UNIVERSITY OF BIRMINGHAM

Machine Learning: Unsupervised Learning

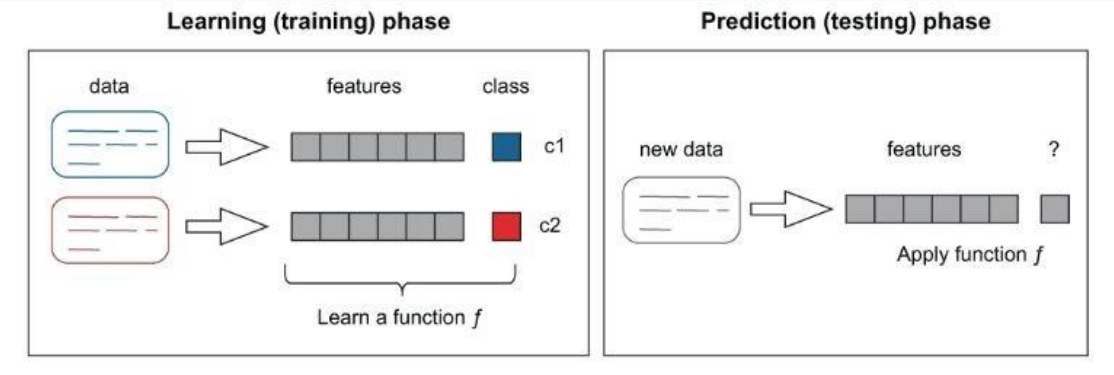
Mubashir Ali
m.ali.16@bham.ac.uk

Learning outcomes

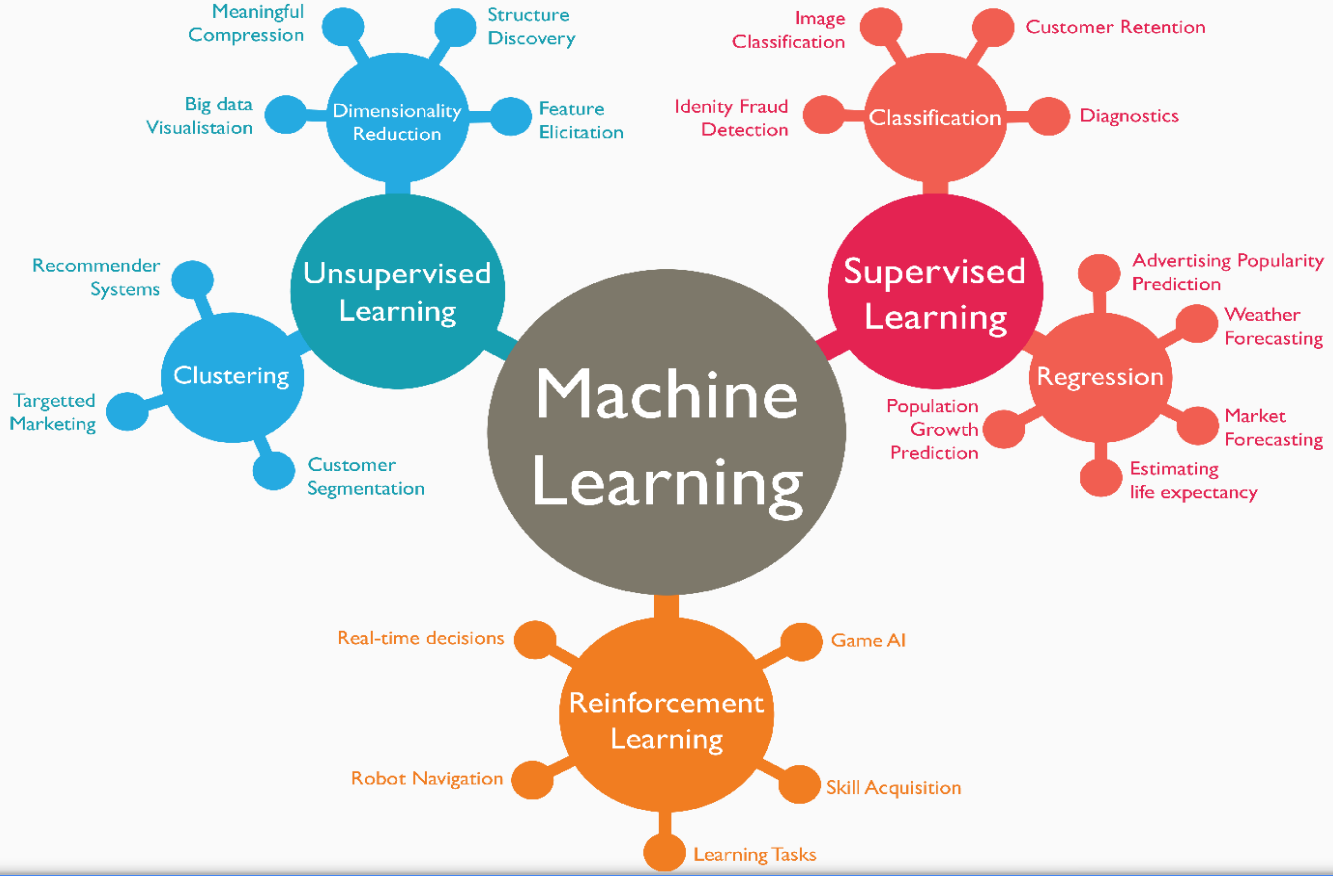
- **Applications:** Topic analysis using unsupervised approaches
- **ML concepts and techniques:**
  - k-means clustering
  - topic modelling with Latent Dirichlet Allocation (LDA)
- **Practical skills:** you will learn how to structure an unsupervised ML application and how to implement it in practice and evaluate the results

Recap: Classification Problem

- Last week we discussed **supervised machine learning approaches** and **classification** problems.
- Applicable where you have access to **representative data** labelled with **classes** relative to the given application.

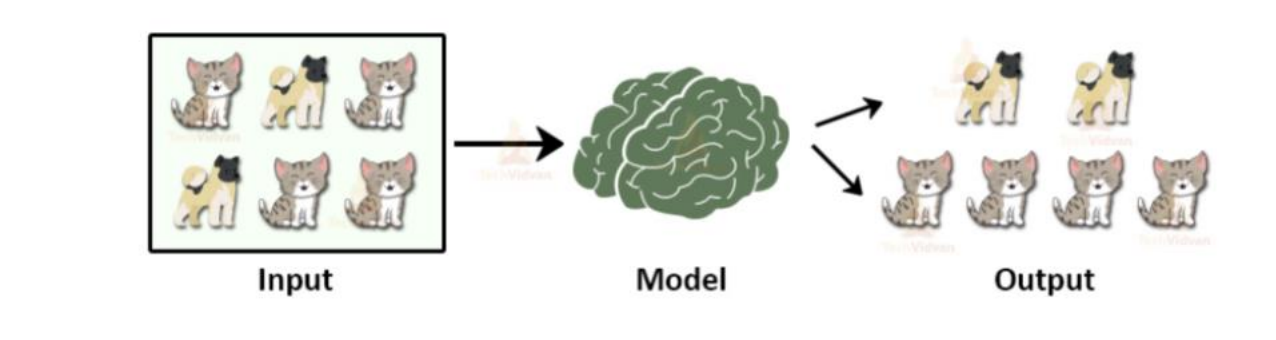


Types of Machine Learning



Unsupervised Learning

- Learning interesting patterns from dataset with no labels $x^{(1)}$,....., $x^{(n)}$



- **Clustering** algorithm tries to detect similar groups.
- **Dimensionality reduction** tries to simplify the data without losing too much information.

