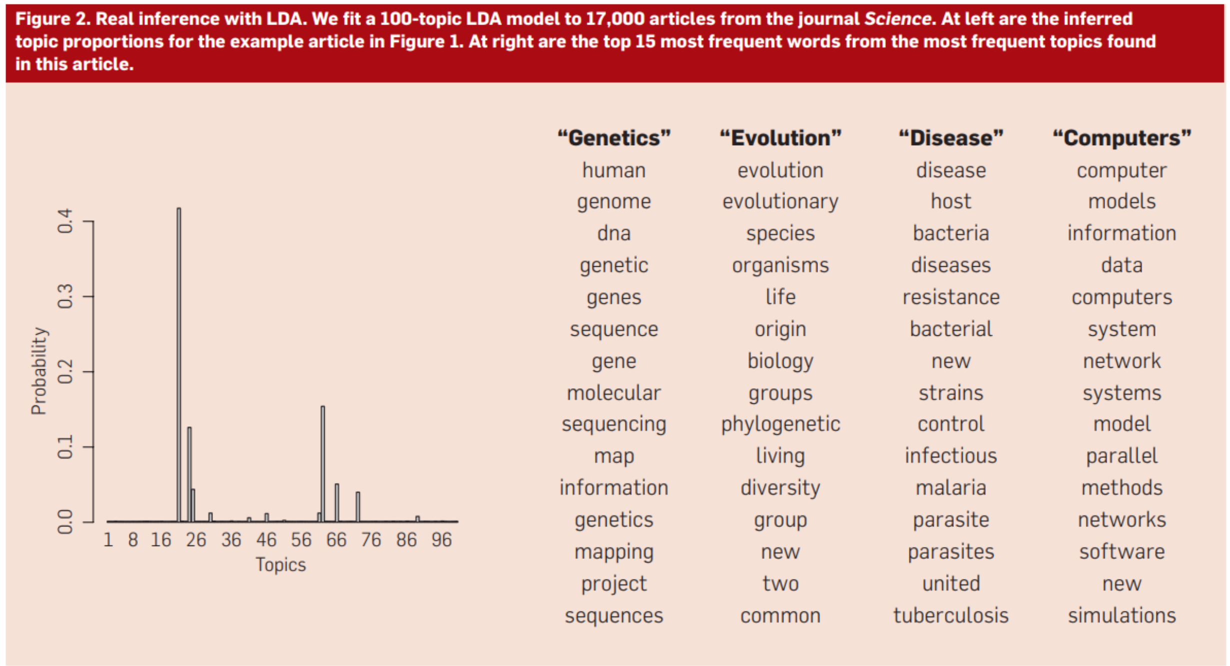


Figure 2. Real inference with LDA. We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.

“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

COMP90042 LECTURE 15

TOPIC MODELS

INTRODUCTION

- A librarian team just digitalised thousands of documents. Now they need to organise them.
- Twitter wants to find which topics are trending today. (half a billion tweets per day)
- A company wants to make sense of thousands of reports from its employees.
- Maybe we can use text classification?
  - Get some annotated data, train a classifier, predict classes for unseen texts, report.

INTRODUCTION

- A librarian team just digitalised thousands of documents. Now they need to organise them.
  - What about documents with multiple classes?
- Twitter wants to find which topics are trending today. (half a billion tweets per day)
  - What about new classes?
- A company wants to make sense of thousands of reports from its employees.
  - What **are** classes?

PROBLEMS WITH TEXT CLASSIFICATION

- Simple solutions based on text classification not feasible.
  - Documents can have multiple classes (multidisciplinary books, for instance).
  - Lack of annotated data.
  - Sometimes we don't even know which classes we expect from the data or we are interested into. Remember unsupervised information extraction (OpenIE)?

TOPICS

- We need a concept which is more open-ended than "classes": **topics**.
- A topic can have a specific semantic interpretation but does not have a label.
- Additionally, documents can have multiple topics.
- This concept enables us to perform open-ended, **unsupervised** learning on documents.
  - The standard algorithm for this is called Latent Dirichlet Allocation (LDA). We will derive it in this lecture.

TOPICS - EXAMPLE

- “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.”

