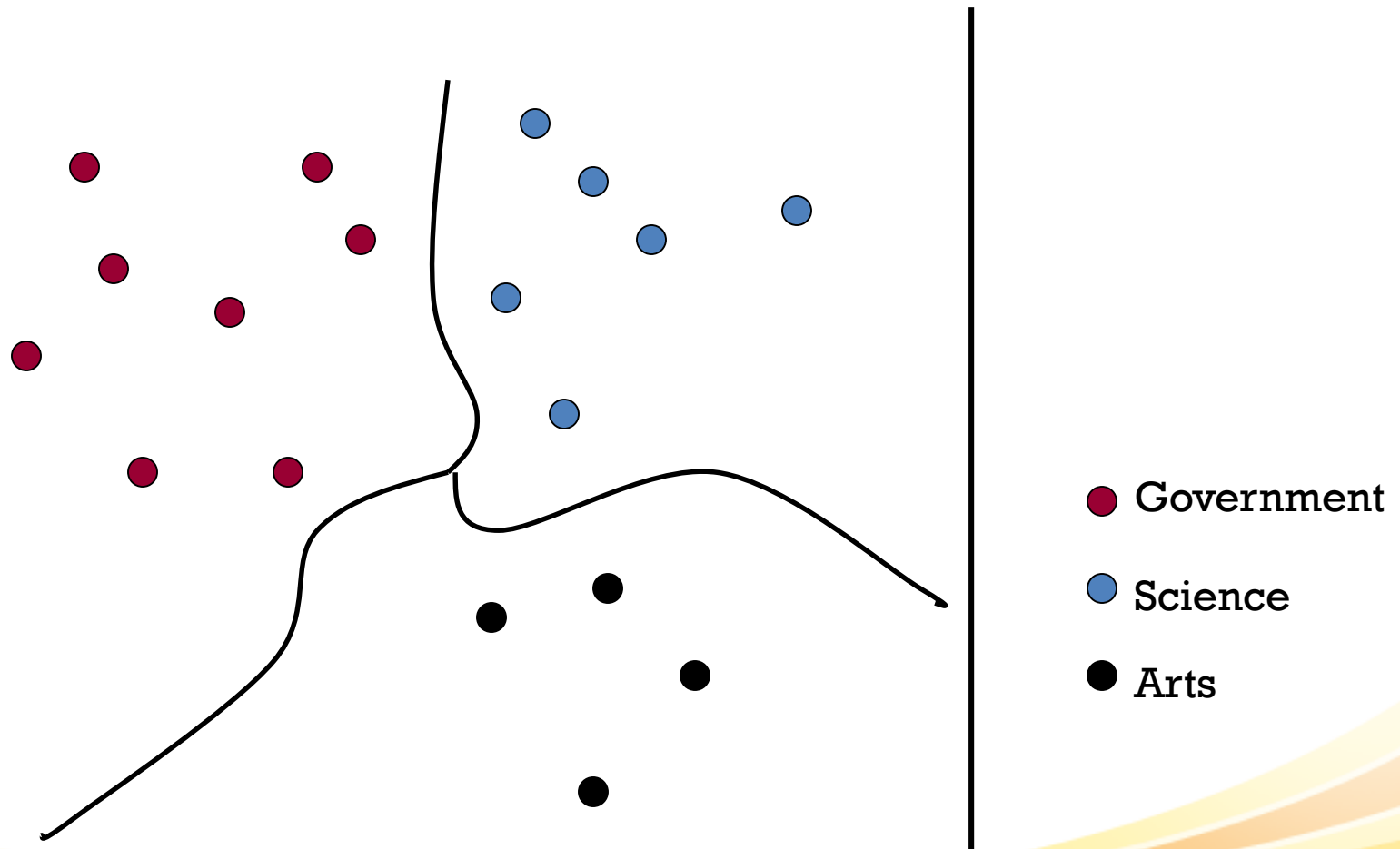
KNN CLASSIFICATION



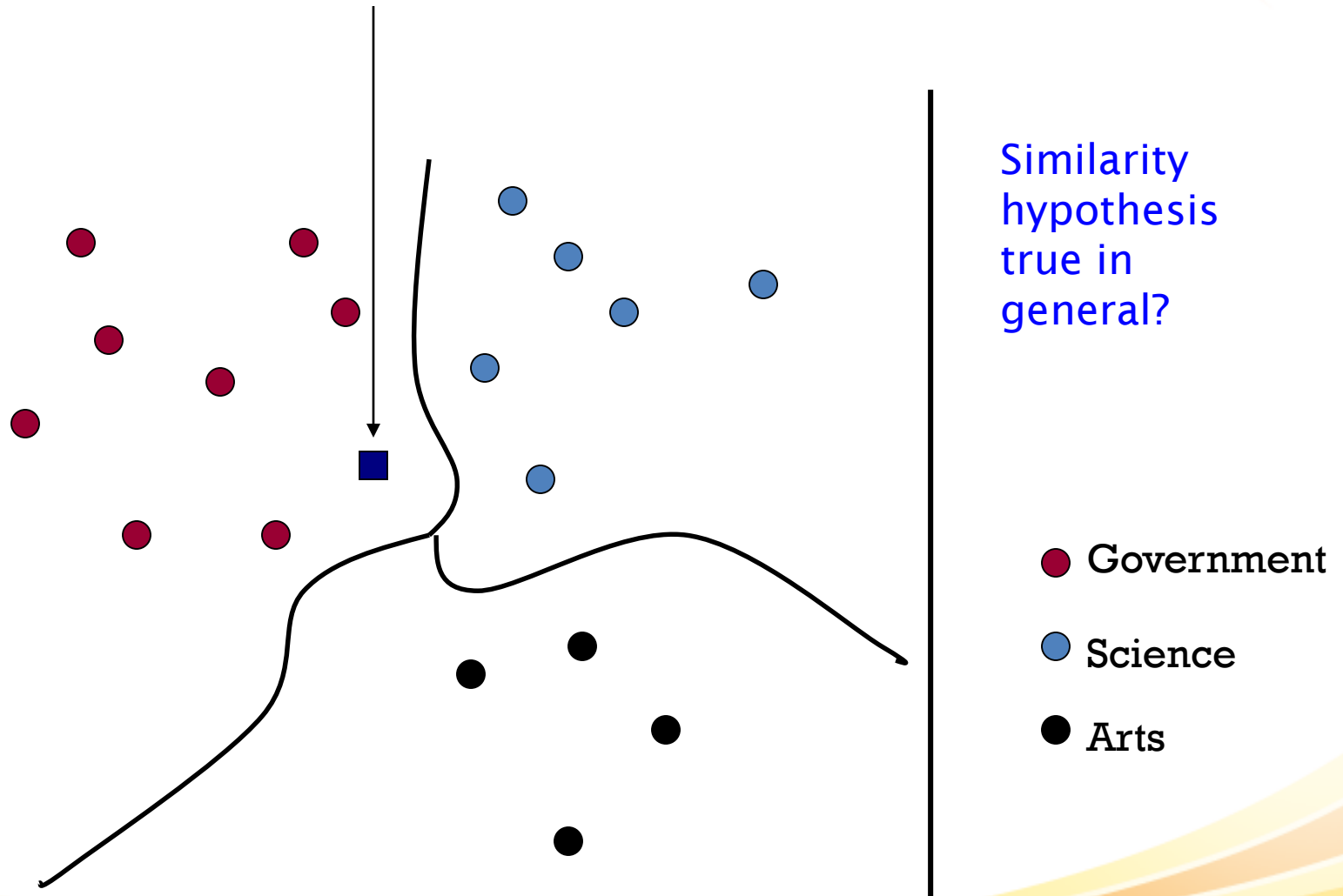
Classification Using Vector Spaces

- Each training doc a point (vector) labeled by its topic (= class)
- Hypothesis: docs of the same class form a contiguous region of space
- We define surfaces to delineate classes in space

Classes in a Vector Space



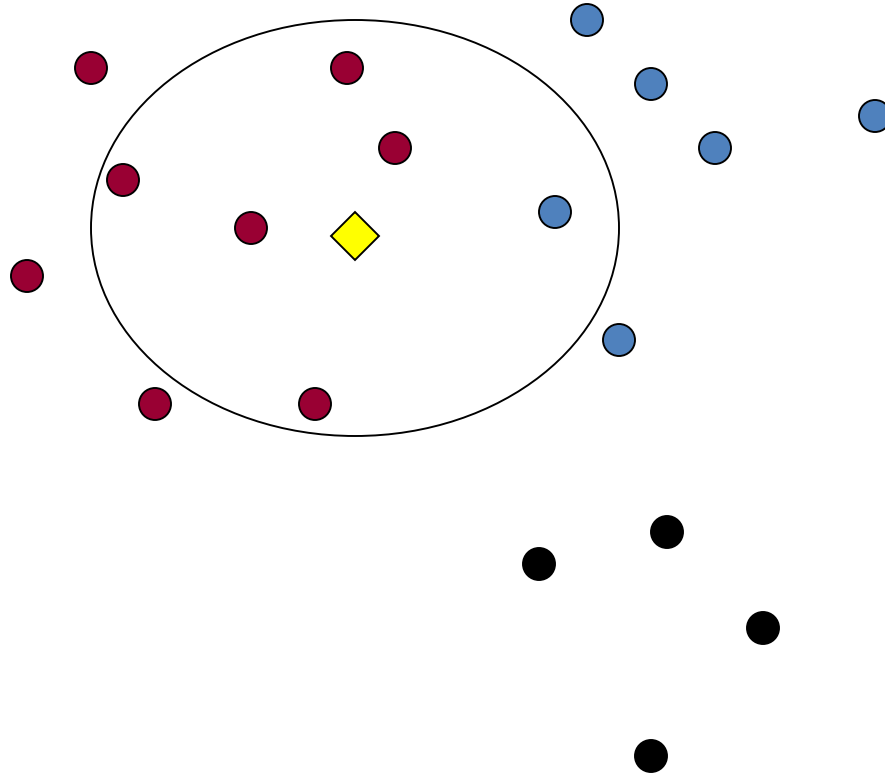
Test Document = Government






k Nearest Neighbor Classification

- To classify document d into class c
- Define k -neighborhood N as k nearest neighbors of d
- Count number of documents i in N that belong to c
- Estimate $P(c|d)$ as i/k
- Choose as class $\operatorname{argmax}_c P(c|d)$ [= majority class]