USL Applications in Intelligent Interactive Systems

- Natural Language Processing (NLP)
- Game AI
- Pattern recognition in images or signals
- Human activity recognition
- Smart home automation
- Customer segmentation
- Anomaly detection
- Financial market analysis
- Many more

Topic Classification (Supervised)

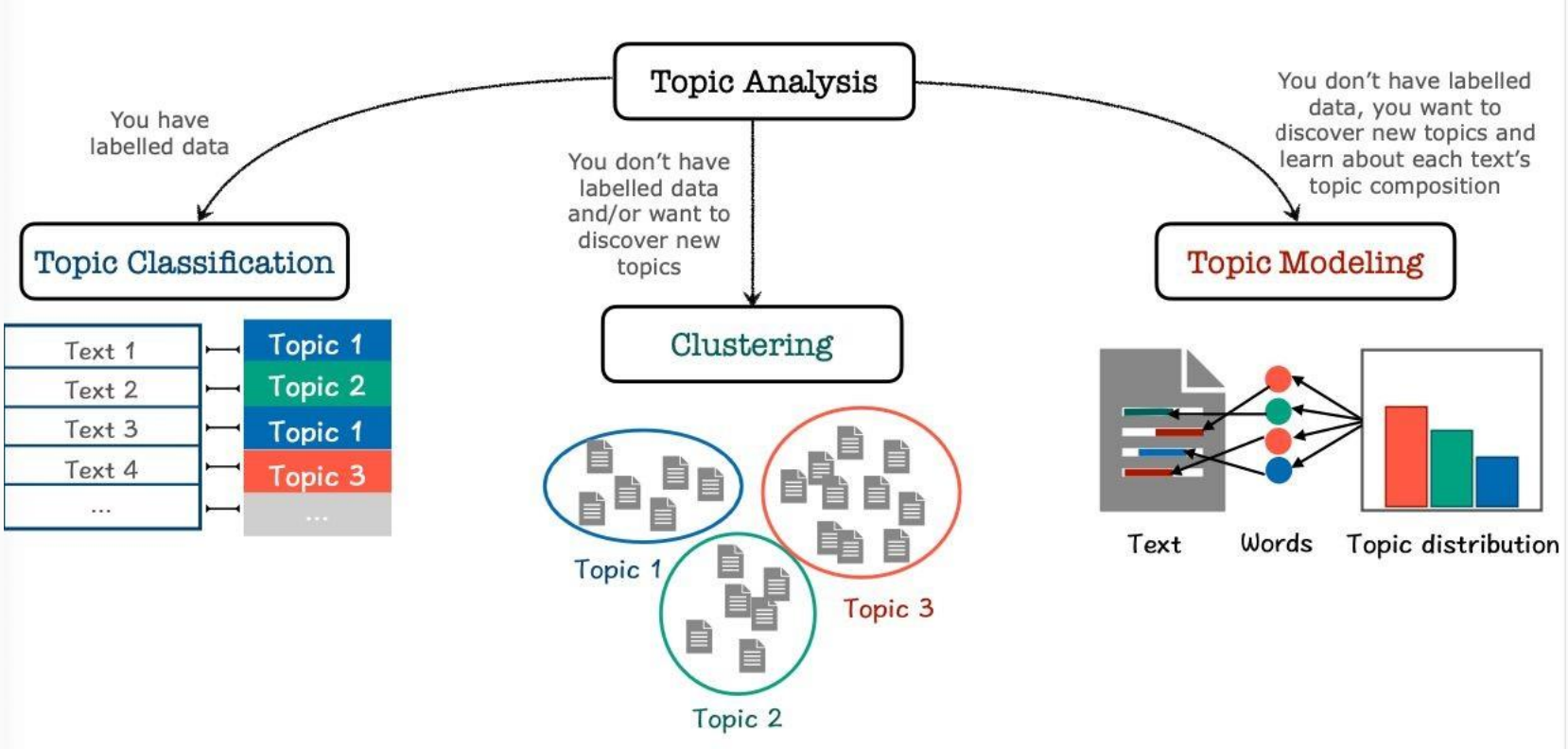
- Multi-class classification problem
- Example: what topic does a news story belong to?
- **20 Newsgroups**
- ~20,000 newsgroup documents
- Partitioned almost evenly across …. 20 different newsgroups.
- Computers, sport, space, religion etc.
- Supervised ML can be useful given ambiguous nature of words:
    - political **system,** operating **system, system** of beliefs

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

General Comparison

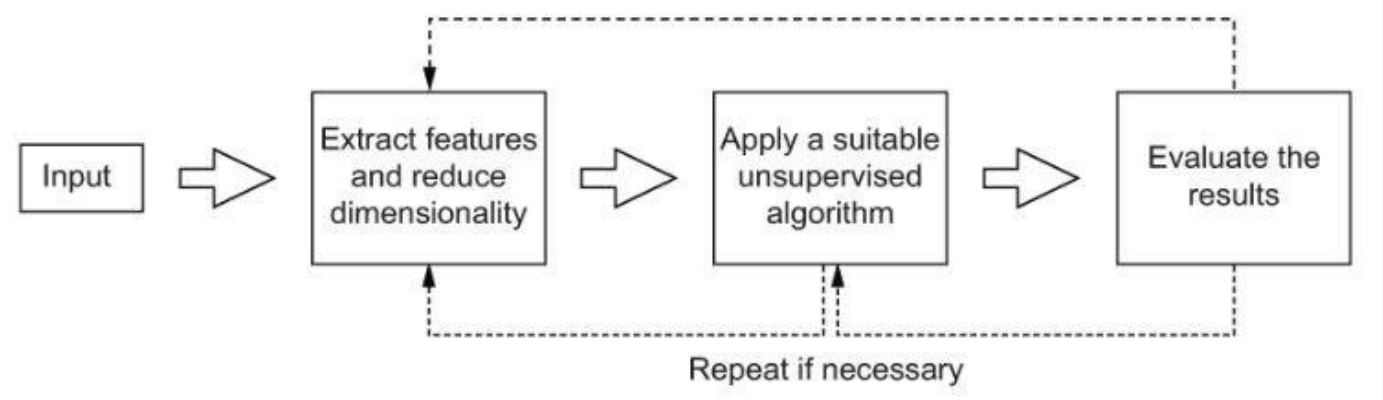
- Supervised approach requires reliable labels, but…
  - Cost associated
  - Annotators may disagree
  - Different world views
  - Unconscious biases
- Unsupervised approaches let the composition of the data come through in classification
  - No pre-assigned labels
  - Discover clases based on features

Topic analysis



Topic Discovery

Mental model for application of unsupervised approaches:



Similar to supervised approach

Key difference: algorithm identifies groups of similar texts without linking them to classes

Dimensionality reduction is very important, why?

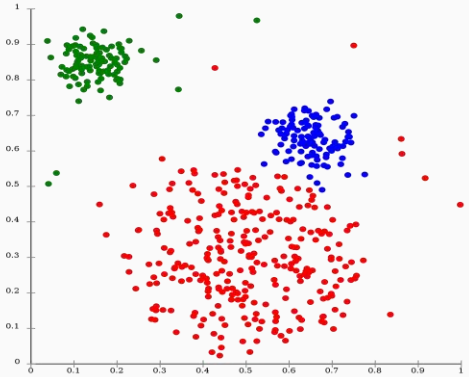
Feature representation

Feature vector, but no notion of what the class is.

	f_0="car"	f_1="engine"	f_2="tree"	f_3="moon"	f_0="rocket"	…	Cluster
Text 1	0.55	0.45	0.08	0.01	0.05	…	
Text 2	0.65	0.72	0.01	0.01	0.02	…	
…	…	…	…	…	…	…	…
Text n-1	0.02	0.07	0.09	0.66	0.81	…	
Text n	0.01	0.01	0.05	0.67	0.99	…	

Clustering

- Aims to identify groups of instances that are similar to each other
- Similarity with respect to a given application
  - Topic analysis
  - Customer segmentation
  - Human activity recognition



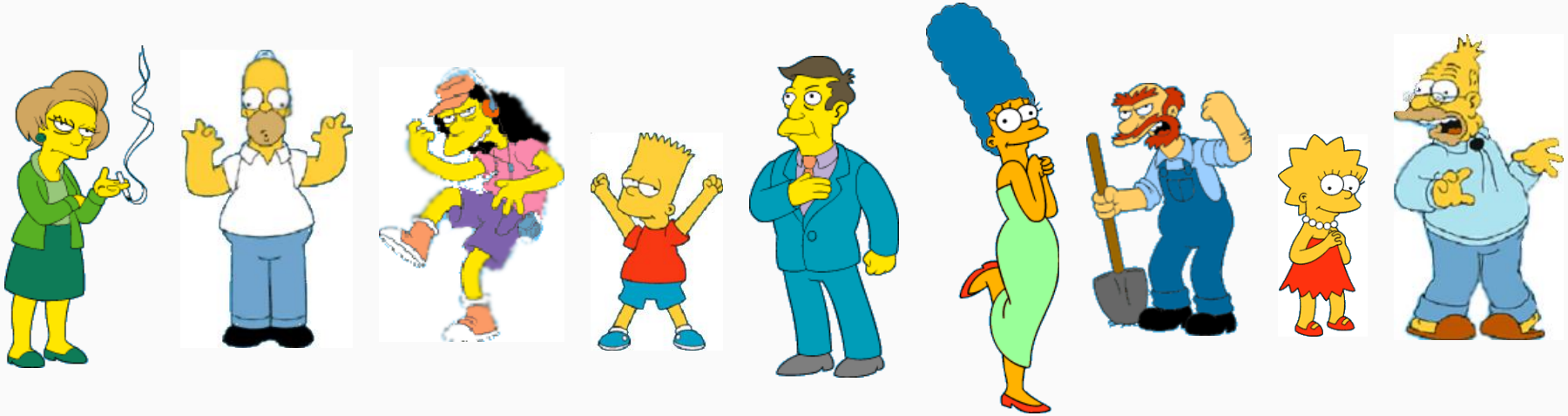
- Ensure dissimilar instances given the context end up in different groups
- These groups are the clusters

