Clustering & Topic Modeling

Text Mining, Transforming Text into Knowledge

Ayoub Bagheri

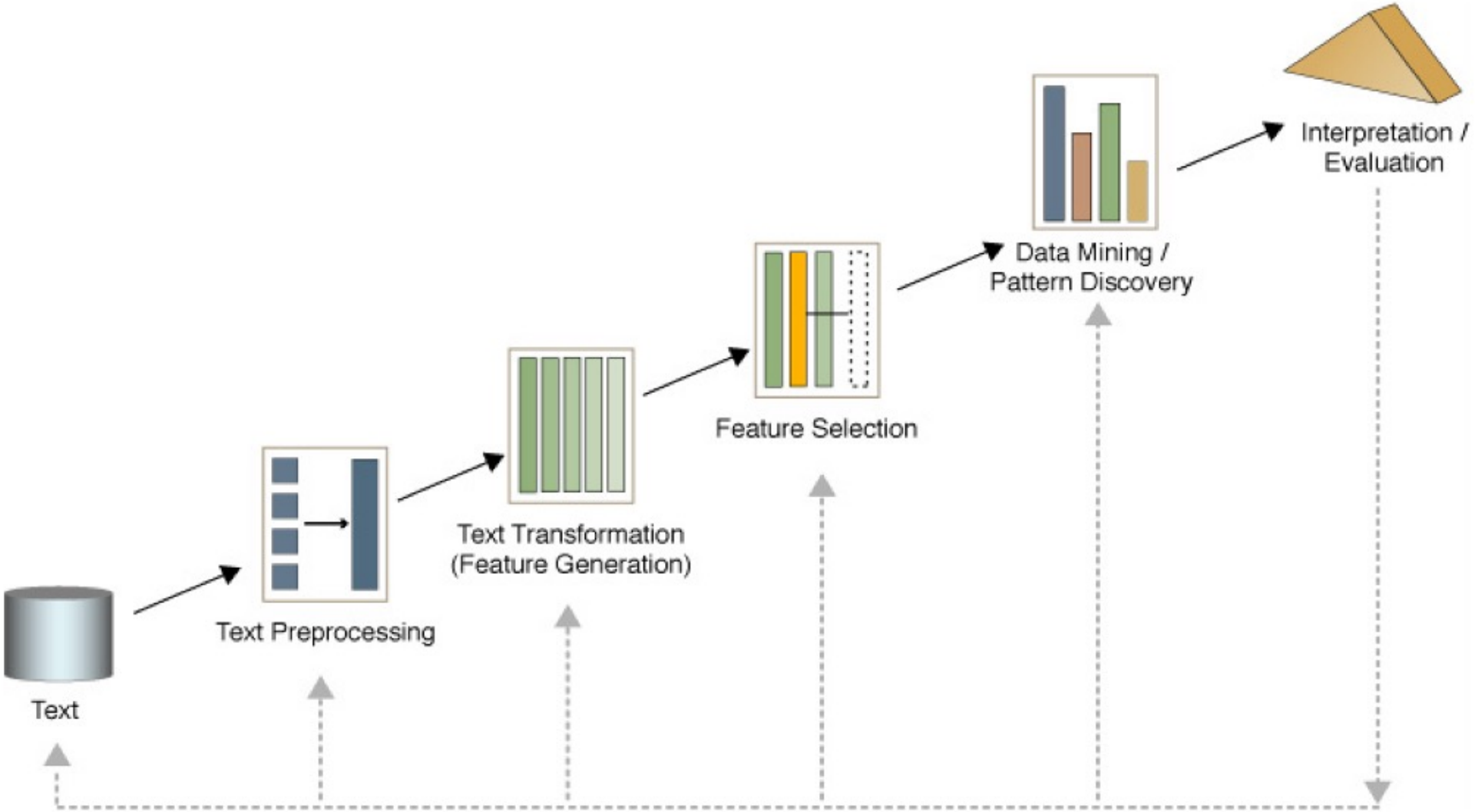
Last week

- Feature selection
- Different methods
- Feature reduction

Today

- Text clustering
- Topic modeling
- Evaluation

Text mining process



Text mining process

- **Data: Text**
- **Text Preprocessing:** is the process of cleaning, normalizing, and structuring raw text data into a format suitable for analysis or input into NLP models. (week 2)
- **Text transformation, feature generation**: involves converting text data into a different format or structure, such as numerical vectors or simplified forms, to make it suitable for analysis or modeling. (weeks 1, 2, 3, 6, 7, 8)
- **Feature selection**: is the process of identifying and selecting the most relevant features from a dataset to improve model performance and reduce complexity. (week 4)
- **Data mining, pattern discovery**: is the process of extracting meaningful patterns and knowledge from text. (weeks 3, 5, 7, 8, 9)
- **Interpretation / Evaluation:** is the process of understanding and explaining the model and patterns / is the assessment process to measure performance and quality. (weeks 3-9)

Text Clustering

Clustering

Find subgroups (clusters) of similar examples in a dataset.