Example: $k=6$ (6NN)



$P(\text{science} | \text{ })?$ 

-  Government
-  Science
-  Arts

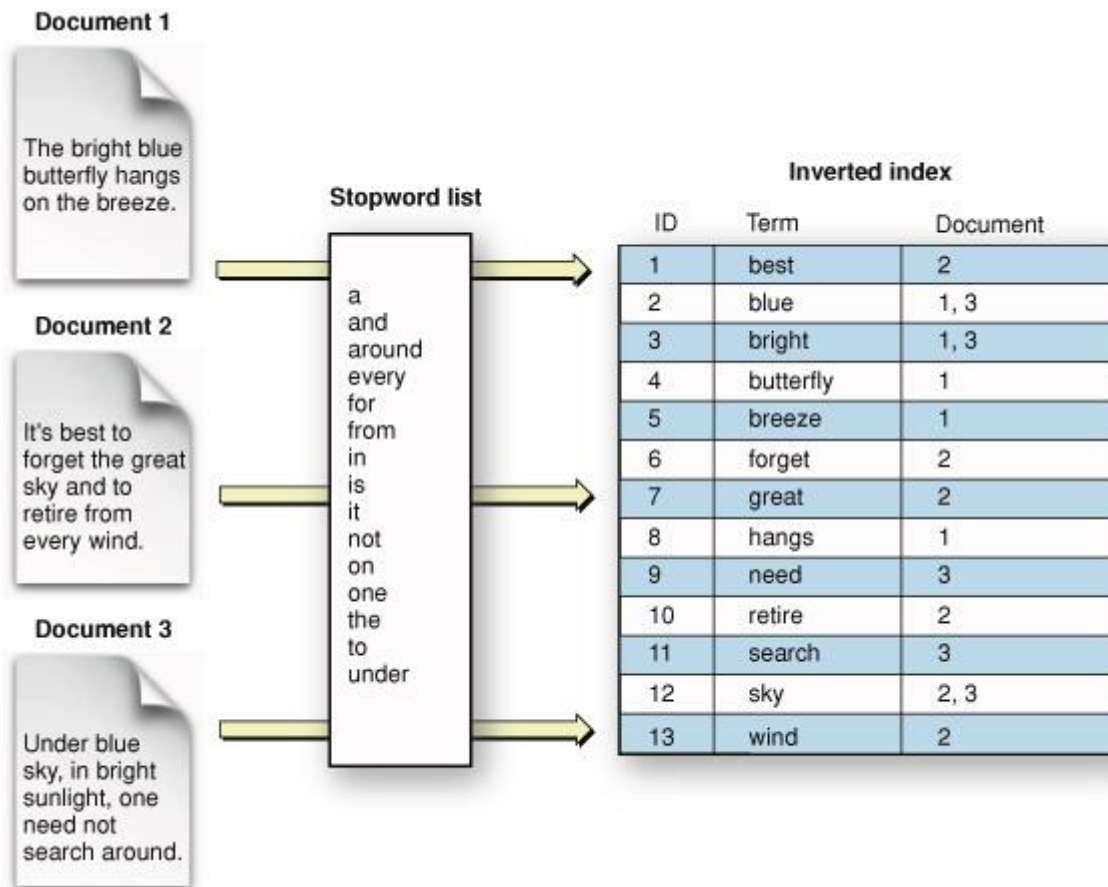
Nearest-Neighbor Learning Algorithm

- Learning is just storing the representations of the training examples in D .
- Testing instance x :
 - Compute similarity between x and all examples in D .
 - Assign x the category of the most similar example in D .
- Does not explicitly compute a generalization or category prototypes.
- Also called:
 - Case-based learning
 - Memory-based learning
 - Lazy learning

Similarity Metrics

- Nearest neighbor method depends on a similarity (or distance) metric.
- Simplest for continuous m -dimensional instance space is *Euclidian distance*.
- Simplest for m -dimensional binary instance space is *Hamming distance* (number of feature values that differ).
- For text, cosine similarity of tf.idf weighted vectors is typically most effective.

Illustration of Inverted Index



Nearest Neighbor with Inverted Index

- Naively finding nearest neighbors requires a linear search through $|D|$ documents in collection
- But determining k nearest neighbors is the same as determining the k best retrievals using the test document as a query to a database of training documents.
- Use standard vector space inverted index methods to find the k nearest neighbors.
- **Testing Time:** $O(B |V_t|)$ where B is the average number of training documents in which a test-document word appears.
 - Typically $B \ll |D|$

kNN: Discussion

- No feature selection necessary
- Scales well with large number of classes
 - Don't need to train n classifiers for n classes
- Classes can influence each other
 - Small changes to one class can have ripple effect
- Scores can be hard to convert to probabilities
- No training necessary
 - Actually: perhaps not true. (Data editing, etc.)



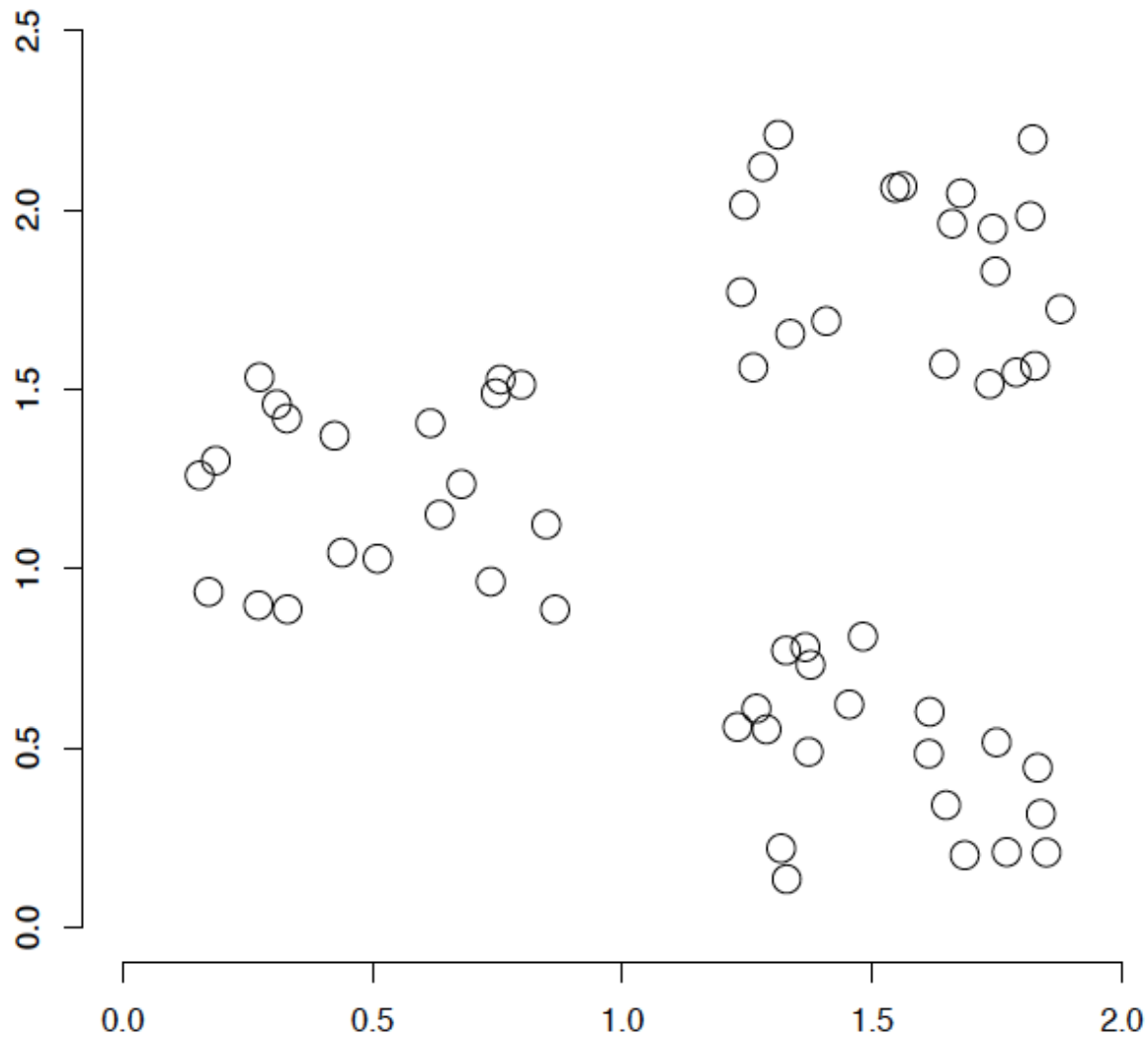
CLUSTERING



What is clustering?