Clustering

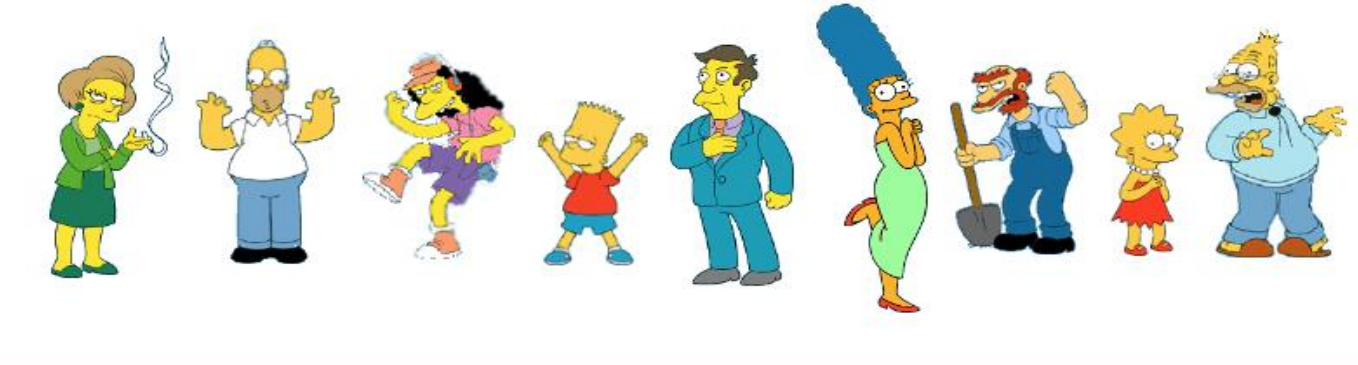
- Organizing data into classes such that there is
  - high intra-class similarity
  - Low inter-class similarity

- Finding the class labels and the number of classes directly from the data (in contrast to classification).
- More informally, finding natural grouping among objects.

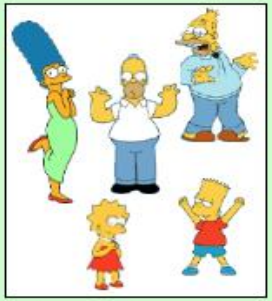
What is natural grouping among these objects?



What is natural grouping among these objects?

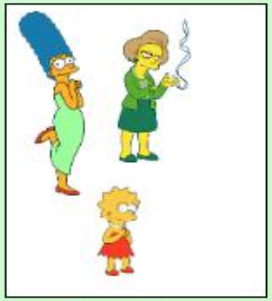


Clustering is subjective



Simpson's Family

School Employees



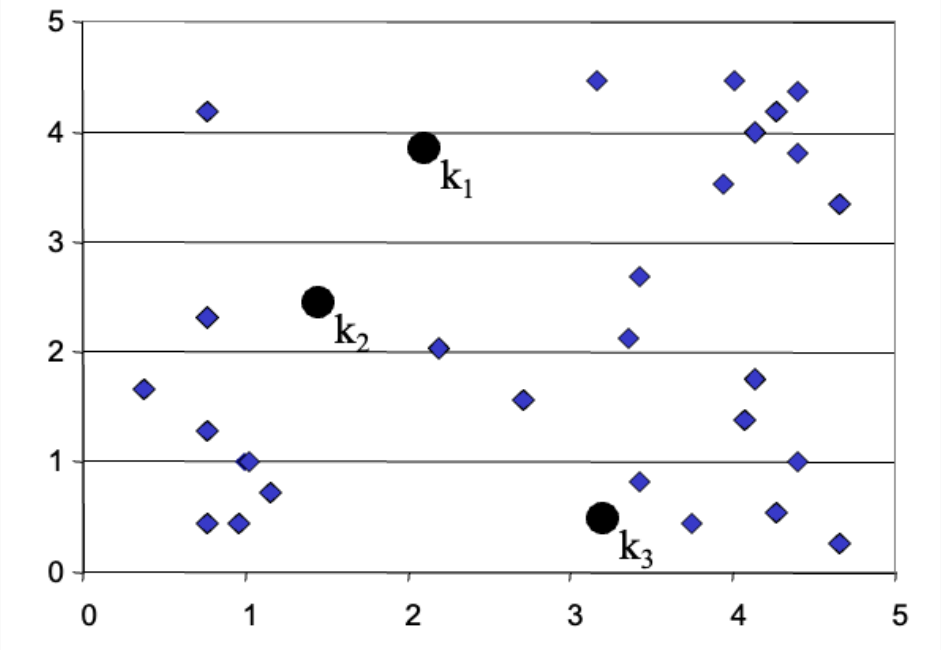
Females

Males

K-means clustering

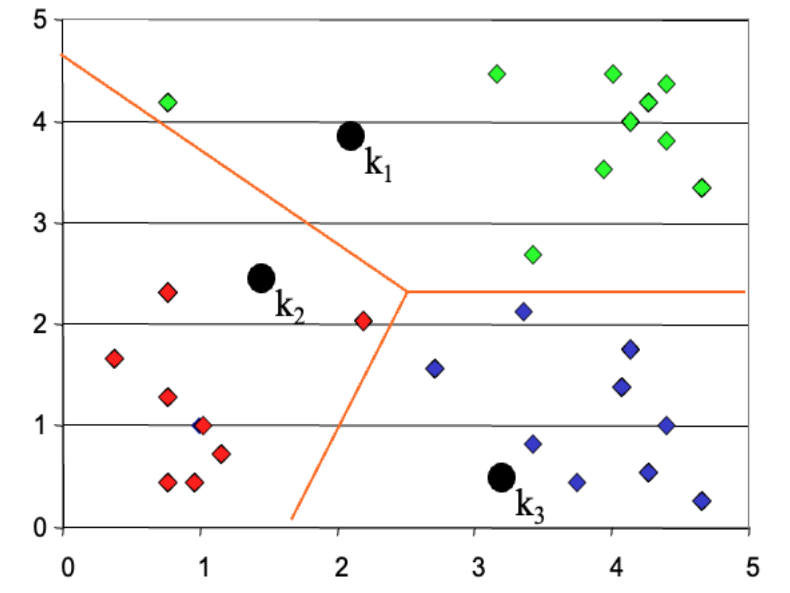
Steps in k-means clustering

- **1. Choose a centroid for each cluster randomly.**
  - e.g randomly select $centroid_1$, $centroid_2$ and $centroid_3$



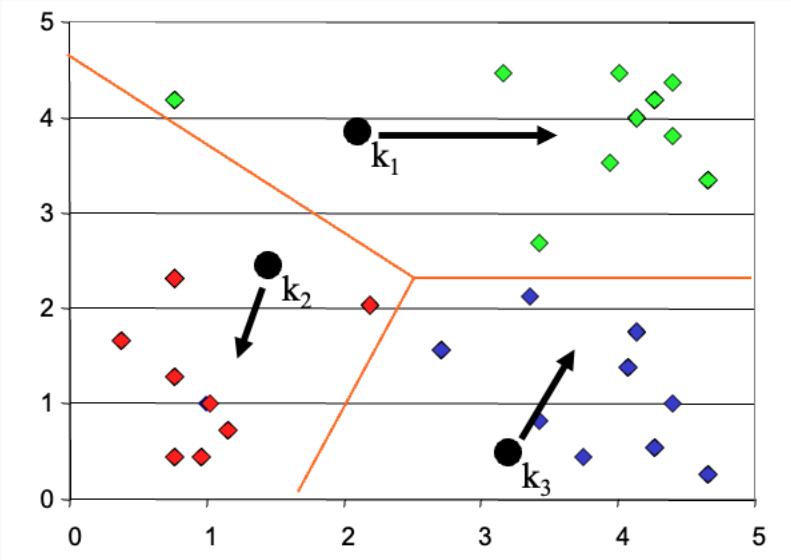
Steps in k-means clustering

- **2. For each object in the collection, estimate it's distance from each centroid**
- Estimate distance for each object using the coordinates of the object and calculating the Euclidean distance
- Do for each object and assign it to $cluster_1$, $cluster_2$ or $cluster_3$ based on whichever is closer to $centroid_1$ $centroid_2$ or $centroid_3$