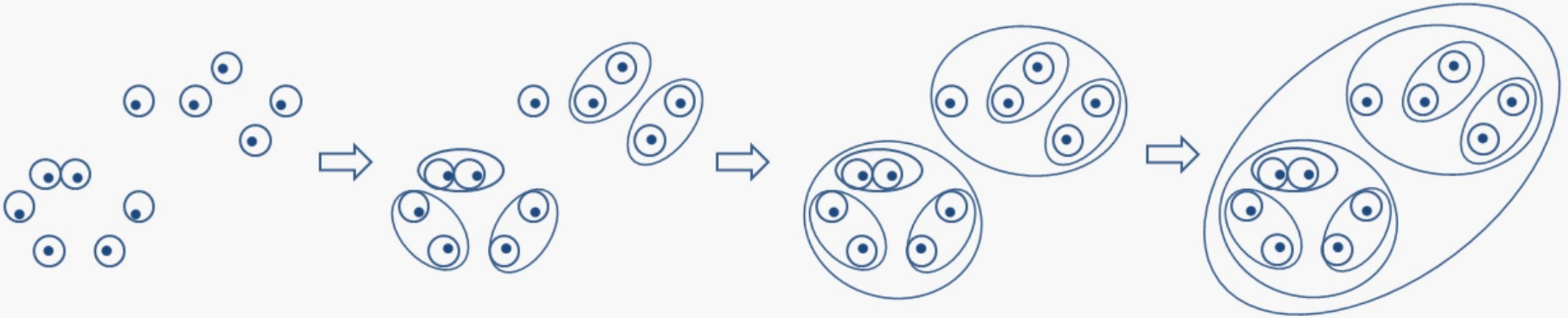
Hierarchical clustering

Hierarchical clustering

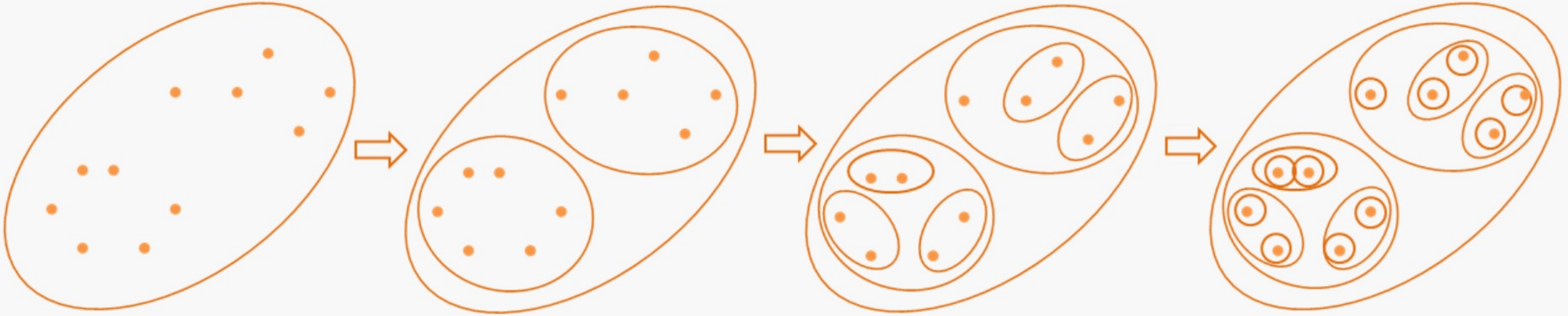
- Bottom-up agglomerative clustering
  - For each observation, compute the distance to all other observations
  - Assign all examples to their individual cluster
  - Combine most similar cluster
  - Keep combining clusters until there is only one cluster left
  - Select number of clusters for the final solution

Divisive: start with one and keep splitting most different

Agglomerative Hierarchical Clustering



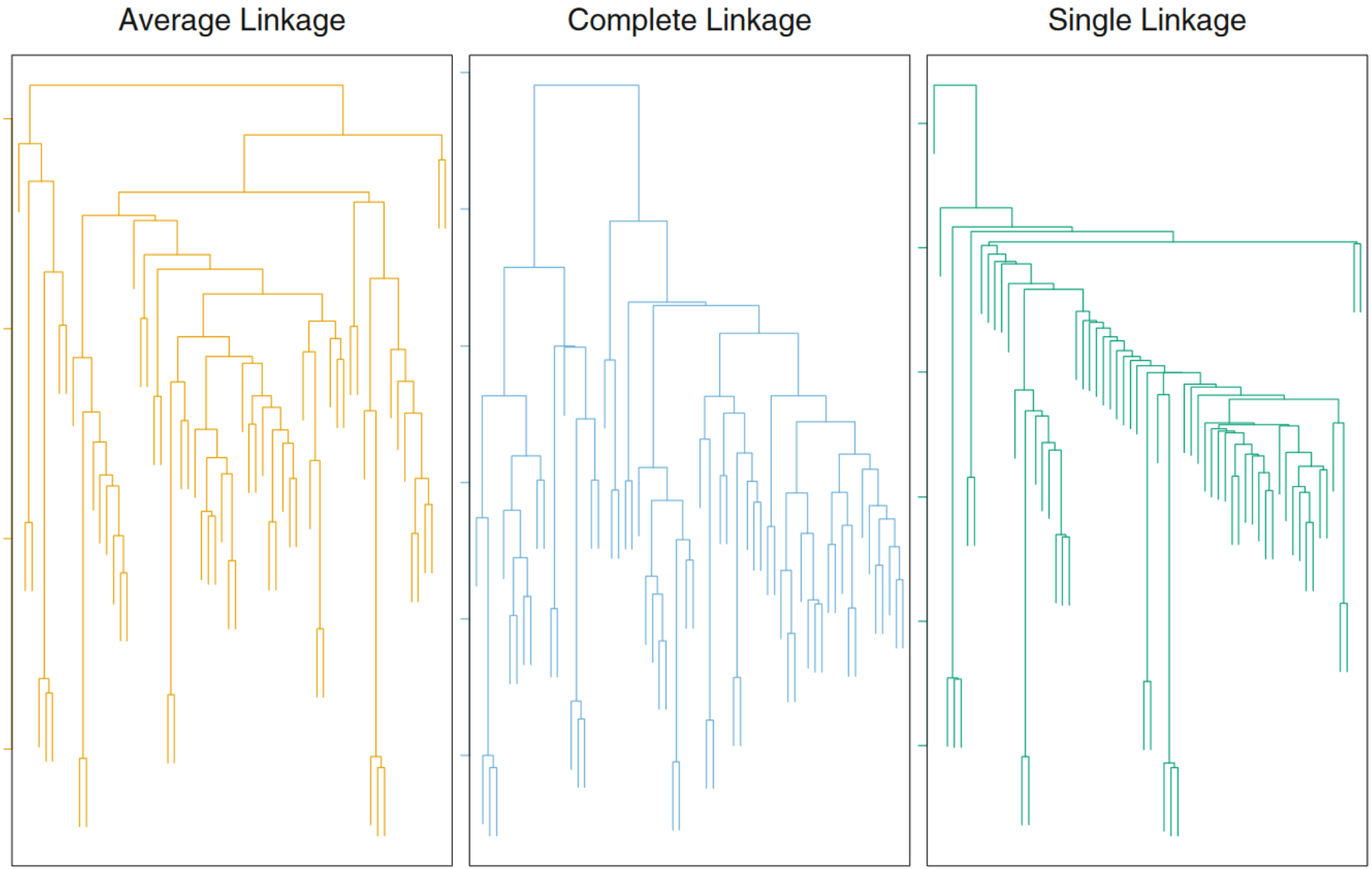
Divisive Hierarchical Clustering



https://quantdare.com/hierarchical-clustering/

Linkage	*Description*
Complete	Maximal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities. Single linkage can result in extended, trailing clusters in which single observations are fused one-at-a-time.
Average	Mean intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

TABLE 12.3. *A summary of the four most commonly-used types of linkage in hierarchical clustering.*


Average Linkage
Complete Linkage
Single Linkage

K-means clustering

K-means clustering

- Predefine the number of clusters (K)
- Apply an algorithm to assign observations to clusters
- Top-down method