- **Clustering**: the process of grouping a set of objects into classes of similar objects
 - Documents within a cluster should be similar.
 - Documents from different clusters should be dissimilar.
- The commonest form of *unsupervised learning*
 - Unsupervised learning = learning from raw data, as opposed to supervised data where a classification of examples is given
 - A common and important task that finds many applications in IR and other places

A data set with clear cluster structure



- How would you design an algorithm for finding the three clusters in this case?

Citation ranking

scott fahlman - ResearchIndex document query - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://citeseer.ist.psu.edu/cis?q=scott+fahlman&cs=1

CiteSeer Find: Documents Citations

Searching for **PHRASE** scott fahlman.
Restrict to: [Header](#) [Title](#) Order by: [Expected citations](#) [Hubs](#) [Usage](#) [Date](#) Try: [Google \(CiteSeer\)](#) [Google \(Web\)](#) [Yahoo!](#) [MSN](#) [CSB](#) [DBLP](#)
31 documents found. Order: number of citations.

[The Cascade-Correlation Learning Architecture - Fahlman, Lebiere \(1991\)](#) (Correct) (369 citations)
The Cascade-Correlation Learning Architecture **Scott Fahlman**, Christian Lebiere
www.cs.cmu.edu/afs/cs/project/connect/tr/cascor-tr.ps.Z

[Multitask Learning - Caruana \(1997\)](#) (Correct) (38 citations)
Tom Dietterich, Virginia de Sa, Dayne Freitag, **Scott Fahlman**, Ken Lang, Tom Mitchell, Andrew Moore, Dean
reports-archive.adm.cs.cmu.edu/anon/1997/CMU-CS-97-203.ps.Z

[Design and Analysis of a Computational Model of Cooperative... - Potter \(1997\)](#) (Correct) (27 citations)
performance data documented in chapter 6 **Scott Fahlman** at Carnegie Mellon University for use of his
www.cs.gmu.edu/~mpotter/pubs/thesis1.ps.gz

[Speeding Up Backpropagation Algorithms By Using... - Joost, Schiffmann \(1997\)](#) (Correct) (9 citations)
cancels the update values near to zero b **Scott Fahlman** called this effect sigmoide prime factor. 2
spica.fernuni-hagen.de/publikat/IJUFKS.pdf

[Some Notes on Neural Learning Algorithm Benchmarking - Prechelt \(1995\)](#) (Correct) (8 citations)
for their work and comments. Thanks to **Scott Fahlman** for making Proben1 available on his nnbench
www.wipd.ira.uka.de/~prechelt/Biblio/neurocomp95.ps.gz

[Reinforcement Learning Through Gradient Descent - Baird, III \(1999\)](#) (Correct) (7 citations)
committee: Andrew Moore (chair) Tom Mitchell **Scott Fahlman** Leslie Kaelbling, Brown University Copyright
I thank Leslie Kaelbling, Tom Mitchell, and **Scott Fahlman** for agreeing to be on my committee, and for
www.ri.cmu.edu/pub_files/pub2/baird_leemon_1999_1/baird_leemon_1999_1.ps.gz

[Experiments with the Cascade-Correlation Algorithm - Yang, Honavar \(1991\)](#) (Correct) (4 citations)
of TR 91-16 4 The authors wish to thank Dr. **Scott Fahlman** for his feedback on an early draft of this
ftp.cis.ohio-state.edu/pub/neuroprose/yang.cascor.ps.Z

[Speech Recognition using Neural Networks - Tebelskis \(1995\)](#) (Correct) (3 citations)
final dissertation. I would also like to thank **Scott Fahlman**, my first advisor, for channeling my early
<http://citeseer.ist.psu.edu/context/8437/3769>

Automata

Citation graph browsing

Citations: The cascade-correlation learning architecture - Fahlman, Lebiere (ResearchIndex) - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://citeseer.ist.psu.edu/cs?q=dbnum%3D1%2CGID%3D8437%2CDID%3D3769%2Cstart%3D50%2Cdl Google

372 citations found. Retrieving documents...

C. Fahlman, C. Lebiere, *The cascade-correlation learning architecture*, in: D. Turetzky (Ed.), *Advances in Neural Information Processing Systems*, Vol. 2, Morgan Kaufmann, Los Altos, 1990, pp. 524-532.

CiteSeer [Home/Search](#) [Document Details and Download](#) [Summary](#) [Related Articles](#) [Check](#)

This paper is cited in the following contexts:

[Documents 51 to 100](#) [Previous 50](#) [Next 50](#)

[Evolving Fuzzy Neural Networks for Supervised/Unsupervised... - Kasabov \(2001\)](#) (1 citation) (Correct)

....above and that have influenced the development of EFuNNs. **These are methods and systems for: adaptive [6,7,8,9,19,53,58,61,71] learning [4,5,7,8,14,30,46,47,48] incremental lifelong learning [69,35,36,82] on line [17,21,22,28,31,35,36,42,44,61,66,67,69] constructivist structural [15,19,11,14,9] that is supported by biological facts [14,62,73,77, 82] selectivist structural learning [26,29,49,56,59,64,50,32] hybrid constructivist selectivist structural learning 2 [52,66,70,31] knowledge based learning neural networks (KBNN) 57,24,25,30,33,38,44, 45,51,63,76,77,83] The EFuNN model**

Fahlman, C., and C. Lebiere, "The Cascade- Correlation Learning Architecture", in: Turetzky, D (ed) *Advances in Neural Information Processing Systems*, vol. 2, Morgan Kaufmann, 524-532 (1990).

[A Dynamic Neural Network for Continual - Classification Lang Warwick](#) (Correct)

....mistaken for dynamic neural networks as they can be described as non linear, dynamic system in state space [5] Hopfield networks are not truly dynamic, as the network has no provision for perpetual novelty. **Incremental Networks are an example of the how the network is dynamic during training [2].** Pruning removes neurons and links during the network s operation. **Although** these algorithms supply efficient network architectures, they are not purely dynamic as neither the topology nor the knowledge within the network changes once the network has finished learning. **In trajectory**

Fahlman S. E., Lebiere C. "The Cascade-Correlation Learning Architecture." Technical Report CMU-CS-90-100, School of Computer Science, Carnegie Mellon University, Pittsburg, PA. 1990.

[A Future for Dynamic Neural Networks - Lang \(2000\)](#) (Correct)

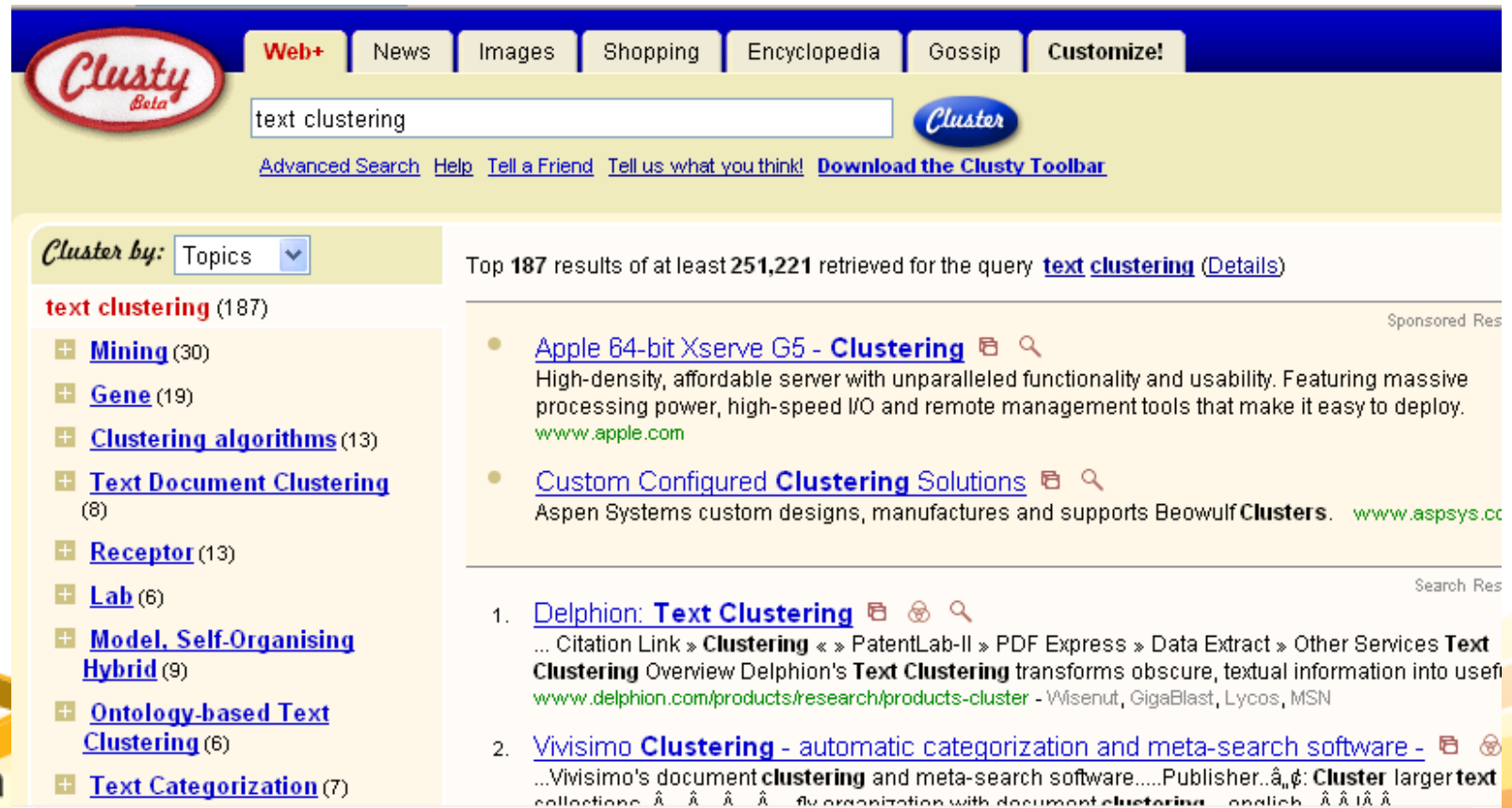
....of the network, it can be constructed during learning and then reduced later until an optimum solution is gained. **Constructive methods (also called growing) start with an input and output layer and add hidden nodes or weights (links) during learning until the network can represent the function [4,5,6].** Reduction (also known as pruning) removes superfluous parts of the network, while still representing the object function [7,8] Incremental networks are useful as they provide the user with less parameters to decide upon before learning has been done. **Classically**, the number of nodes the