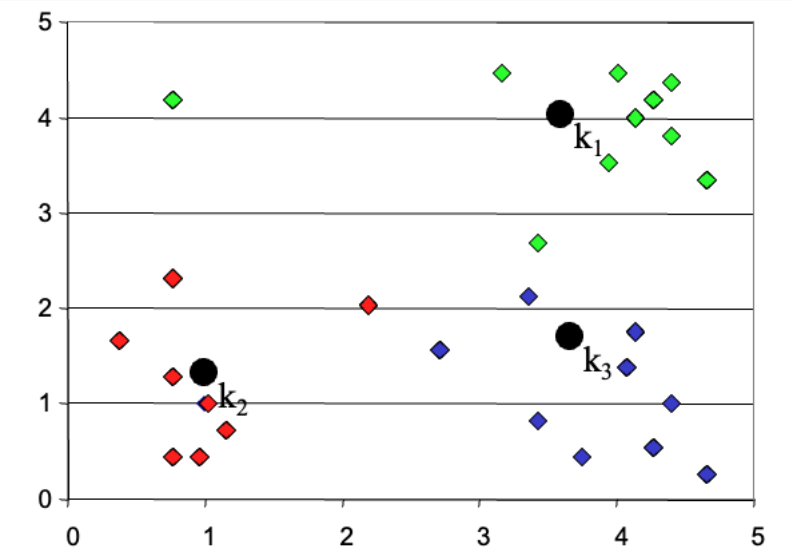
Steps in k-means clustering

- **3. Now you have cluster allocation, re-estimate where centroids for each cluster lie**
- This is based on the average values from the object assigned to the relevant cluster
- This step may **change** the centroids



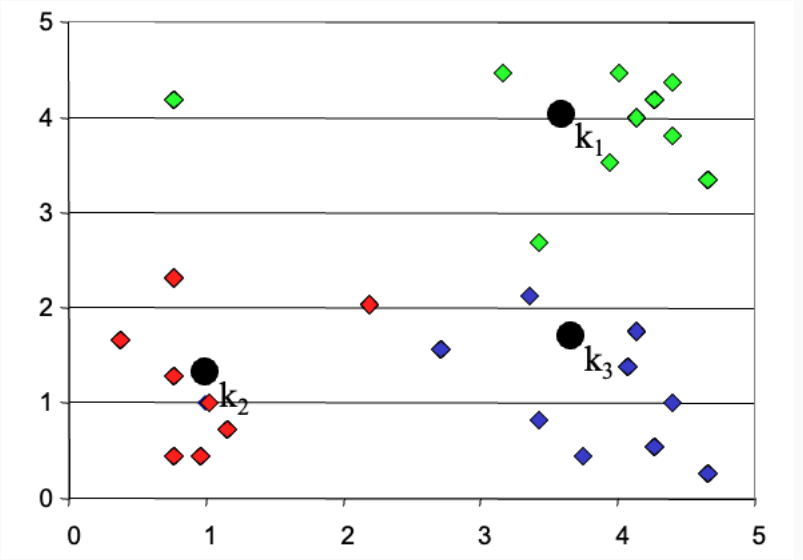
Steps in k-means clustering

- **4. Reallocate the points to clusters measuring distances to new centroids**
- Essentially, repeat step 2.
- Clusters may change too.
- If cluster allocation doesn't change, stop algorithm

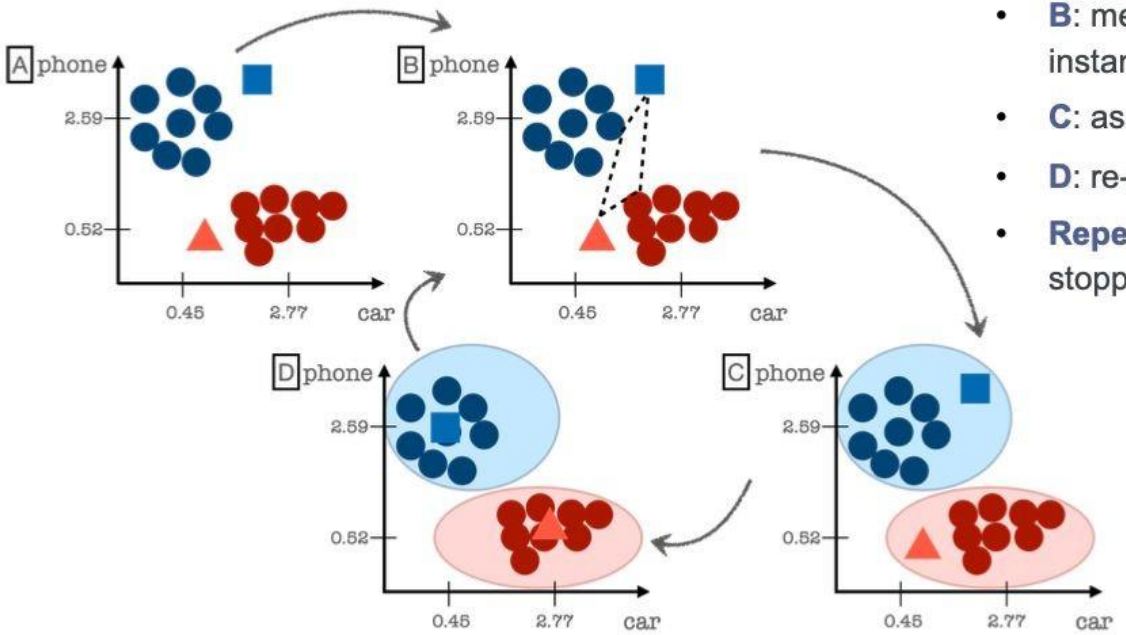


Steps in k-means clustering

- **5 to n. Repeat steps 3 and 4**
- Reestimate centroids based on new cluster allocations
- Cluster based on new centroids
- Do so for a specific number of steps *n*
- Or, until allocation of points to clusters no longer changes



1 page summary



- **A**: select K centroids randomly
- **B**: measure distances from each instance to each centroid
- **C**: assign instances to clusters
- **D**: re-estimate centroids
- **Repeat** B-D until one of the stopping criteria are satisfied

More formally…

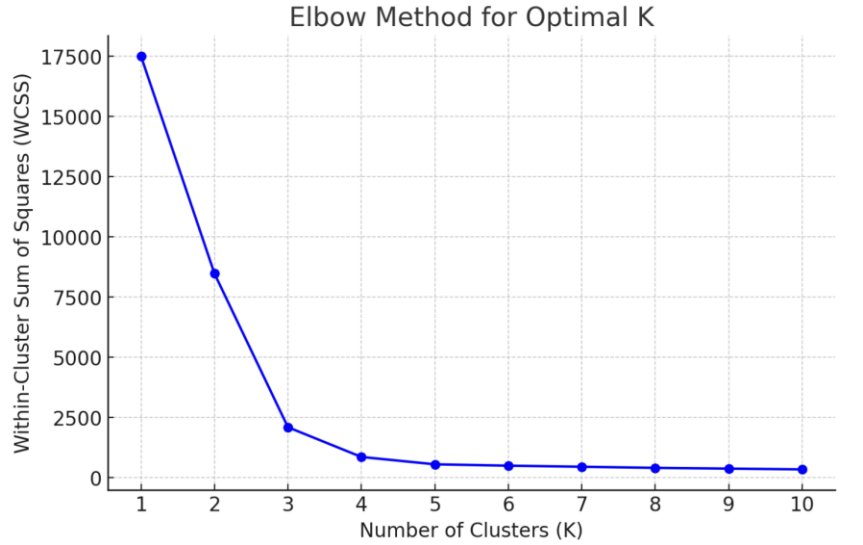
1. Randomly select centroids μ_1 and μ_2 for K=2
2. For each object in the dataset, estimate its distance from each centroid (μ_1 and μ_2) and link it to the cluster with nearest centroid (ω_1 or ω_2)

$$\begin{cases} x \in \omega_1 & \text{if } |\vec{x} - \vec{\mu}(\omega_1)|^2 \leq |\vec{x} - \vec{\mu}(\omega_2)|^2 \\ x \in \omega_2 & \text{otherwise} \end{cases}$$