K-means clustering algorithm

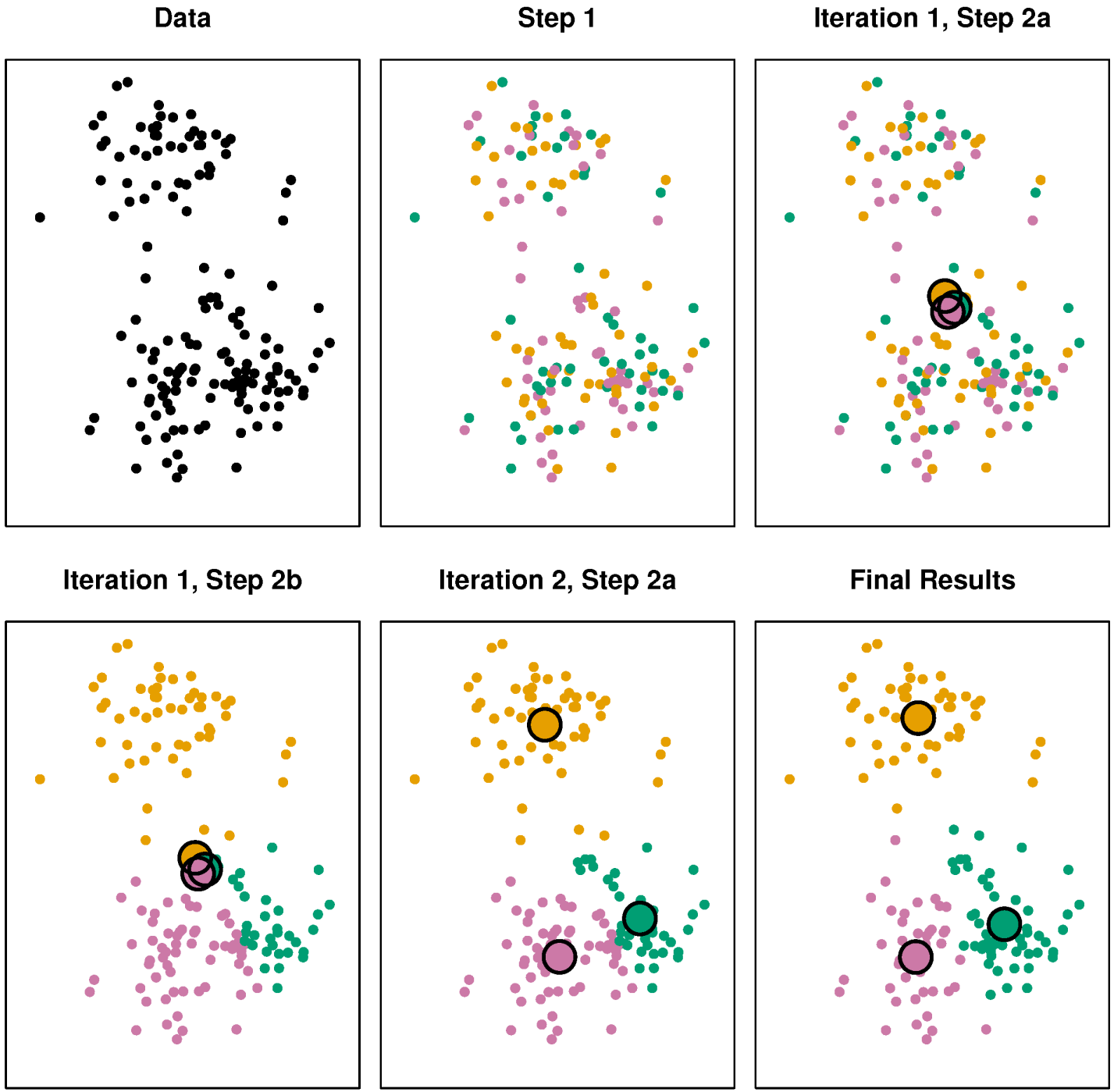
1. Randomly assign examples to K clusters

2a. Calculate the centroid (per-feature mean) for each cluster

2b. Assign each example to the cluster belonging to its closest centroid

3. If the assignments changed, go to step 2a, else stop

K-means clustering



ISLR2 fig 12.8

K-means clustering

- K and the distance metric are hyperparameters
- Distance metric can be anything, just like in hierarchical clustering
  - Determined by the structure / content of the data
  - Usually euclidian distance
- K is determined in advance by the data scientist based on knowledge about the data or the goal of the analysis
  - Perhaps there are generally 2 types of geyser eruption because of physics
  - We may have resources to approach customers in at most 3 different ways

K-means clustering

- Because the initialization is random, the result is random
  - Label switching: cluster 1, 2, 3 may end up in each other's locations
  - Some examples at the boundary may end up in different clusters altogether
  - Use multiple starts to obtain the best solution

Evaluating cluster results

Evaluating cluster results

Many options (hyperparameters) in clustering methods

- They need to be tuned to obtain “good” clusters.
- “small decisions with big consequences” (ISLR2, section 12.4.3)
- In other words: clusters are sensitive to hyperparameter choice

Three methods to assess whether obtained clusters are “good”

- Stability
- External validity
- Internal validity

Cluster stability

How stable are the clusters?
Two different ways

Bootstrap stability
- Compute cluster solutions on M bootstrap samples (Hennig, 2007)
- Compare the similarity of the M cluster solutions to the original solution
- More similar is better!

Leave-one-feature-out
- Leave a feature out and perform clustering
- Compare the similarity of the cluster solutions to the original solution
- More similar is better!

External validity

- Are the clusters associated with external feature Y?
- Making unsupervised supervised (a little cheating)

- Examples:
  - Are my customer segments based on spending associated with the demographics of the customers?
  - Are the geyser eruption types strongly correlated with water pressure or temperature?
  - Can I recognize the person in the vector quantized picture?

Internal validity

- Use measure to calculate how close are documents in a cluster and how distant are documents in different clusters

- Coherence
  - Inter-cluster similarity v.s. intra-cluster similarity
  - Davies–Bouldin index

$$DB = \frac{1}{k}\sum_{i=1}^{k}\max_{j\neq i}\left(\frac{\sigma_i+\sigma_j}{d(c_i,c_j)}\right)$$ ← *Evaluate every pair of clusters*

where k is total number of clusters, σ_i is average distance of all elements in cluster i from the cluster center, $d(c_i,c_j)$ is the distance between cluster centroid c_i and c_j.

  - We prefer smaller DB-index!