(most popular) alternative is to arbitrarily decide on a few topologies to try and then select the topology that performed the best according to predecided criteria. **By far**

Done Automata

Clustering: Navigation of search results

- For grouping search results thematically
 - clusty.com / Vivisimo



The screenshot displays the Clusty Beta search engine interface. At the top, there is a navigation bar with tabs for Web+, News, Images, Shopping, Encyclopedia, Gossip, and Customize!. Below this is a search bar containing the text 'text clustering'. To the right of the search bar is a 'Cluster' button. Below the search bar are links for 'Advanced Search', 'Help', 'Tell a Friend', 'Tell us what you think!', and 'Download the Clusty Toolbar'.

On the left side, there is a 'Cluster by:' dropdown menu set to 'Topics'. Below this is a list of clusters for the query 'text clustering' (187 results):

- [Mining](#) (30)
- [Gene](#) (19)
- [Clustering algorithms](#) (13)
- [Text Document Clustering](#) (8)
- [Receptor](#) (13)
- [Lab](#) (6)
- [Model, Self-Organising Hybrid](#) (9)
- [Ontology-based Text Clustering](#) (6)
- [Text Categorization](#) (7)

On the right side, the main search results area shows 'Top 187 results of at least 251,221 retrieved for the query [text clustering](#) (Details)'. The results are listed in two columns. The first column contains two sponsored results:

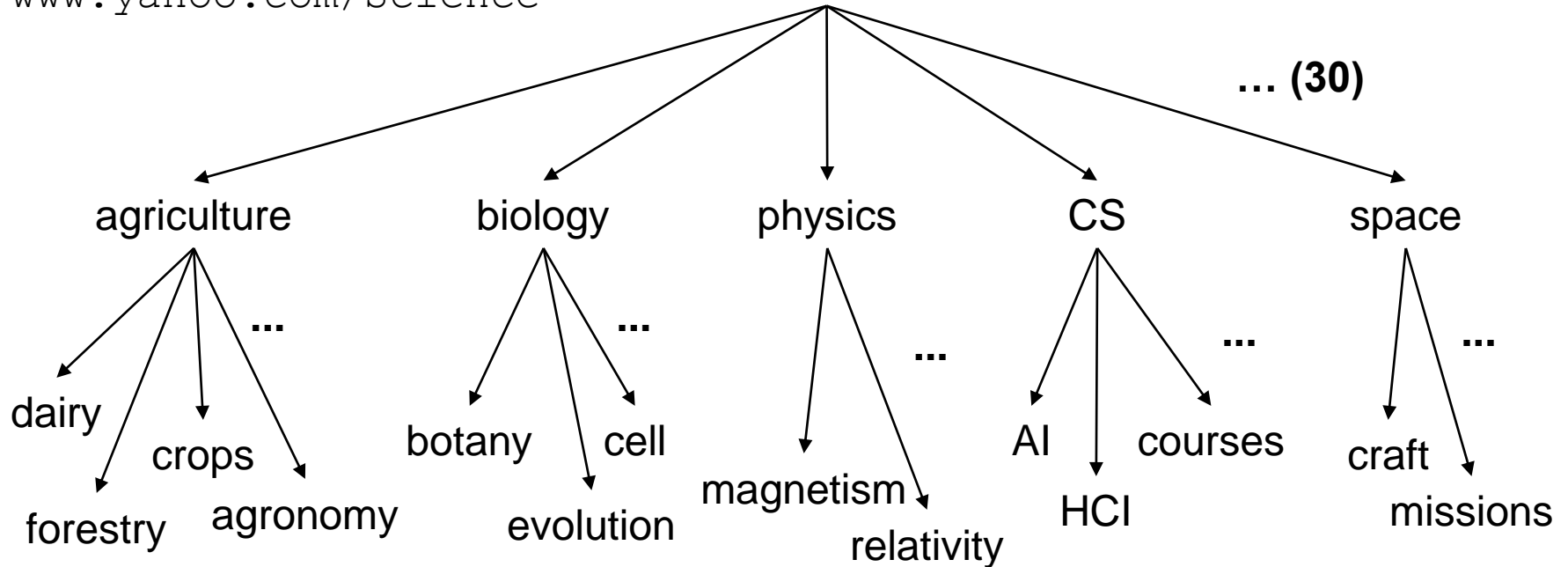
- [Apple 64-bit Xserve G5 - Clustering](#) (Sponsored Res)
High-density, affordable server with unparalleled functionality and usability. Featuring massive processing power, high-speed I/O and remote management tools that make it easy to deploy. [www.apple.com](#)
- [Custom Configured Clustering Solutions](#)
Aspen Systems custom designs, manufactures and supports Beowulf Clusters. [www.aspsys.cc](#)

The second column contains two organic search results:

- [Delphion: Text Clustering](#)
... Citation Link » **Clustering** « » PatentLab-II » PDF Express » Data Extract » Other Services **Text Clustering** Overview Delphion's **Text Clustering** transforms obscure, textual information into usefu [www.delphion.com/products/research/products-cluster](#) - Wisenut, GigaBlast, Lycos, MSN
- [Vivisimo Clustering - automatic categorization and meta-search software -](#)
...Vivisimo's document **clustering** and meta-search software.....Publisher..â„¢: **Cluster** larger text collections. â„¢ â„¢ â„¢ â„¢ fly organization with document **clustering**... english. â„¢ â„¢ â„¢

Clustering: Corpus browsing

www.yahoo.com/Science

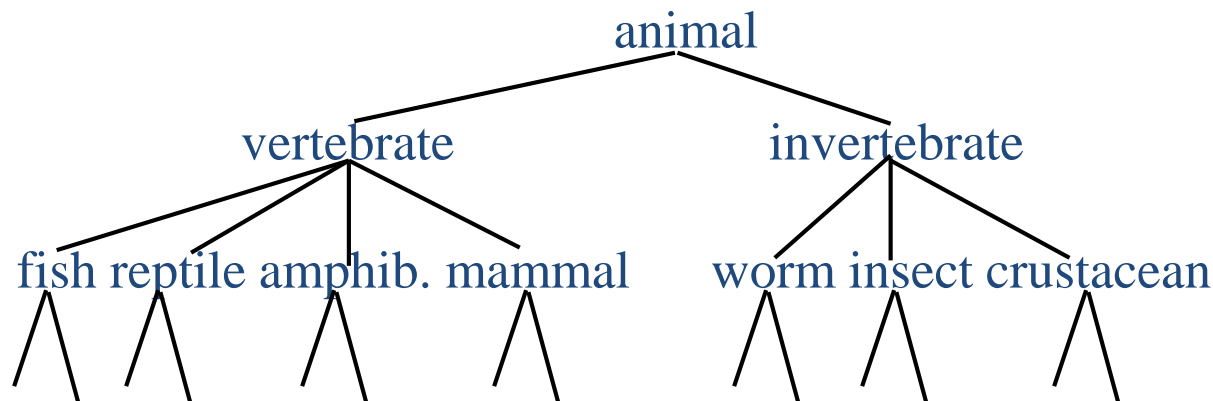


Clustering considerations

- What does it mean for objects to be similar?
- What algorithm and approach do we take?
 - Top-down: k-means
 - Bottom-up: hierarchical agglomerative clustering
- Do we need a hierarchical arrangement of clusters?
- How many clusters?
- Can we label or name the clusters?

Hierarchical Clustering

- Build a tree-based hierarchical taxonomy (*dendrogram*) from a set of documents.