3. Based on cluster allocation, re-estimate centroids
4. Re-allocated object wrt new centroids; re-estimate centroids
5. Continue till no further movement or max iteration reached (stopping) criteria

Choosing the best K

- Elbow Method



- Silhouette Score
  - How well points are clustered by computing **cohesion vs. separation.**
  - Higher the Silhouette Score, the better the clustering.
  - Typically between 0.5 and 1 is ideal

Dimensionality Reduction for Clustering

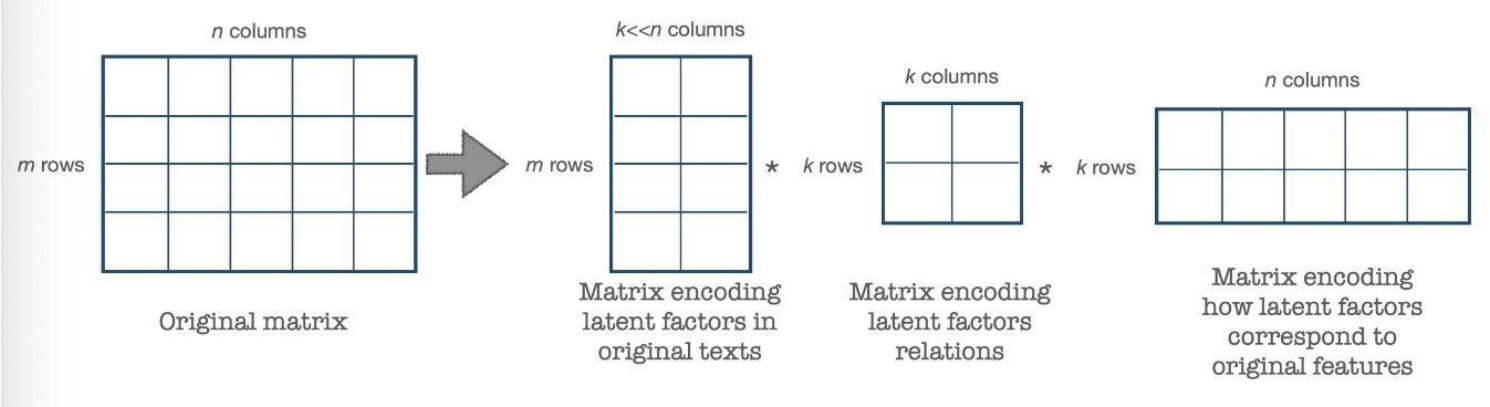
Dimensionality Reduction for Clustering

- Feature selection for unsupervised ML similar to supervised ML
  - Words
  - Filter stopwords
  - Apply weighting
- Issues with unsupervised ML
  - Calculating distance from each doc to each centroid for x iterations
  - Documents are short but feature space can be large, even after stopword removal. Vectors sparse
- Remove very frequent words due to lack of informativeness
- Apply Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD)

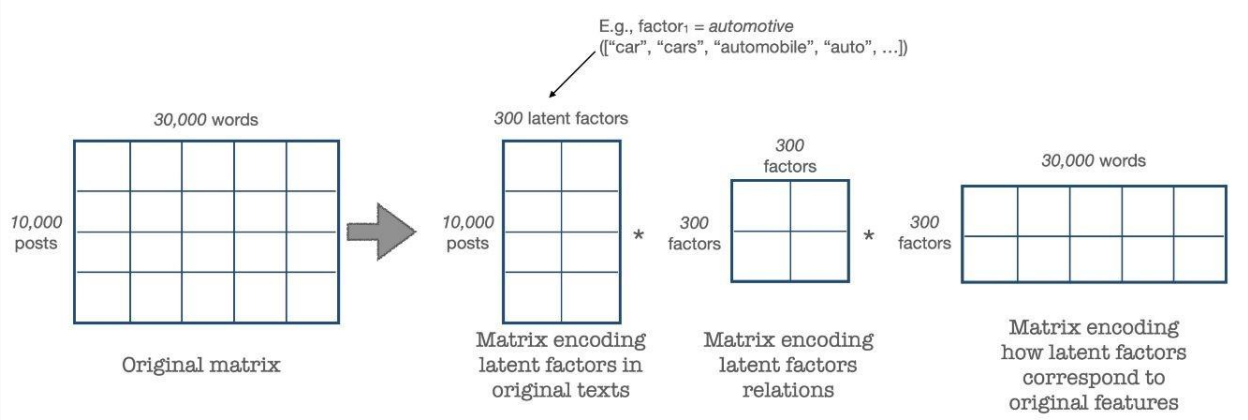
- Goal: reduce dimensionality to make clustering more efficient while maintaining useful information.
- SVD tries to distil and summarize the information contained in the original huge matrix down to a much smaller dimensions. e.g 30,000 to 300 dimensions
- Simplification via decomposition from one matrix to 3 smaller ones
    - When multiplying the 3, you get back the original

Singular Value Decomposition



- First encodes *m* texts in terms of smaller *k* features (concepts, or latent factors)
- Second describes relations between the *k* concepts
- Third encodes the relation between the *k* concepts and the original *n* word features

Interpretation of SVD

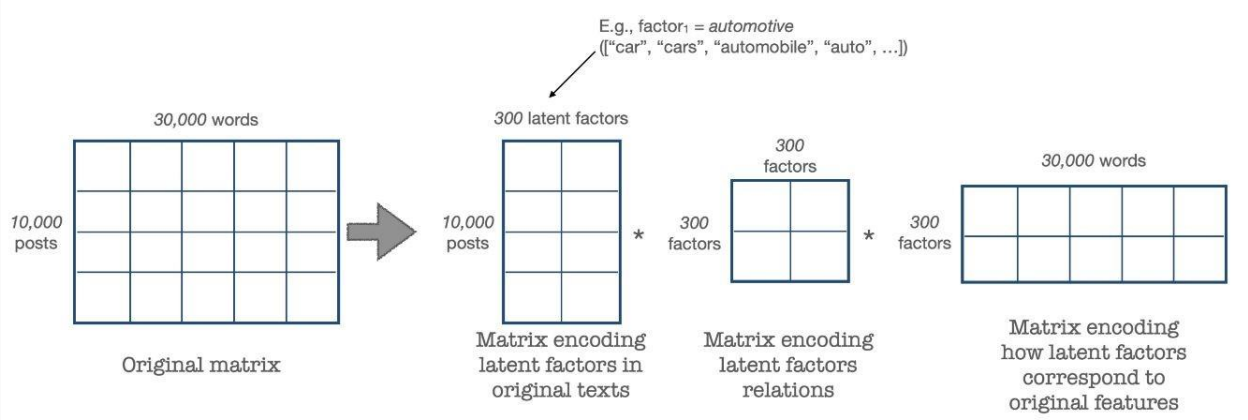


Each original word represents its own dimension

Separate column in original *n* dimensional matrix

There might represent the same 'concept' so if merged into a single concept this would help reduce the word space

Interpretation of SVD



m x k matrix is reduced document-by-latent factors matrix

k most salient factors

k x k matrix encodes how latent factors correspond to each other

k x n matrix tells you how to interpret the relations between factors and original words

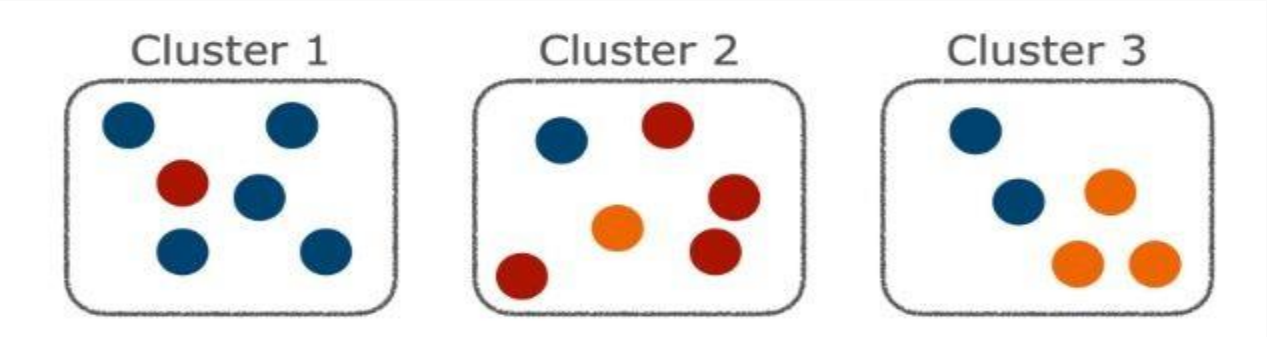
Evaluating Unsupervised ML algorithms

Evaluation of clustering algorithms

- Goal of clustering: attain high intra-cluster similarity
  - (all documents in a cluster are similar)
  - *Internal criterion* -relies on definition of space and distance metrics
- Might be useful to evaluate usefulness of identified clusters w.r.t **downstream applications**
  - Information retrieval -does the clustering of search results help a user?
- External criterion evaluation: compare results to labelled benchmark
  - Compare clusters identified by algorithms to topics (classes) originally assigned to data