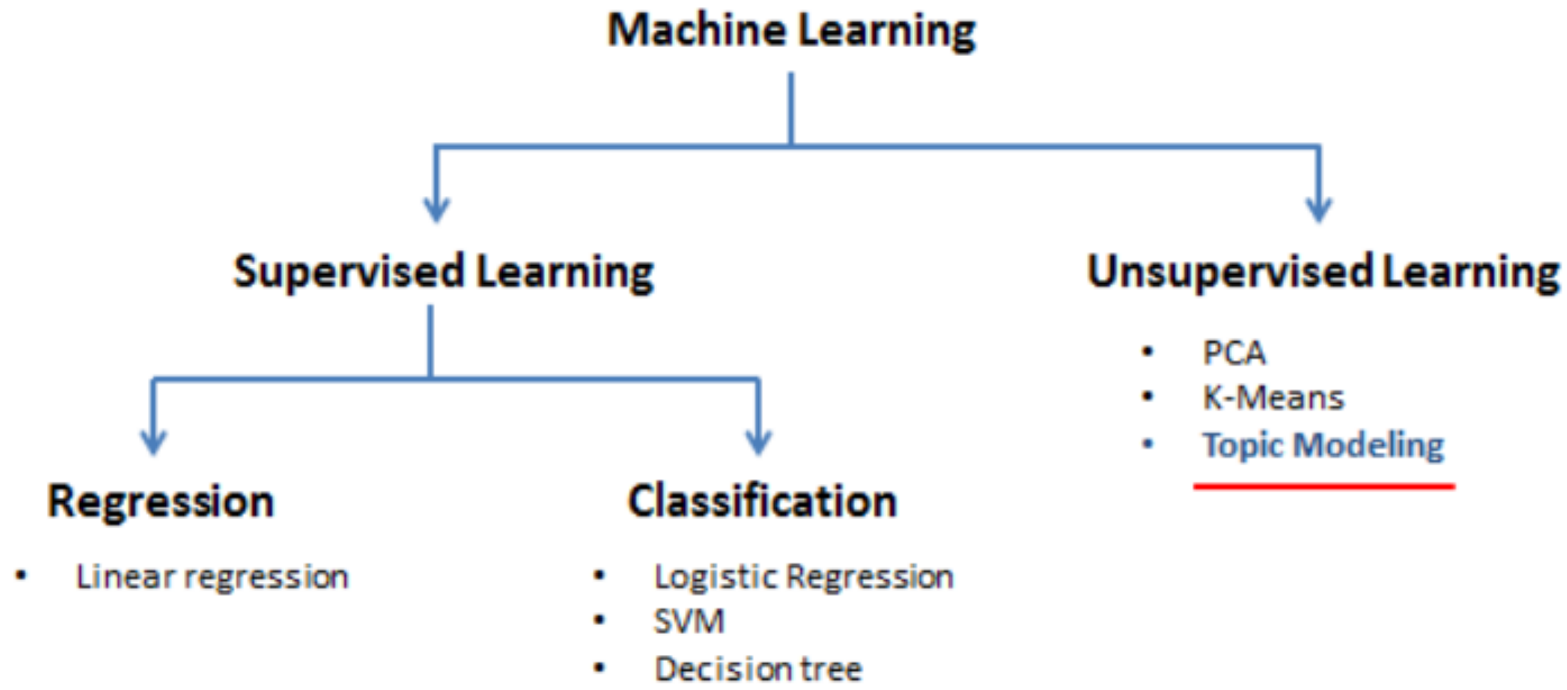
Validation in sklearn

- https://scikit-learn.org/stable/modules/clustering.html#clustering-performance-evaluation
- Rand index
- Mutual Information based scores
- Homogeneity, completeness and V-measure
- Fowlkes-Mallows scores
- Silhouette Coefficient
- Calinski-Harabasz Index
- Davies-Bouldin Index
- Contingency Matrix
- Pair Confusion Matrix

Break

Topic Modeling

Topic modeling

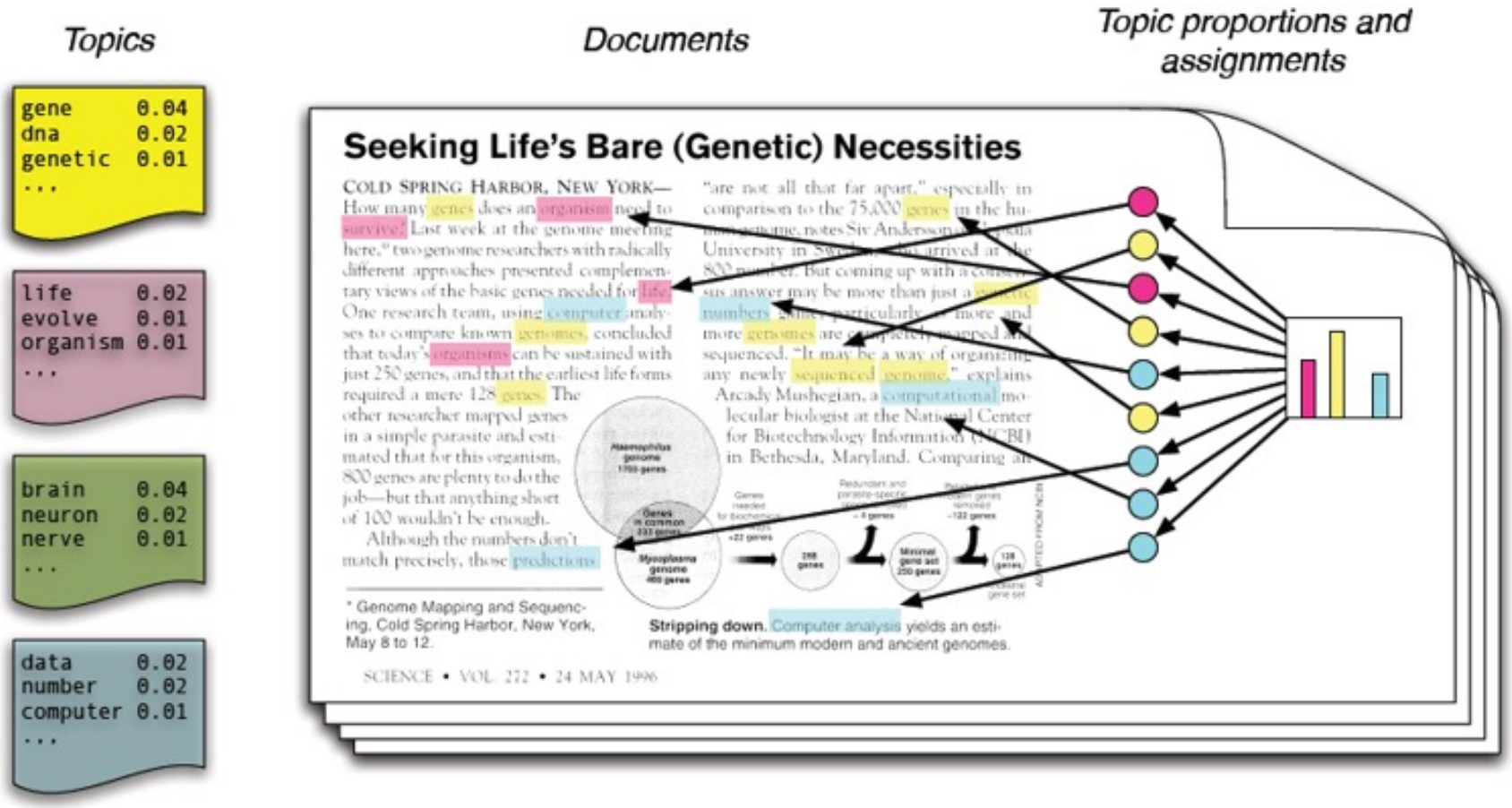


https://thinkinfi.com/

What is a Topic?