TOPICS - EXAMPLE

- "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**."
- Topic 1: "genes", "genome", "genomes"
- Topic 2: "organism", "survive", "life", "organisms"
- Topic 3: "computer"
- **Key assumption:** words within a document have a single topic.

TOPICS - EXAMPLE

  - Topic 1: “genes”, “genome”, “genomes”
  - Topic 2: “organism”, “survive”, “life”, “organisms”
  - Topic 3: “computer”
- We can assign topics to documents based on topic frequency, for instance.
  - Document 1: topics 2 and 3
  - Document 2: topics 1 and 3
- What does topic “1” mean?
  - Need to inspect the words and **manually** assign semantics.
  - We will revisit this later.

TOPIC MODELLING

- “How many **genes** does an **organism** need to **survive**?”
- How/**0** many/**0** genes/**1** does/**0** an/**0** organism/**2** need/**0** to/**0** survive/**2** ?/**0**
- First idea: unsupervised HMMs
  - Split each document into sentences.
  - Assume, for instance, 100 topics. Initialise, run EM.
- Assign topics based on frequency.
  - Problem: this will likely assign the same topics to every document.
  - Limitation: a word’s topic is independent of the document.

TOPIC MODELLING

- Second idea: **per-document** unsupervised HMMs.
  - The topic distribution now varies for each document, which is exactly we are aiming for.
  - But this simple approach will not account for how topics relate across documents. Topic “4” in document “1” might be the same as topic “7” in document “5”. We need to **tie** the parameters across all documents.

TOPIC MODELLING

- Let’s formalise that:
  - $P(w = \text{gene}, t = 1 | d = 7) =$
  - $P(w = \text{gene} | t = 1, d = 7) \ * P(t = 1 | d = 7)$
- Two **key simplifications**:
- Emission probabilities are shared across documents
  - $P(w = \text{gene} | t = 1, d = 7) = P(w = \text{gene} | t = 1)$
- Topic of word does not depend on previous words’ topics. Only depends on the document.
  - $P(t = 1 | d = 7)$ becomes a single parameter.

TOPIC MODELLING

- $P(w = \text{gene}, t = 1 | d = 7) =$
  - $P(w = \text{gene} | t = 1, \beta_1) \; * P(t = 1 | \theta_7)$
- Parameters are now:
  - β (one per **topic**): the distribution of words given a topic
  - θ (one per **document**): the distribution of topics given a document
- We can use EM to train this. Given a topic initialisation:
  - β: count (expected) word-topic frequencies (in the whole corpus) and normalise
  - θ: count (expected) topic-document frequencies and normalise

EM FOR TOPIC MODELLING

- E-step
  - $P(t = 1 | w = \text{gene}, d = 7, \beta_1, \theta_7) = \frac{P(w{=}gene,\, t{=}1 | d{=}7,\, \beta_1,\, \theta_7)}{\sum_k P(w{=}gene{,}t{=}k | d{=}7,\, \beta_k,\, \theta_7)}$
- M-step
  - $\theta_7^1 = \frac{\sum_{i \in d} P(t{=}1 | w_i{,}\beta_1{,}\theta_7)}{\sum_k \sum_{i \in d} P(t{=}k | w_i{,}\beta_k{,}\theta_7)} \approx \frac{\#\text{ tokens in } d \text{ with topic } 1}{\#\text{ tokens in } d}$
  - $\beta_1^w = \sum_{d \in D} \frac{\sum_{i:\, w_i = w} P(t{=}1 | w_i{=}w{,}\beta_1{,}\theta_d)}{\sum_{v \in V} \sum_{i: w_i = v} P(t{=}k | w_i{=}v{,}\beta_1{,}\theta_7)}$
  - $\approx \frac{\#\text{ tokens with type } w \text{ and with topic } 1}{\#\text{ tokens with topic } 1}$