Purity

- Assign class to each cluster based on majority of instances in cluster
- Estimate accuracy by counting number of correctly assigned documents then dividing by N
    - N is total number of docs in collection
- Purity is similar to accuracy, range from 0 to 1



Homogeneity and Completeness

- **Homogeneity** measures extent to which each clusters contains only members of a single class.
- Similar to precision estimating proportion of correct class predictions among all instances assigned to a class by a classifier.
- **Completeness** estimates extent to which all members of a class are assigned to same cluster.
- Similar to recall

V-measure

- Equivalent to *F-measure* in the unsupervised context.
- Harmonic mean of homogeneity and completeness

Topic Modelling

Scenario

- Content manager for a large news platform
- Wide variety of authors with a *well-established* set of topics:
  - Politics, sport, art, science
- Given new articles your system should determine which topic it belongs to and post to relevant part of the platform

Questions given the scenario

1. Can you use your knowledge of NLP and ML algorithms to help you automate this process?
    a. Text classification based approaches that we've learned so far are applicable
    b. But, reliant on high quality labelled data.
2. What if you suspect that a new set of uncovered topics emerges? How can you discover these?
    a. Unsupervised ML seems to be suitable approach
3. What if articles lend themselves to multiple topics?
    a. e.g football & rugby posts might have originally been clustered as "sport"
    b. or e.g forsale and autos

Latent Dirichlet Allocation can be used to detect multiple topics

Topic models

- Reminder: key differences between supervised and unsupervised ML is whether the classes are known in advance
  - So supervised algorithm learns a function mapping features to class labels
  - Unsupervised ML algorithms do not come with such an assumption
- Topic modelling is a text-mining technique aiming to discover abstract “topics” from document sets
- Relies on documents on a particular topic using a particular vocabulary
- Unlike hard clustering, topic modelling assumes the presence of multiple topics per document, *in different proportions*

Latent Dirchlet Allocation

- David M. Blei, Andrew Y. Ng, Michael I. Jordan (2003) “Latent Dirchlet Allocation” in Journal of Machine Learning Research, 3, pp.993-1022.
- Tackle the problem of modelling text corpora and other collections of discrete data
- Goal is to find short *descriptions* of members of a collection to enable efficient processing of large collections
- While preserving the statistical relationships that are useful for:
    - Classification, novelty detection, summarization and similarity and relevance judgements

Topics

gene	0.04
dna	0.02
genetic	0.01
...	

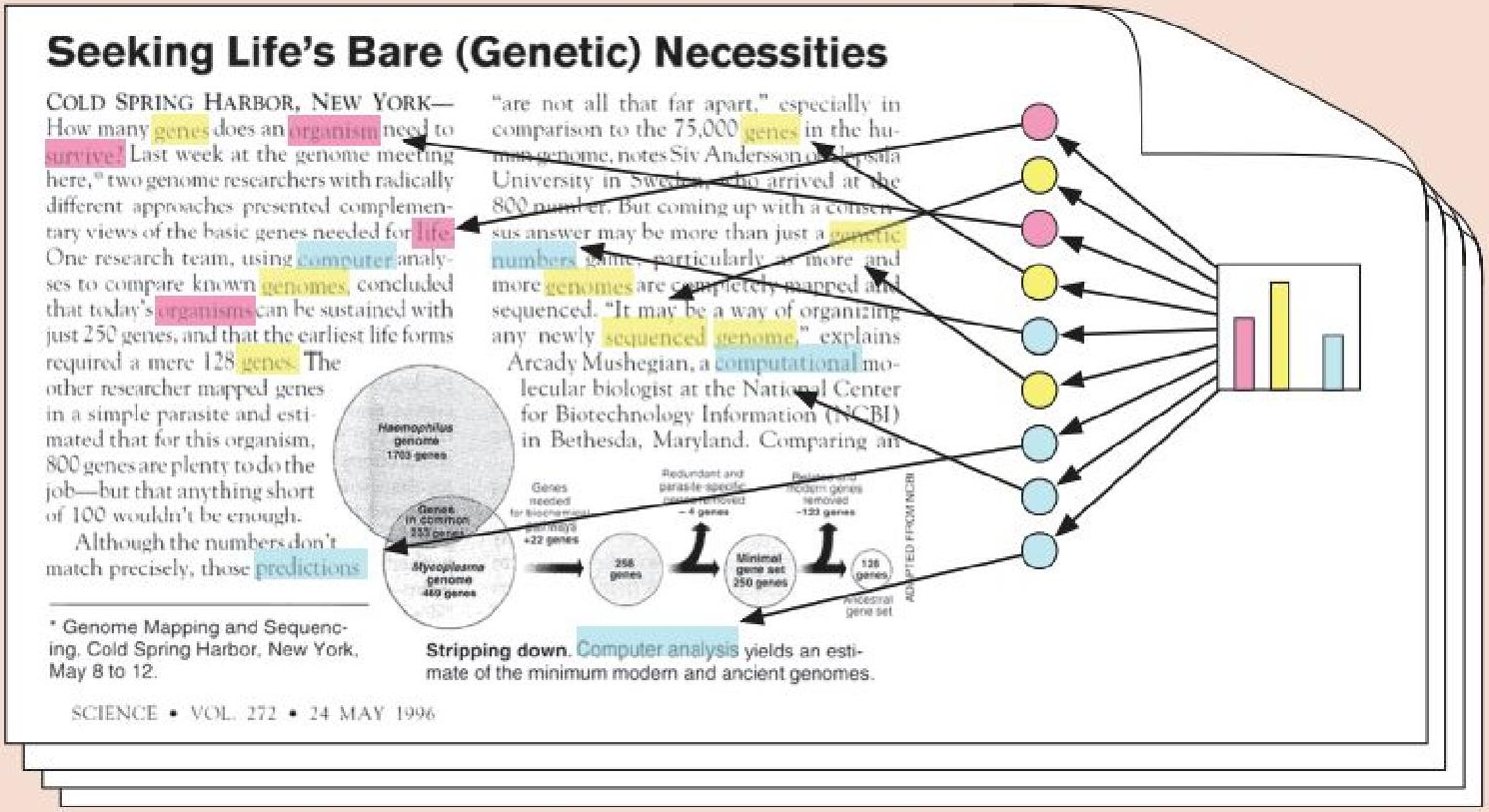
life	0.02
evolve	0.01
organism	0.01
...	

brain	0.04
neuron	0.02
nerve	0.01
...	

data	0.02
number	0.02
computer	0.01
...	

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

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

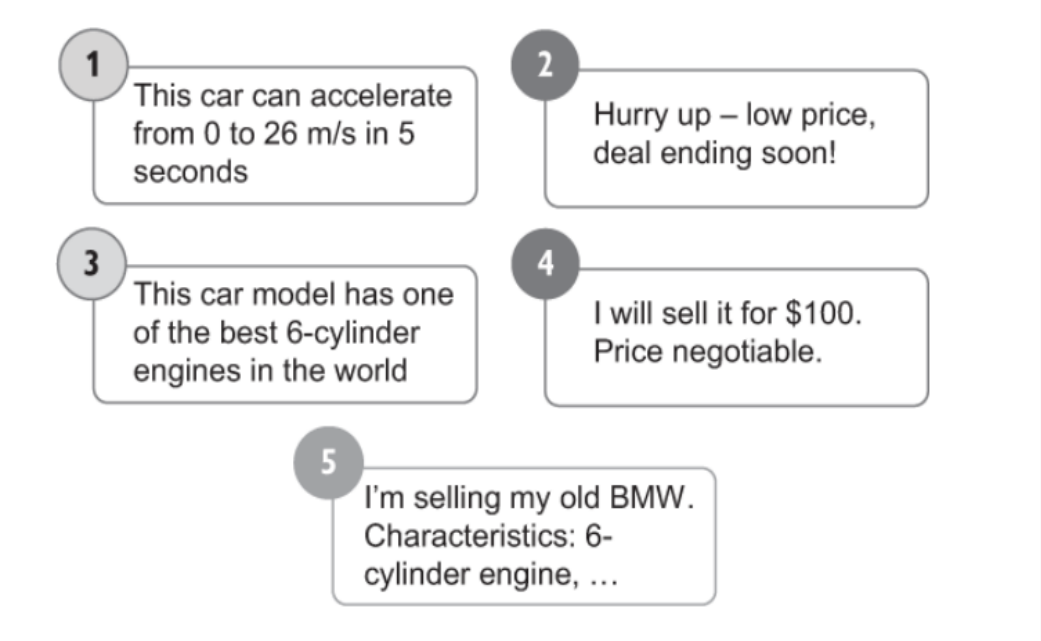
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996