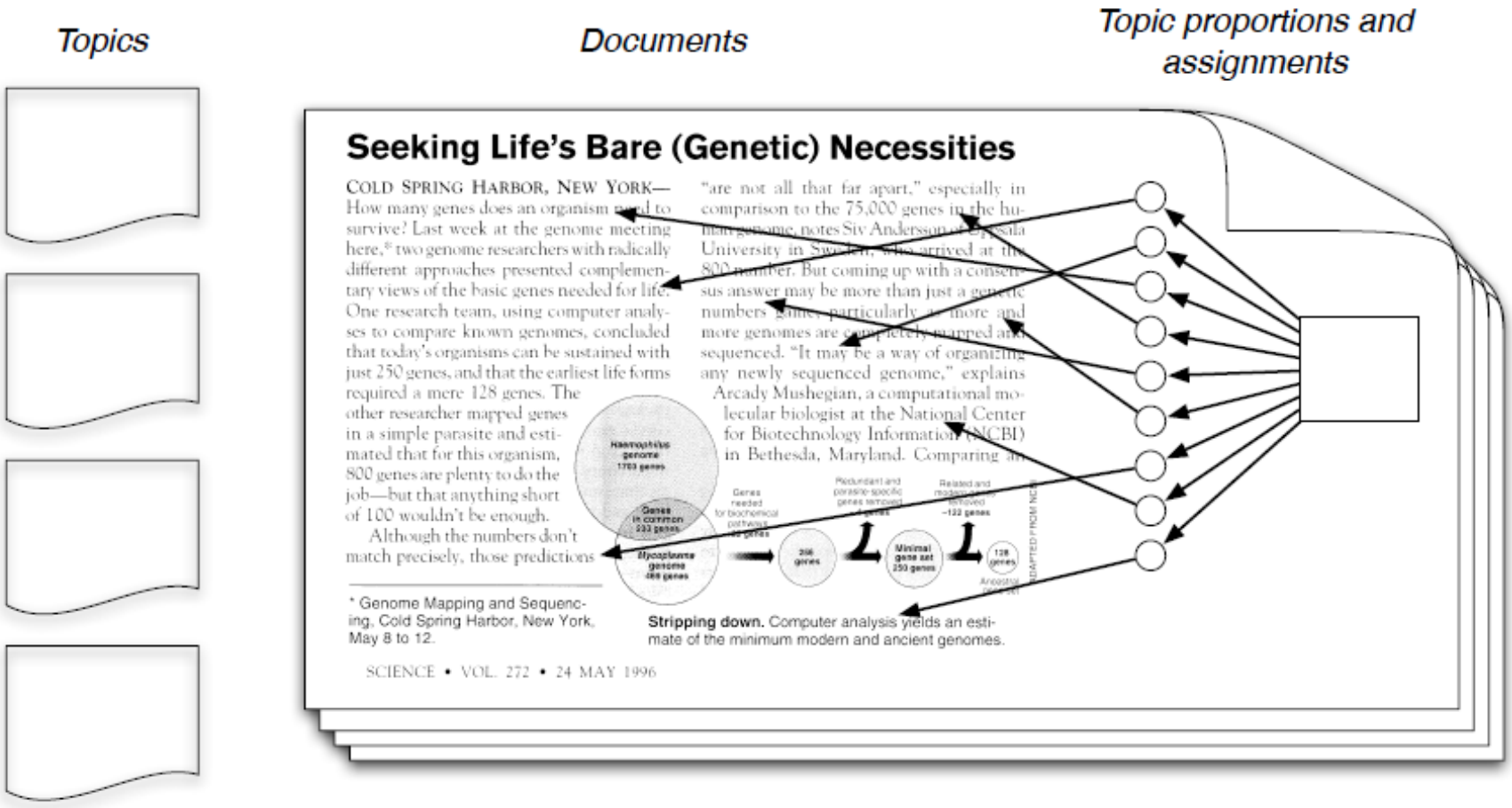
- A probabilistic distribution over words
- A broad concept/theme, semantically coherent, which is hidden in documents
  - e.g., politics; sports; technology; entertainment; education etc.

The goal



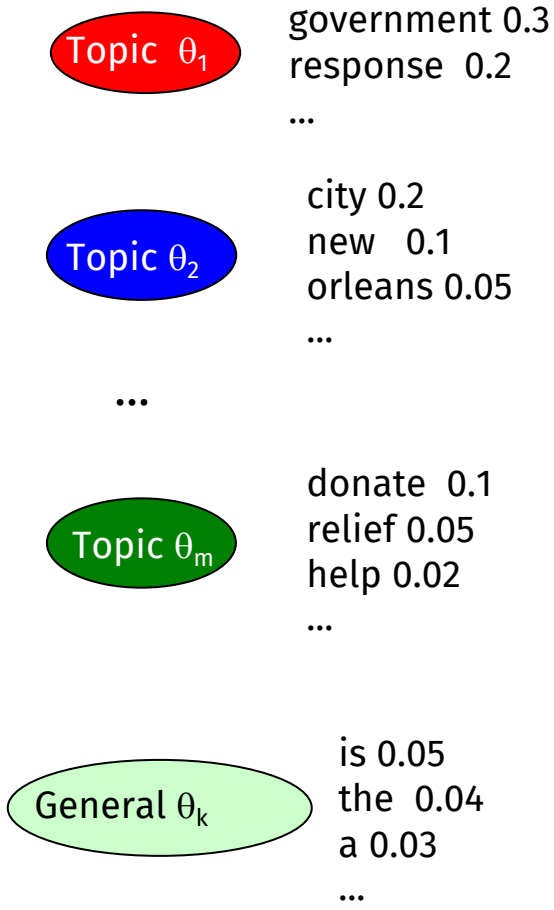
Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *the Journal of machine Learning research*, *3*, 993-1022. https://dl.acm.org/doi/pdf/10.5555/944919.944937

Reality


Topics
Documents
Topic proportions and assignments
Seeking Life's Bare (Genetic) Necessities
COLD SPRING HARBOR, NEW YORK—
How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.
Although the numbers don't match precisely, those predictions
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 689 genes
Genes needed for biochemical pathways 38 genes
256 genes
Redundant and parasite-specific genes removed 128 genes
Minimal gene set 250 genes
Related and nonessential genes removed ~122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

Document as a mixture of topics

[Criticism of government response to the hurricane primarily consisted of criticism of its response to the approach of the storm and its aftermath, specifically in the delayed response] to the [flooding of New Orleans. … 80% of the 1.3 million residents of the greater New Orleans metropolitan area evacuated] …[Over seventy countries pledged monetary donations or other assistance]. …



How can we discover these topic-word distributions?