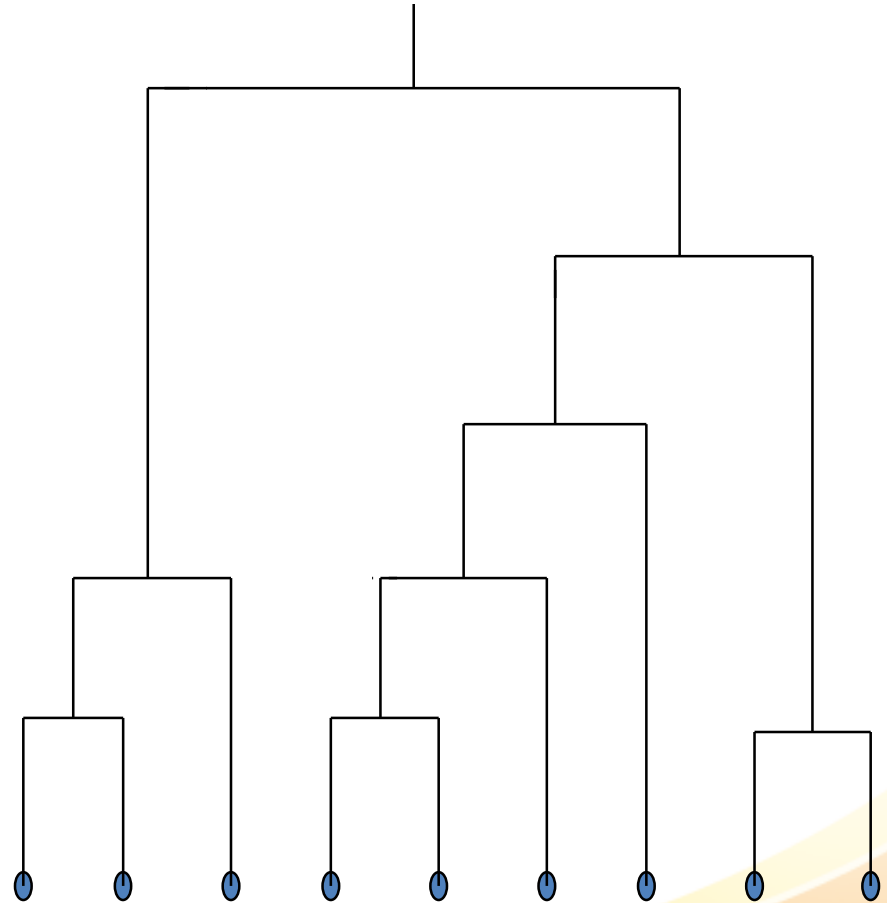


- One approach: recursive application of a partitional clustering algorithm.



Dendrogram: Hierarchical Clustering

- Clustering obtained by cutting the dendrogram at a desired level: each connected component forms a cluster.



Hierarchical Agglomerative Clustering (HAC)

- Starts with each doc in a separate cluster
 - then repeatedly joins the closest pair of clusters, until there is only one cluster.
- The history of merging forms a binary tree or hierarchy.

Note: the resulting clusters are still “hard” and induce a partition

Closest pair of clusters

- Many variants to defining closest pair of clusters
- **Single-link**
 - Similarity of the *most* cosine-similar (single-link)
- **Complete-link**
 - Similarity of the “furthest” points, the *least* cosine-similar
- **Centroid**
 - Clusters whose centroids (centers of gravity) are the most cosine-similar
- **Average-link**
 - Average cosine between pairs of elements

Single Link Agglomerative Clustering

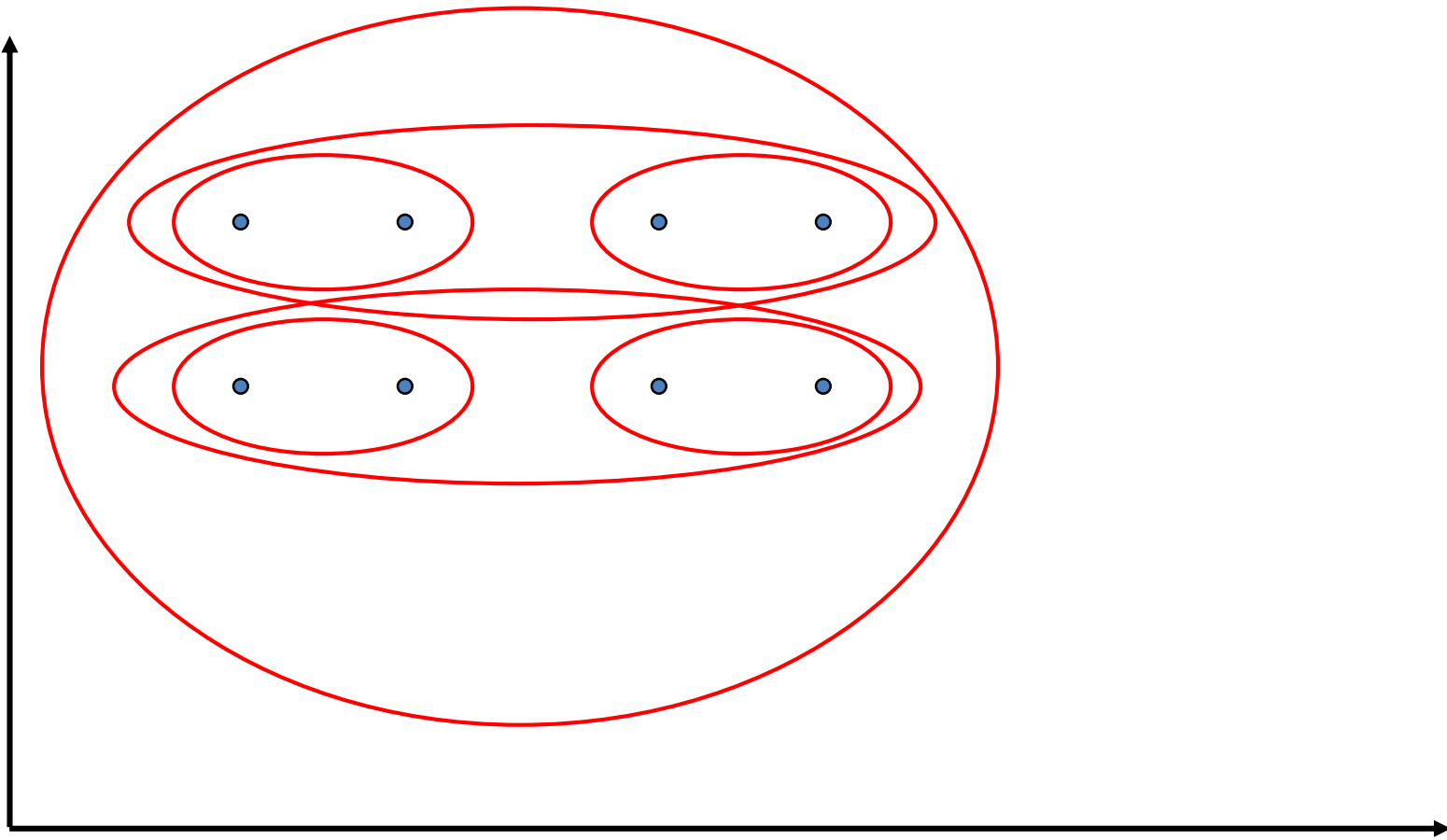
- Use maximum similarity of pairs:

$$\textit{sim}(c_i, c_j) = \max_{x \in c_i, y \in c_j} \textit{sim}(x, y)$$

- Can result in “straggly” (long and thin) clusters due to chaining effect.
- After merging c_i and c_j , the similarity of the resulting cluster to another cluster, c_k , is:

$$\textit{sim}((c_i \cup c_j), c_k) = \max(\textit{sim}(c_i, c_k), \textit{sim}(c_j, c_k))$$

Single Link Example



Complete Link

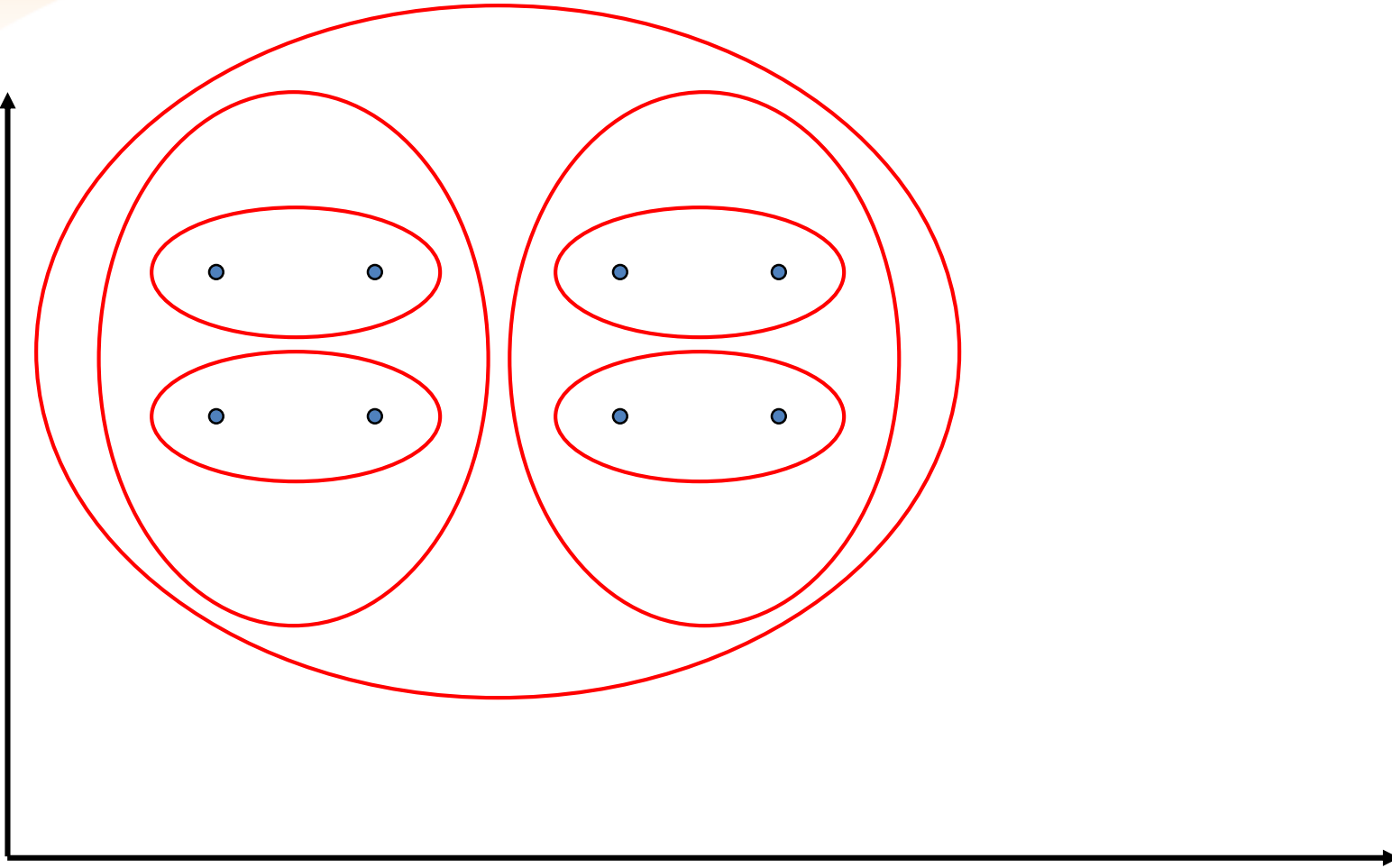
- Use minimum similarity of pairs:

$$\textit{sim}(c_i, c_j) = \min_{x \in c_i, y \in c_j} \textit{sim}(x, y)$$

- Makes “tighter,” spherical clusters that are typically preferable.
- After merging c_i and c_j , the similarity of the resulting cluster to another cluster, c_k , is:

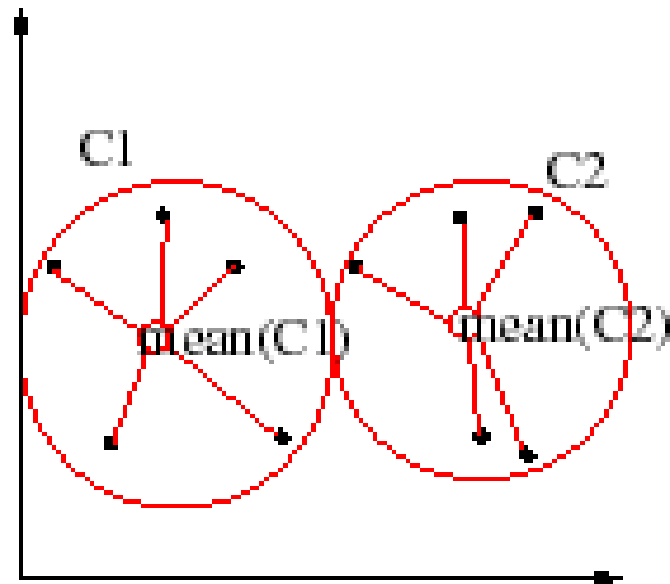
$$\textit{sim}((c_i \cup c_j), c_k) = \min(\textit{sim}(c_i, c_k), \textit{sim}(c_j, c_k))$$

Complete Link Example



K-means and K-medoids algorithms

- Objective function:
Minimize the sum of square distances of points to a *cluster representative (centroid)*
- Efficient iterative algorithms ($O(n)$)



K-Means Clustering

1. Select K seed centroids s.t. $d(c_i, c_j) > d_{min}$
2. Assign points to clusters by minimum distance to centroid

$$Cluster(\vec{p}_i) = \underset{1 \leq j \leq K}{\operatorname{Argmin}} d(\vec{p}_i, \vec{c}_j)$$

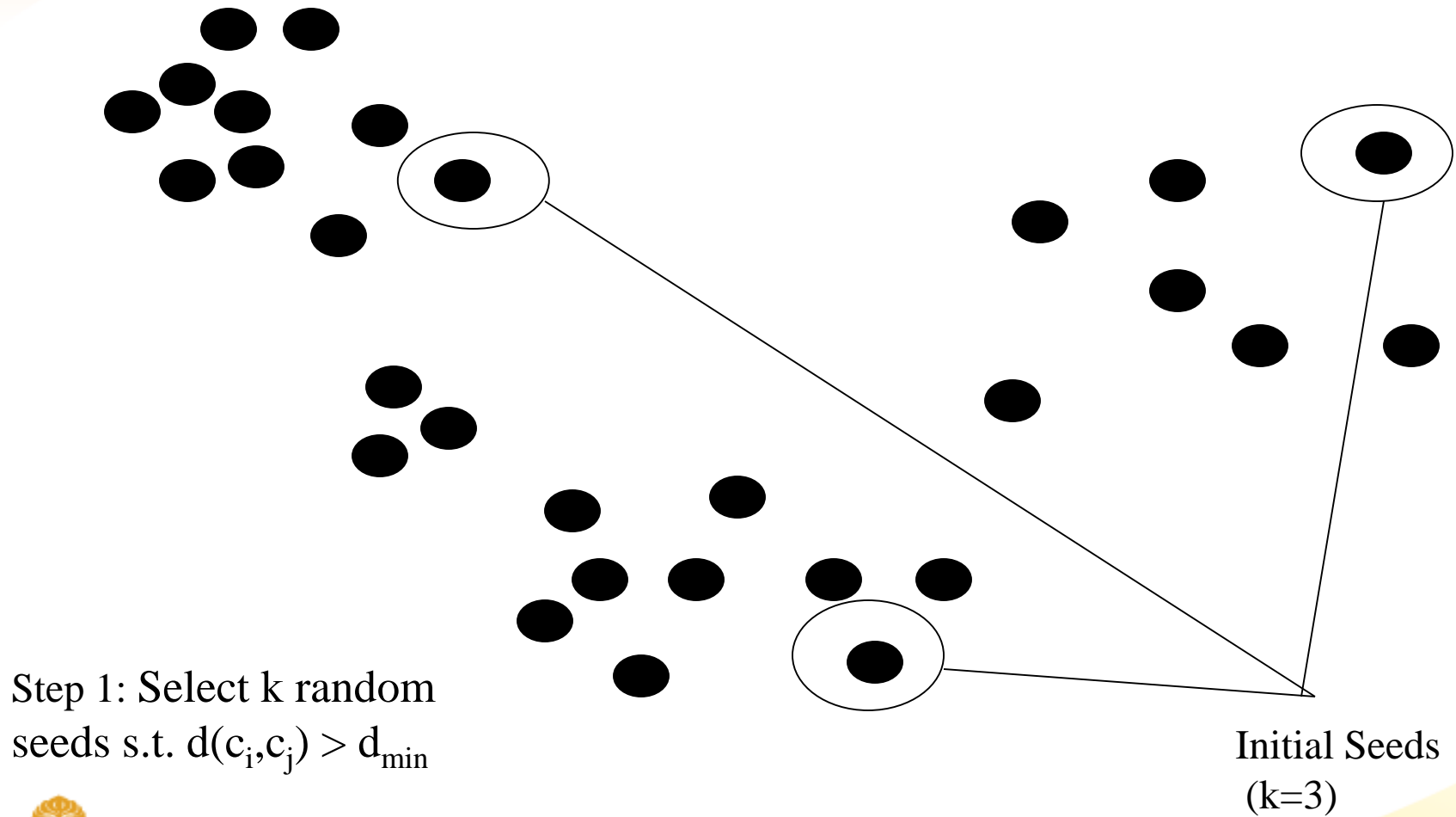
3. Compute new cluster centroids:

$$\vec{c}_j = \frac{1}{n_{Cluster(\vec{p}_i)=j}} \sum \vec{p}_i$$

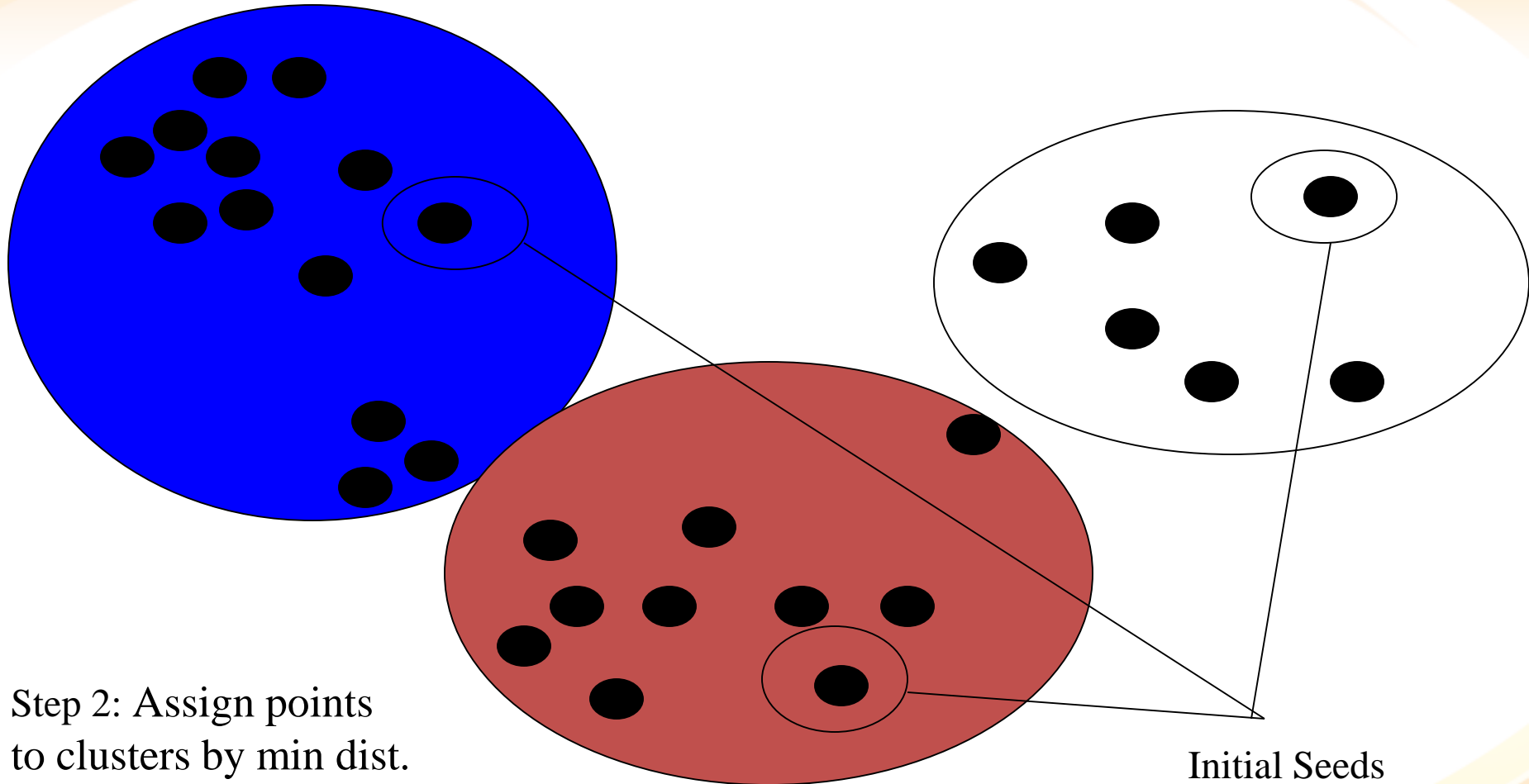
4. Iterate steps 2 & 3 until no points change clusters