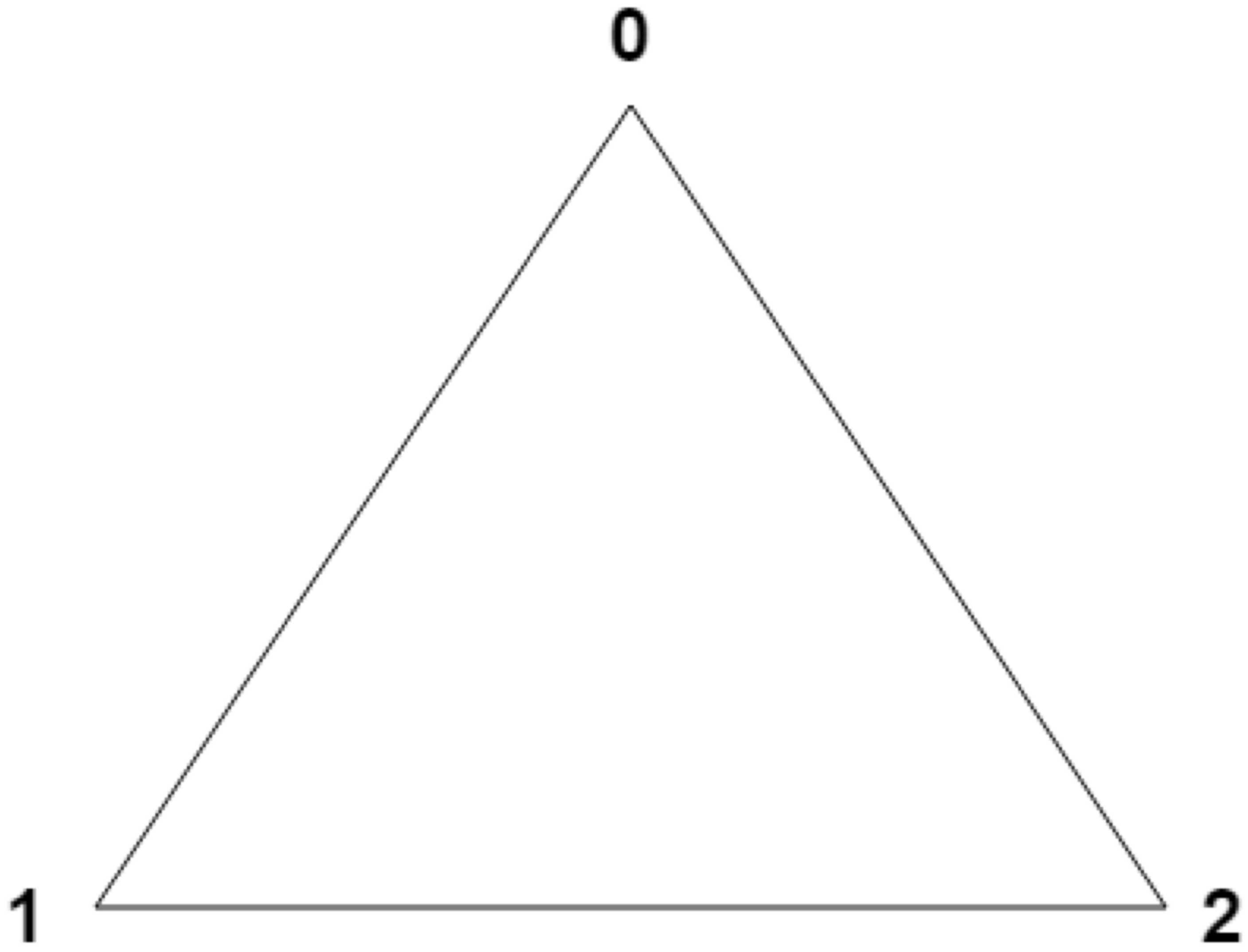
TOPIC MODELLING

- We are almost there! The previous model is very close to Latent Dirichlet Allocation.
- It is missing a key component that we saw in other probabilistic models before: **smoothing**.
- Let’s do add-k smoothing.