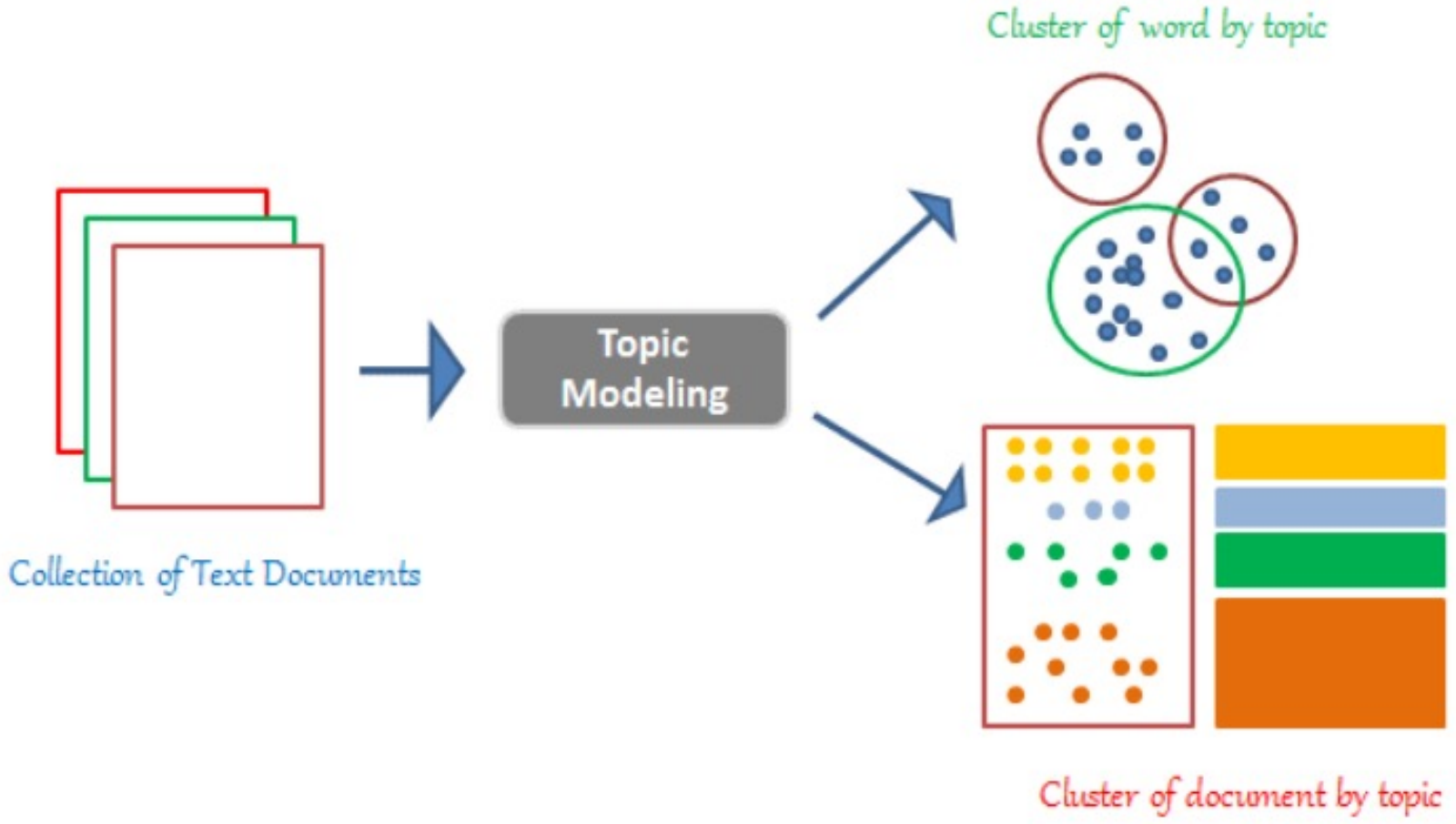
General idea of topic models

- **Topic**: a multinomial distribution over words
- **Document**: a mixture of topics
  - A document is “generated” by first sampling topics from some prior distribution
  - Each time, sample a word from a corresponding topic
  - Many variations of how these topics are mixed
- **Topic modeling**
  - Fitting the probabilistic model to text
  - Answer topic-related questions by computing various kinds of posterior distributions
    - e.g., p(topic|time), p(sentiment|topic)

Applications

- Dimensionality reduction / Summarizing topics
- Clustering / classification
- Predict topic coverage for documents / Model the topic correlations
- Many other text mining tasks