K-Means Clustering: Initial Data Points

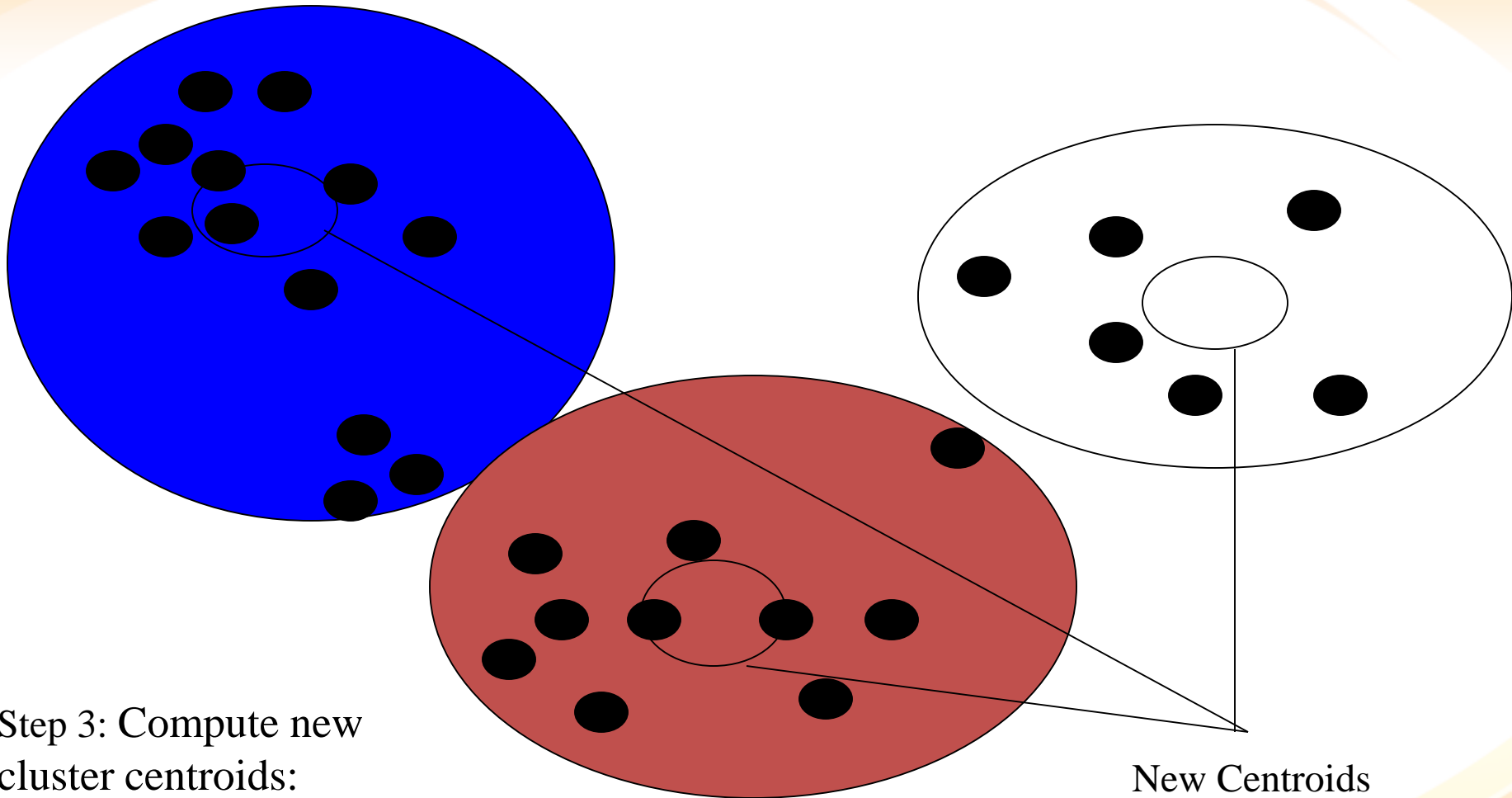


K-Means Clustering: First-Pass Clusters



$$Cluster(\vec{p}_i) = \underset{1 \leq j \leq K}{\operatorname{Argmin}} d(\vec{p}_i, \vec{c}_j)$$

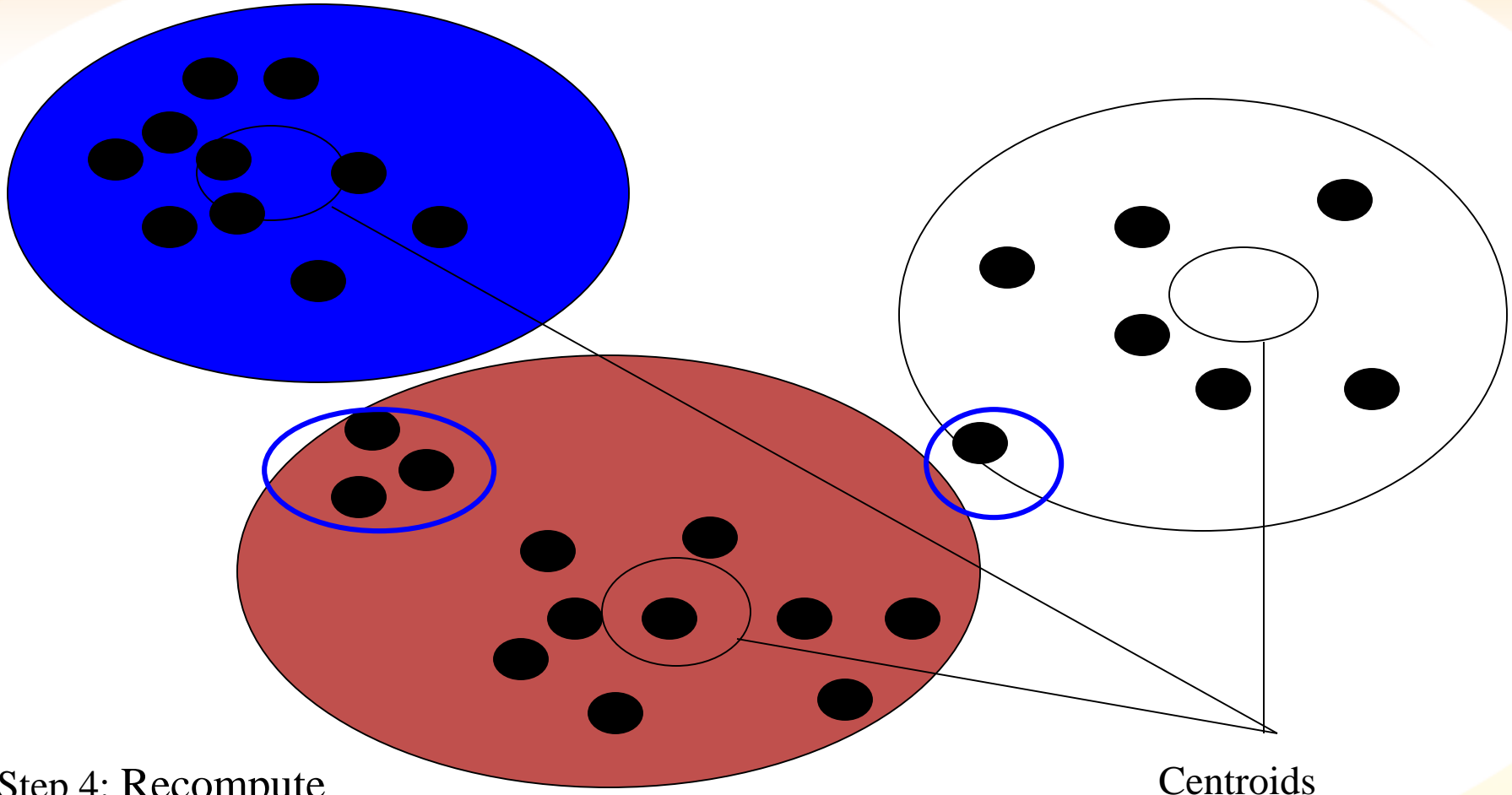
K-Means Clustering: Seeds \rightarrow Centroids