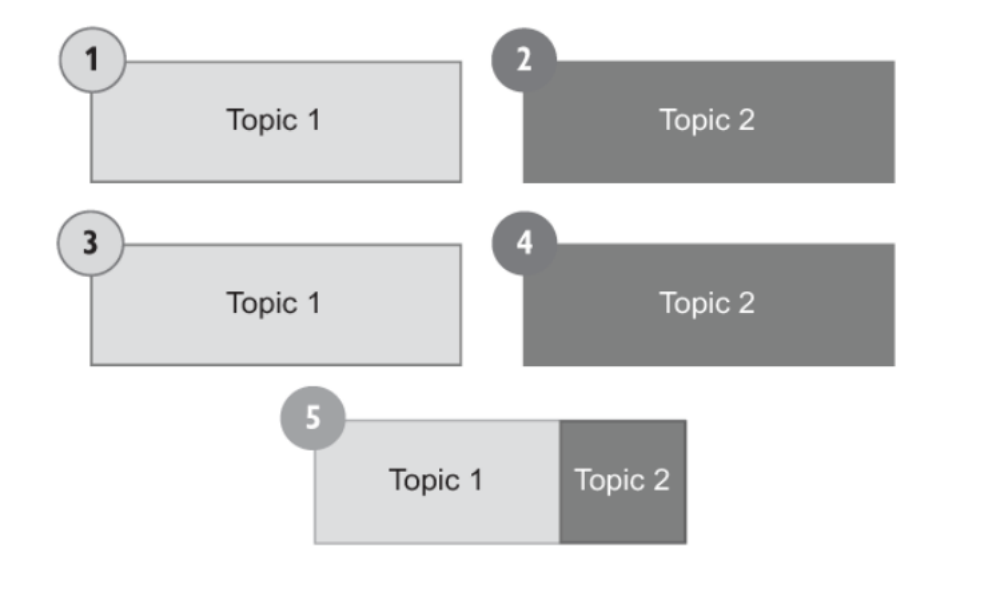
LDA as a generative process

- Imaginary random process by which the model assumes the documents arose or created
- Topic = distribution over fixed vocabulary
- Model assumes that the topics are generated first, before the documents.
- For each document in collection:
  - Randomly choose a distribution over topic
  - For each word in the document
    - Randomly choose a topic from the distribution over topics in step #1
    - Randomly choose a word from the corresponding distribution over the vocabulary

Example


1
This car can accelerate from 0 to 26 m/s in 5 seconds
2
Hurry up – low price, deal ending soon!
3
This car model has one of the best 6-cylinder engines in the world
4
I will sell it for $100. Price negotiable.
5
I'm selling my old BMW. Characteristics: 6-cylinder engine, …

Example

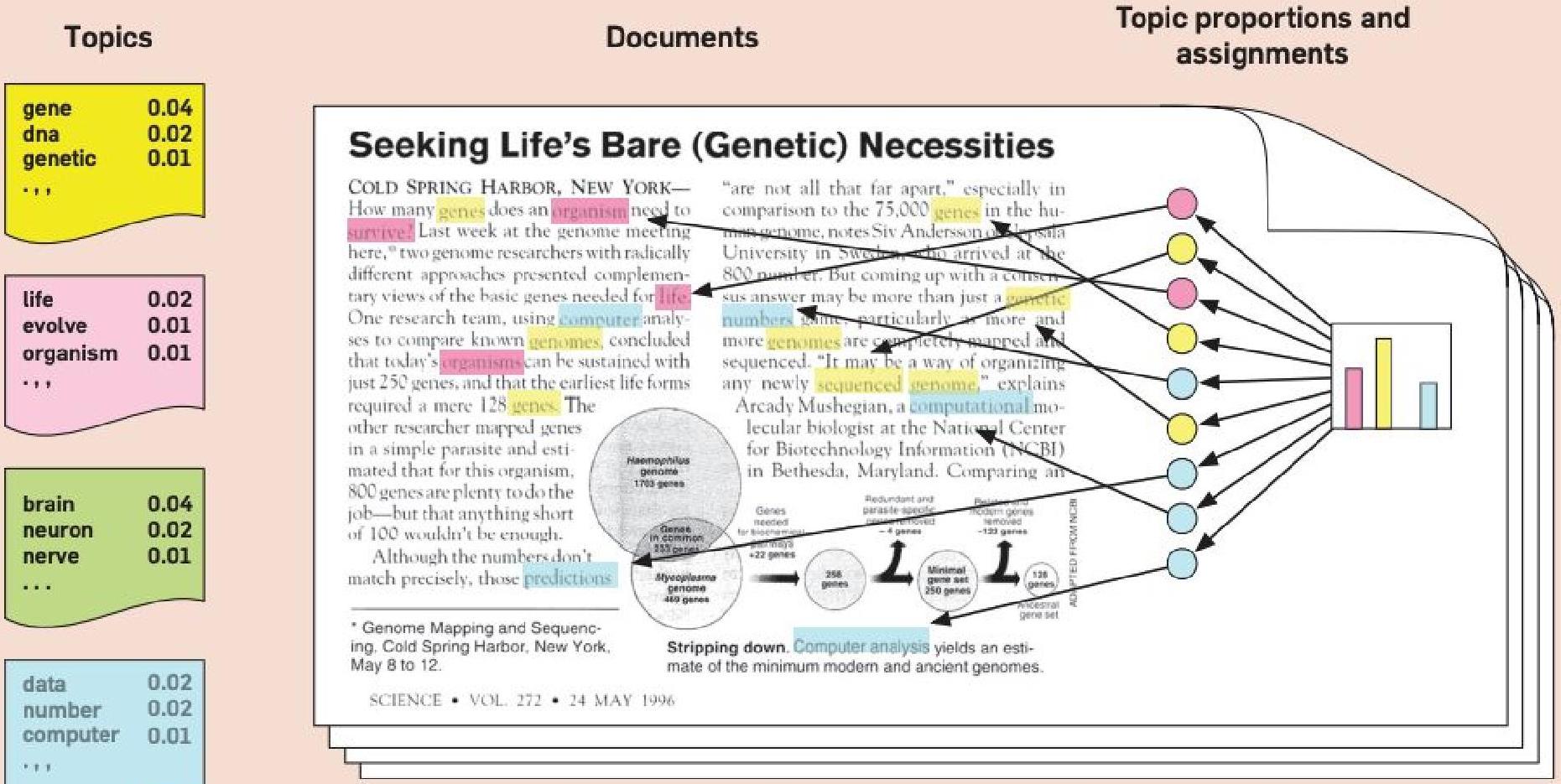


LDA as generative process

- This model reflects the intuition that documents exhibit multiple topics
- Each document exhibits the topics in different proportions
- Each word in each document is drawn from one of the topics
- Selected topic is chosen from the per-document distribution over topics
- LDA gets it's name from the per-document topic distribution that is called the *Dirichlet distribution*
- In the generative process for LDA, the Dirichlet is used to *allocate* the words of the document. So where does the *latent* come from?

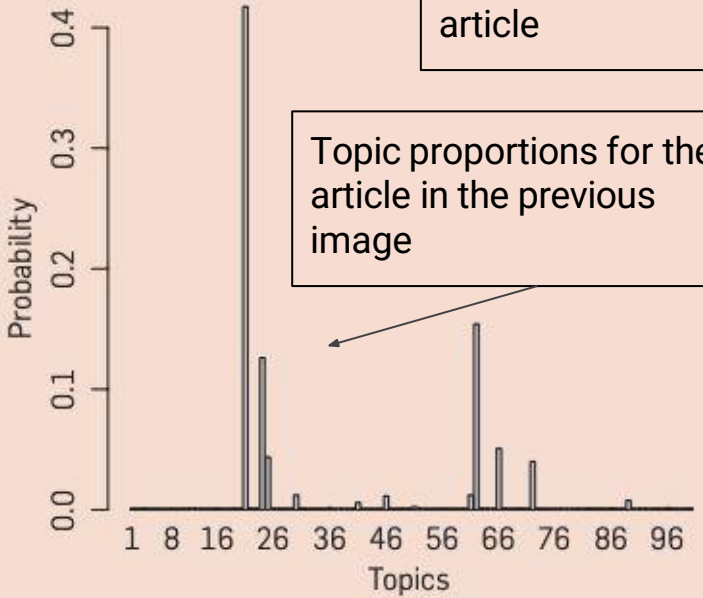
Hidden Structures

- Reminder of goal: discover new topics from document set
- Document are observed
- Topics, per-document topic distributions and the per-document per-word topic assignments are *hidden structure*
- The central computational problem for topic modeling is to use the observed documents to infer the hidden structure
- What is the hidden structure that likely generated the observed collection?


Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
...
life 0.02
evolve 0.01
organism 0.01
...
brain 0.04
neuron 0.02
nerve 0.01
...
data 0.02
number 0.02
computer 0.01
...
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 333 genes
Mycoplasma genome 469 genes
Genes needed for biochemistry 256 + 0(0) +22 genes
256 genes
Redundant and parasite-specific genes removed – 4 genes
Minimal gene set 250 genes
Related modern 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

Most probable words from most probable topics in article

Topic proportions for the article in the previous image

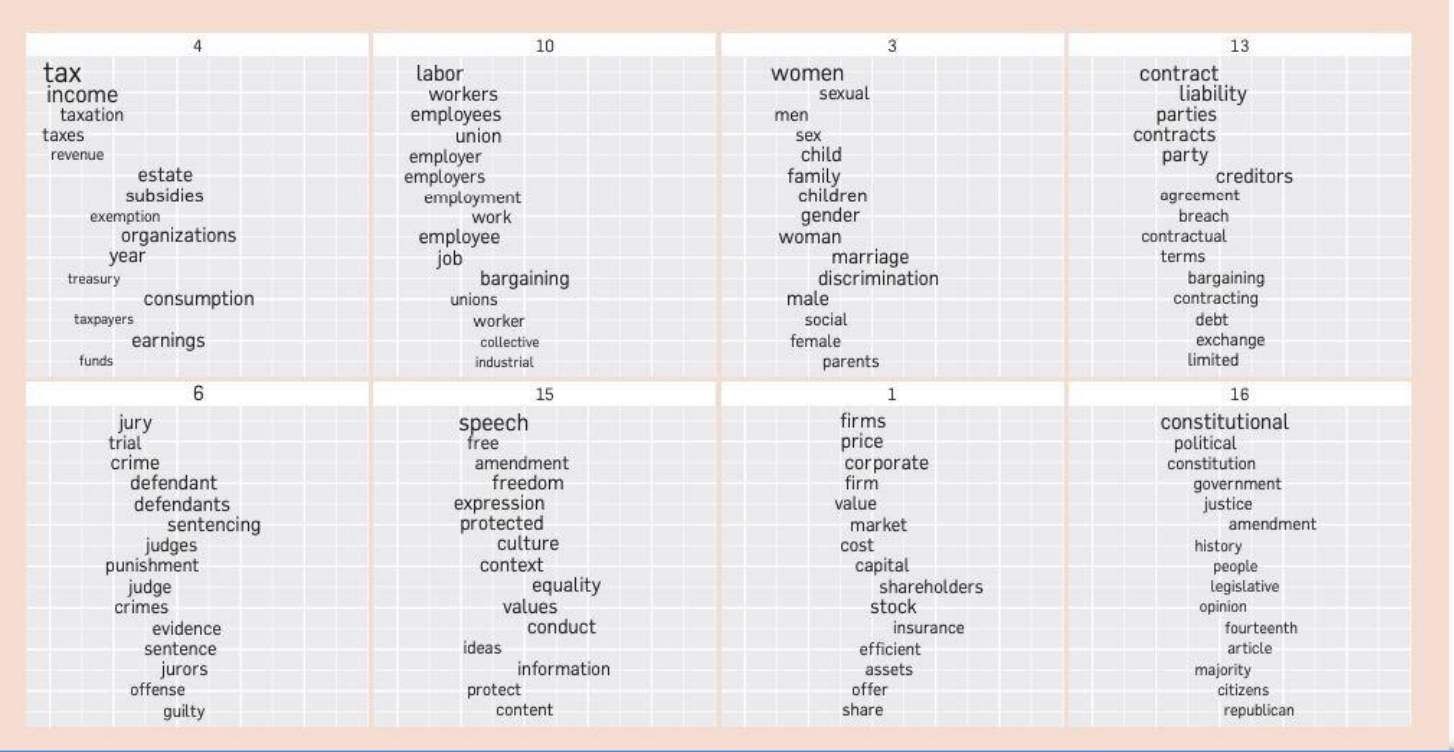


"Genetics"	"Evolution"	"Disease"	"Computers"
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Discovering topics…

- Algorithms have no information about the subjects of the documents
- The articles are **not** labelled with the topics or keywords
- Interpretable topic distributions arise by computing the hidden structure that most likely generated the observed documents

Tpic model fit to the Yale Law Journal



Generative Probabilistic Modelling for LDA

- LDA part of larger field of probabilistic modelling
- Treat data as arising from generative process including hidden variables
- Generative process defines a *joint probability distribution* over both the observed and hidden random variables
- We analyse data using joint distribution to calculate the *conditional (posterior) distribution* of the hidden variables given the observed variables
- Observed variables = words of the documents
- Hidden variables = topic structure

Formal Definition

Joint distribution of all random variables

Observed words in docs

Topic structure

$$p(\beta_{1:K}, \theta_{1:D}, z_{1:D} \mid w_{1:D}) = \frac{p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})}{p(w_{1:D})}$$

Evidence

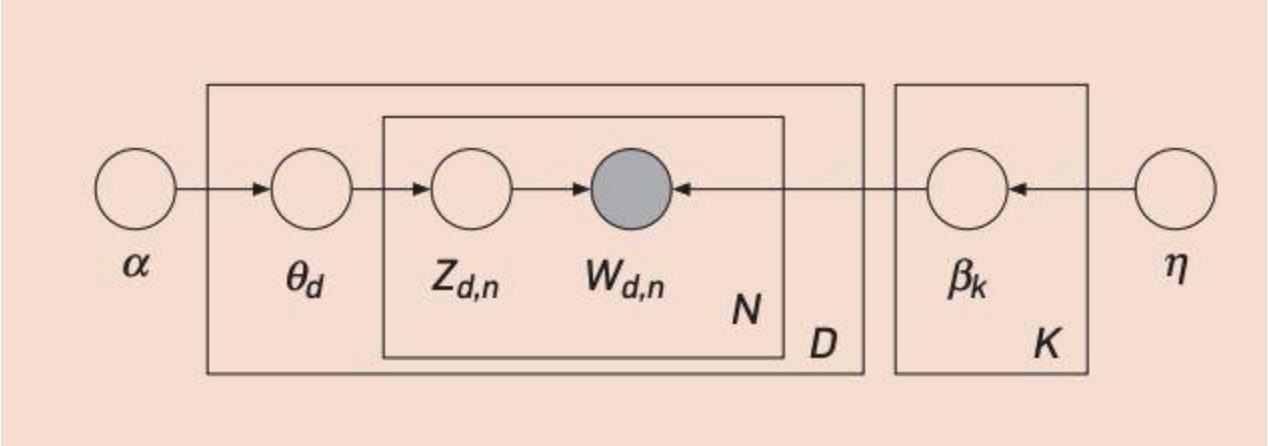
$\beta_{1:k}$ are the topics where each β_k is a distribution over the vocabulary

Topic proportions for the *d*th document are θ_d , where $\theta_{d,k}$ is the topic proportion for topic *k* in document *d*

$z_{1:D}$ are the topic assignments where $z_{d,n}$ is the topic assignment for the *n*th word in document *d*.

$W_{1:D}$ is the set of words where $w_{d,n}$ is the *n*th word in document *d*

Plate Notation for LDA



Each node is a random variable

Hidden nodes: topic proportions, assignment and topics are unshaded

Observed node: the words of the documents are shaded

Rectangles are the plate notation which denotes replication: N plate is the collection of words in the documents, D plate is the collection of document

Code Example

Summary

- We've looked at unsupervised machine learning framework
- No reliance on pre-labelled data
- Algorithms make attempts to partition data or uncover underlying patterns
- Useful where classes are hard to define or data changes over time, or there's just too much data!
- Good for providing novel insights about the data
- K-means clustering
- Evaluation
- Latent Dirichlet Allocation