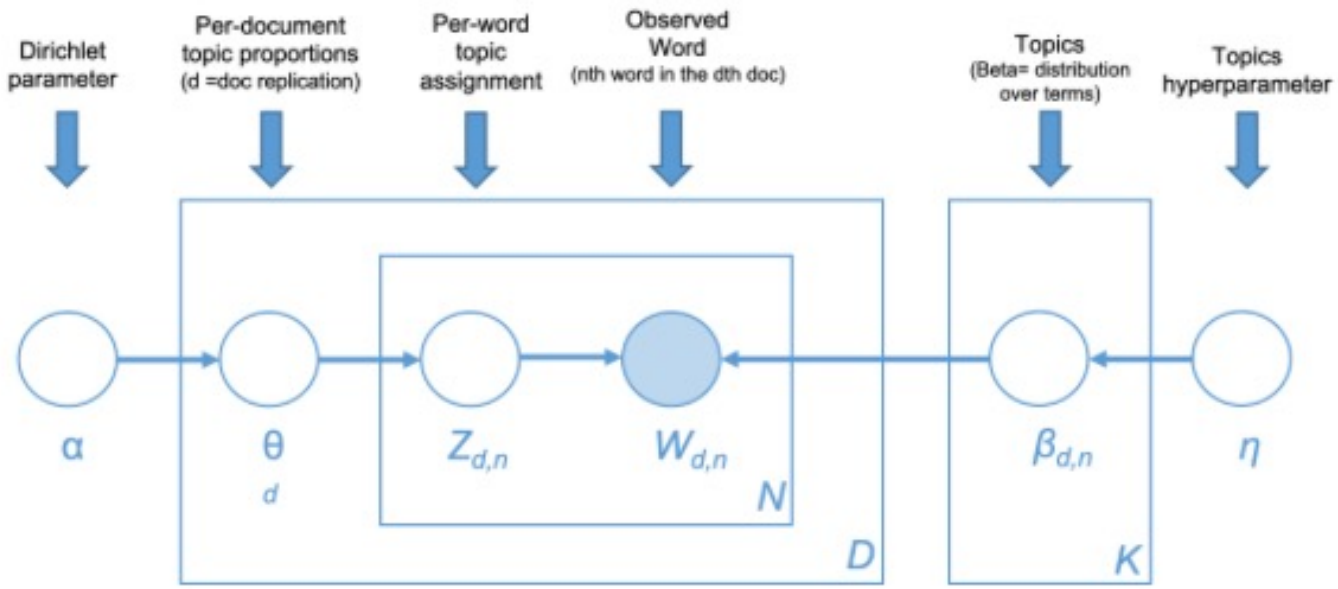
Topic modeling



Latent Dirichlet Allocation

- LDA is most used topic modeling method.
- LDA follows the basic principles of topic modeling.
- In LDA:
  - Documents are represented as a mixture of topics,
  - And a topic is a bunch of words.
  - Those topics reside within a hidden, also known as a latent layer.

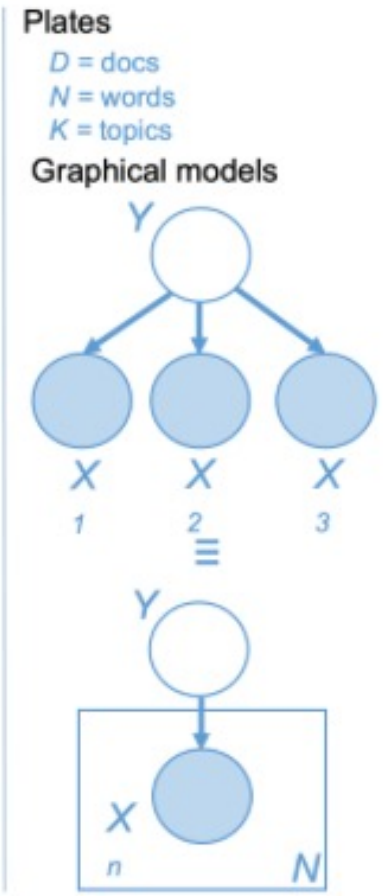
LDA graphical model



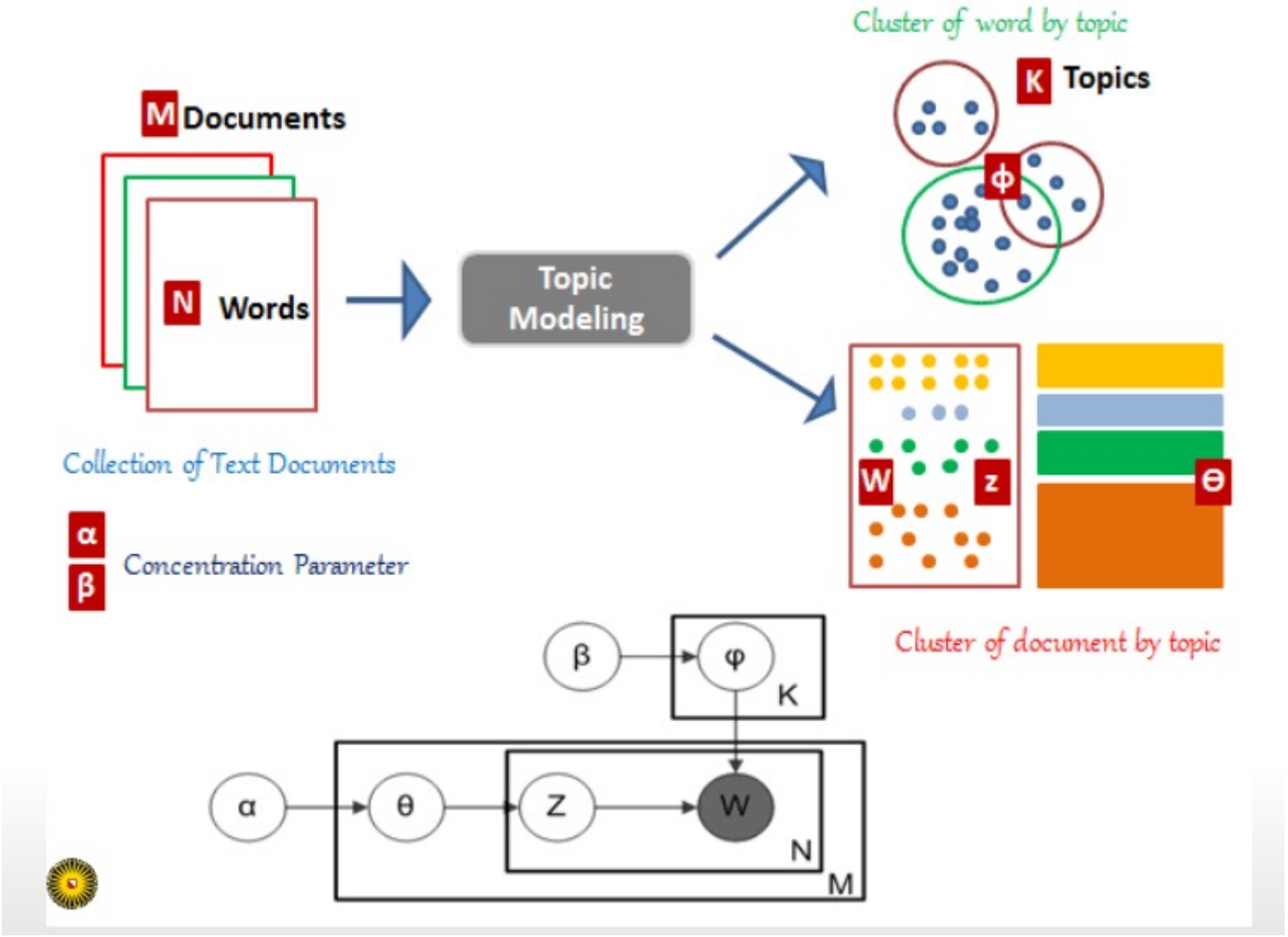
Graphical model representation of LDA. The boxes are "plates" representing replicates.
The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

- Nodes are random variables
- Edges denote possible dependence
- Observed variables are shaded
- Plates denote replicated structure

K	specified number of topics
k	auxiliary index over topics
V	number of words in vocabulary
v	auxiliary index over topics
d	auxiliary index over documents
N_d	document length (number of words)
i	auxiliary index over words in a document
α	positive K-vector
β	positive V-vector
$Dir(\alpha)$	a K-dimensional Dirichlet
$Dir(\beta)$	a V-dimensional Dirichlet
z	Topic indices: $z_{d,i} = k$ means that the i-th word in the d-th document is assigned to topic k



LDA graphical model Explained!