Step 3: Compute new cluster centroids:

$$\vec{c}_j = \frac{1}{n_{Cluster(\vec{p}_i)=j}} \sum \vec{p}_i$$

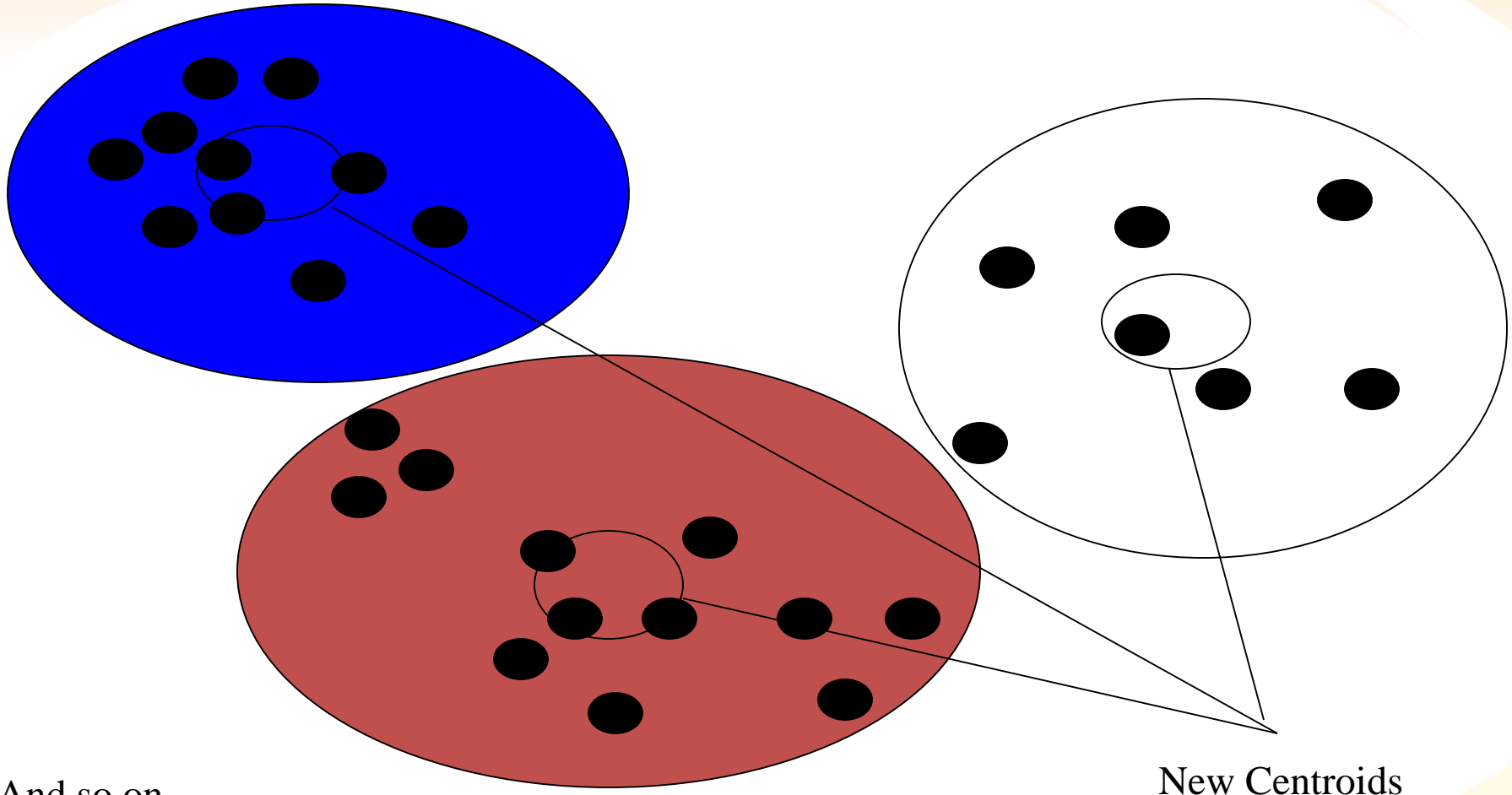
K-Means Clustering: Second Pass Clusters



Step 4: Recompute

$$Cluster(\vec{p}_i) = \underset{1 \leq j \leq K}{\operatorname{Argmin}} d(\vec{p}_i, \vec{c}_j)$$

K-Means Clustering: Iterate Until Stability



And so on.

Major issue - labeling

- After clustering algorithm finds clusters - how can they be useful to the end user?
- Need pithy label for each cluster
 - In search results, say “Animal” or “Car” in the *jaguar* example.
 - In topic trees, need navigational cues.
 - Often done by hand, a posteriori.

How would you do this?

How to Label Clusters

- Show titles of typical documents
 - Titles are easy to scan
 - Authors create them for quick scanning!
 - But you can only show a few titles which may not fully represent cluster
- Show words/phrases prominent in cluster
 - More likely to fully represent cluster
 - Use distinguishing words/phrases
 - Differential labeling
 - But harder to scan

Labeling

- Common heuristics - list 5-10 most frequent terms in the centroid vector.
 - Drop stop-words; stem.
- Differential labeling by frequent terms
 - Within a collection “Computers”, clusters all have the word ***computer*** as frequent term.
 - Discriminant analysis of centroids.
- Perhaps better: distinctive noun phrase

What Is A Good Clustering?

- Internal criterion: A good clustering will produce high quality clusters in which:
 - the intra-class (that is, intra-cluster) similarity is high
 - the inter-class similarity is low
 - The measured quality of a clustering depends on both the document representation and the similarity measure used

External criteria for clustering quality

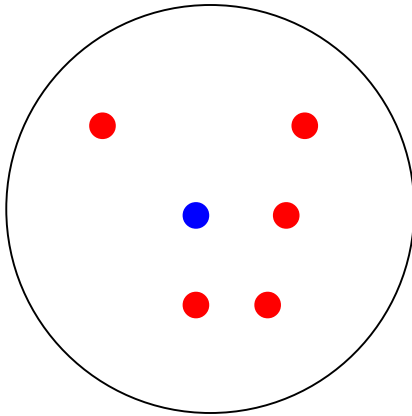
- Quality measured by its ability to discover some or all of the hidden patterns or latent classes in gold standard data
- Assesses a clustering with respect to ground truth ... requires *labeled data*
- Assume documents with C gold standard classes, while our clustering algorithms produce K clusters, $\omega_1, \omega_2, \dots, \omega_K$ with n_i members.

External Evaluation of Cluster Quality

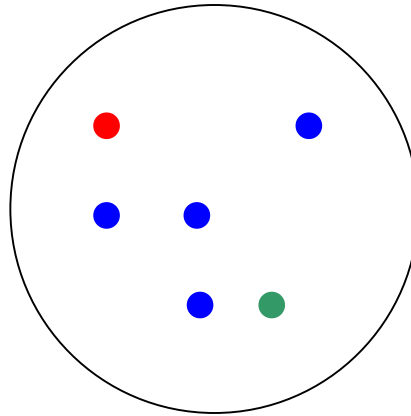
- Simple measure: purity, the ratio between the dominant class in the cluster π_i and the size of cluster ω_i

$$Purity(\omega_i) = \frac{1}{n_i} \max_j (n_{ij}) \quad j \in C$$

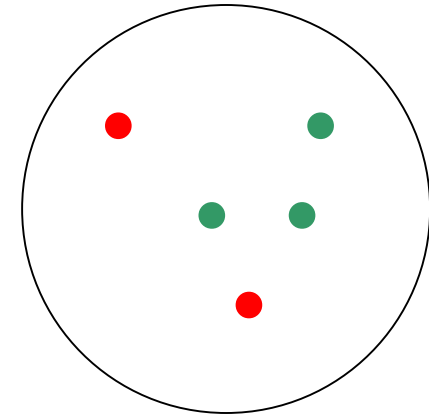
Purity example



Cluster I



Cluster II



Cluster III

Cluster I: Purity = $1/6 (\max(5, 1, 0)) = 5/6$

Cluster II: Purity = $1/6 (\max(1, 4, 1)) = 4/6$

Cluster III: Purity = $1/5 (\max(2, 0, 3)) = 3/5$

Overall: Purity = $1/7 (5+4+3) = 0.71$

Rand Index measures between pair decisions. Here $RI = 0.68$

Number of points	Same Cluster in clustering	Different Clusters in clustering
Same class in ground truth	TP 20	FN 24
Different classes in ground truth	FP 20	FN 72

Rand Index

- We first compute TP + FP The three clusters contain 6, 6, and 5 points, respectively, so the total number of "positives" or pairs of documents that are in the same cluster is:
 - $TP + FP = C(6,2) + C(6,2) + C(5,2) = 40$
- The red pairs in cluster 1, the blue pairs in cluster 2, the green pairs in cluster 3, and the red pair in cluster 3 are true positives:
 - $TP = C(5,2) + C(4,2) + C(3,2) + C(2,2) = 20$
 - $FP = 40 - 20 = 20$
- FN: 5 [pair] (red C1 & C2) + 10 (red C1 & C3) + 2 (red C2 & C3) + 4 (blue C1 & C2) + 3 (green C2 & C3) = 24
- All pair: $N \times (N-1) / 2 = 17 \times 16 / 2 = 136$
- $TN = \text{All pair} - (TP + FP + FN) = 72$

Rand index and Cluster F-measure

$$RI = \frac{A + D}{A + B + C + D}$$

Compare with standard Precision and Recall:

$$P = \frac{A}{A + B}$$

$$R = \frac{A}{A + C}$$

People also define and use a cluster F-measure, which is probably a better measure.

Thank you