Probabilistic modeling

1. Treat data as observations that arise from a generative probabilistic process that includes hidden variables: For documents, the hidden variables reflect the thematic structure of the collection.
2. Infer the hidden structure using posterior inference: What are the topics that describe this collection?
3. Situate new data into the estimated model: How does this query or new document fit into the estimated topic structure?

LDA example

- What is latent Dirichlet allocation? It's a way of automatically discovering topics that these sentences contain.
- Suppose you have the following set of sentences:
  - I like to eat broccoli and bananas.
  - I ate a banana and spinach smoothie for breakfast.
  - Chinchillas and kittens are cute.
  - My sister adopted a kitten yesterday.
  - Look at this cute hamster munching on a piece of broccoli.

LDA example

- Given these sentences and asked for 2 topics, LDA might produce something like:
  - Sentences 1 and 2: 100% Topic A
  - Sentences 3 and 4: 100% Topic B
  - Sentence 5: 60% Topic A, 40% Topic B
  - Topic A: 30% broccoli, 15% bananas, 10% breakfast, 10% munching, … (at which point, you could interpret topic A to be about food)
  - Topic B: 20% chinchillas, 20% kittens, 20% cute, 15% hamster, … (at which point, you could interpret topic B to be about cute animals)
- How does LDA perform this discovery?

LDA training

- Go through each document, and randomly assign each word in the document to one of the K topics.
- Notice that this random assignment already gives you both topic representations of all the documents and word distributions of all the topics (albeit not very good ones).
- So to improve on them, for each document d…
- Go through each word w in d…

LDA training

- And for each topic t, compute two things:
  - p(topic t | document d) = the proportion of words in document d that are currently assigned to topic t, and
  - p(word w | topic t) = the proportion of assignments to topic t over all documents that come from this word w.
- Reassign w a new topic, where we choose topic t with probability p(topic t | document d) * p(word w | topic t)
- In other words, in this step, we're assuming that all topic assignments except for the current word in question are correct, and then updating the assignment of the current word using our model of how documents are generated.

LDA training

- After repeating the previous step a large number of times, you'll eventually reach a roughly steady state where your assignments are pretty good.
- Use these assignments to estimate the topic mixtures of each document (by counting the proportion of words assigned to each topic within that document) and the words associated to each topic (by counting the proportion of words assigned to each topic overall).

LDA in Python

sklearn.decomposition.LatentDirichletAllocation

class sklearn.decomposition.**LatentDirichletAllocation**(*n_components=10, *, doc_topic_prior=None, topic_word_prior=None, learning_method='batch', learning_decay=0.7, learning_offset=10.0, max_iter=10, batch_size=128, evaluate_every=-1, total_samples=1000000.0, perp_tol=0.1, mean_change_tol=0.001, max_doc_update_iter=100, n_jobs=None, verbose=0, random_state=None*) [source]

Examples

```
>>> from sklearn.decomposition import LatentDirichletAllocation
>>> from sklearn.datasets import make_multilabel_classification
>>> # This produces a feature matrix of token counts, similar to what
>>> # CountVectorizer would produce on text.
>>> X, _ = make_multilabel_classification(random_state=0)
>>> lda = LatentDirichletAllocation(n_components=5,
...     random_state=0)
>>> lda.fit(X)
LatentDirichletAllocation(...)
>>> # get topics for some given samples:
>>> lda.transform(X[-2:])
array([[0.00360392, 0.25499205, 0.0036211 , 0.64236448, 0.09541846],
       [0.15297572, 0.00362644, 0.44412786, 0.39568399, 0.003586  ]])
```

Topics learned by LDA

AP corpus

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

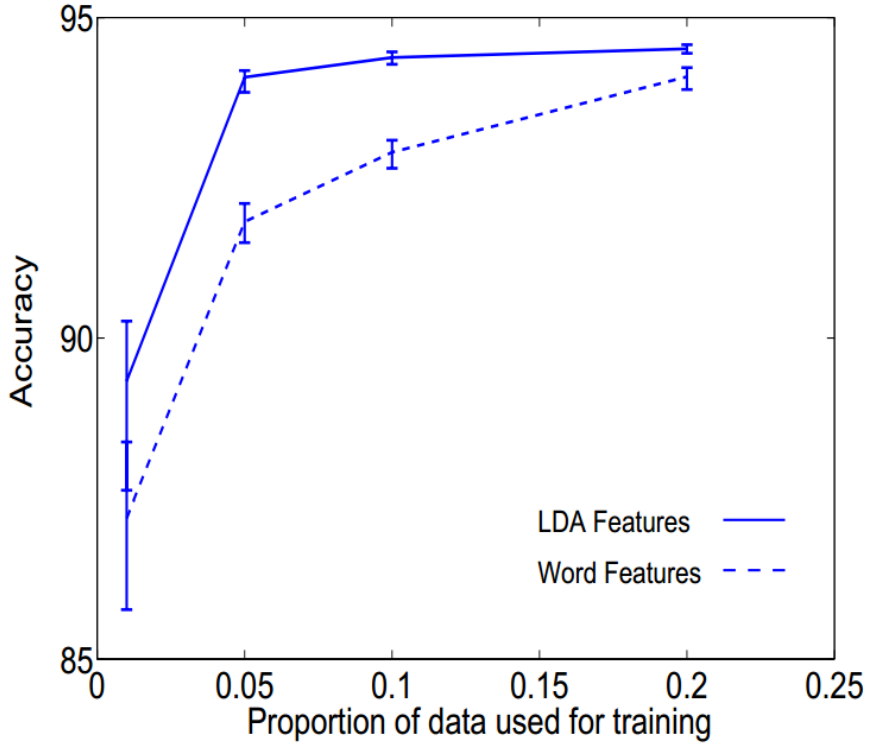
Topic assignments

AP corpus

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Application of learned topics

- Document classification
  - A new type of feature representation



Variants of topic models

- Smoothed LDA
- Correlated Topic Models
- Hierarchical Topic Models
- Dynamic Topic Models
- Contextual Topic Models
- BERTtopic: https://github.com/MaartenGr/BERTopic
- And many more!

Conclusion

Text clustering & topic modelling

In clustering, clusters are inferred from the data without human input (unsupervised learning)

Many ways of influencing the outcome of clustering: number of clusters, similarity measure, representation of documents

Evaluation is important!

Practical

Create document-term matrices on BBC news dataset and apply LDA topic modeling.

Questions?