ADD-K SMOOTHING

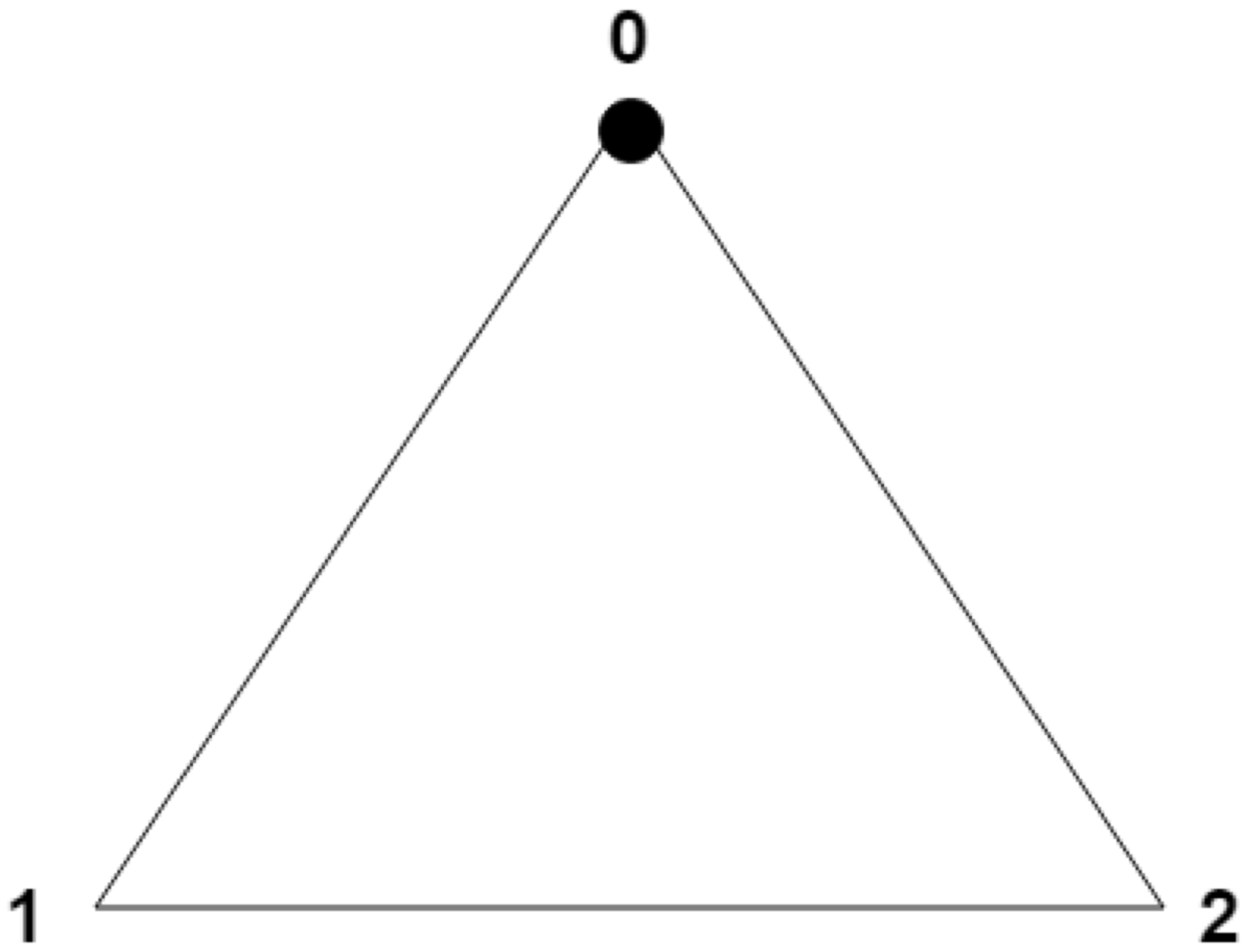
- E-step is the same
- M-step
  - $\theta_7^1 = \frac{\alpha + \sum_{i \in d} P(t=1|w_i,\beta_1,\theta_7)}{\sum_k(\alpha + \sum_{i \in d} P(t = k|w_i, \beta_k, \theta_7))}$
  - $\beta_1^w = \sum_{d \in D} \frac{\eta + \sum_{i:\, w_i=w} P(t=1|w_i=w,\beta_1,\theta_d)}{\sum_{v \in V}(\eta + \sum_{i:w_i=v} P(t = k|w_i = v, \beta_1, \theta_7))}$
- Add-k smoothing can be interpreted as having a **prior distribution** over the parameters θ and β
  - $P(\theta|\alpha) = \text{Dirichlet}(\alpha + 1)$
  - $P(\beta|\eta) = \text{Dirichlet}(\eta + 1)$

THE PROBABILITY SIMPLEX



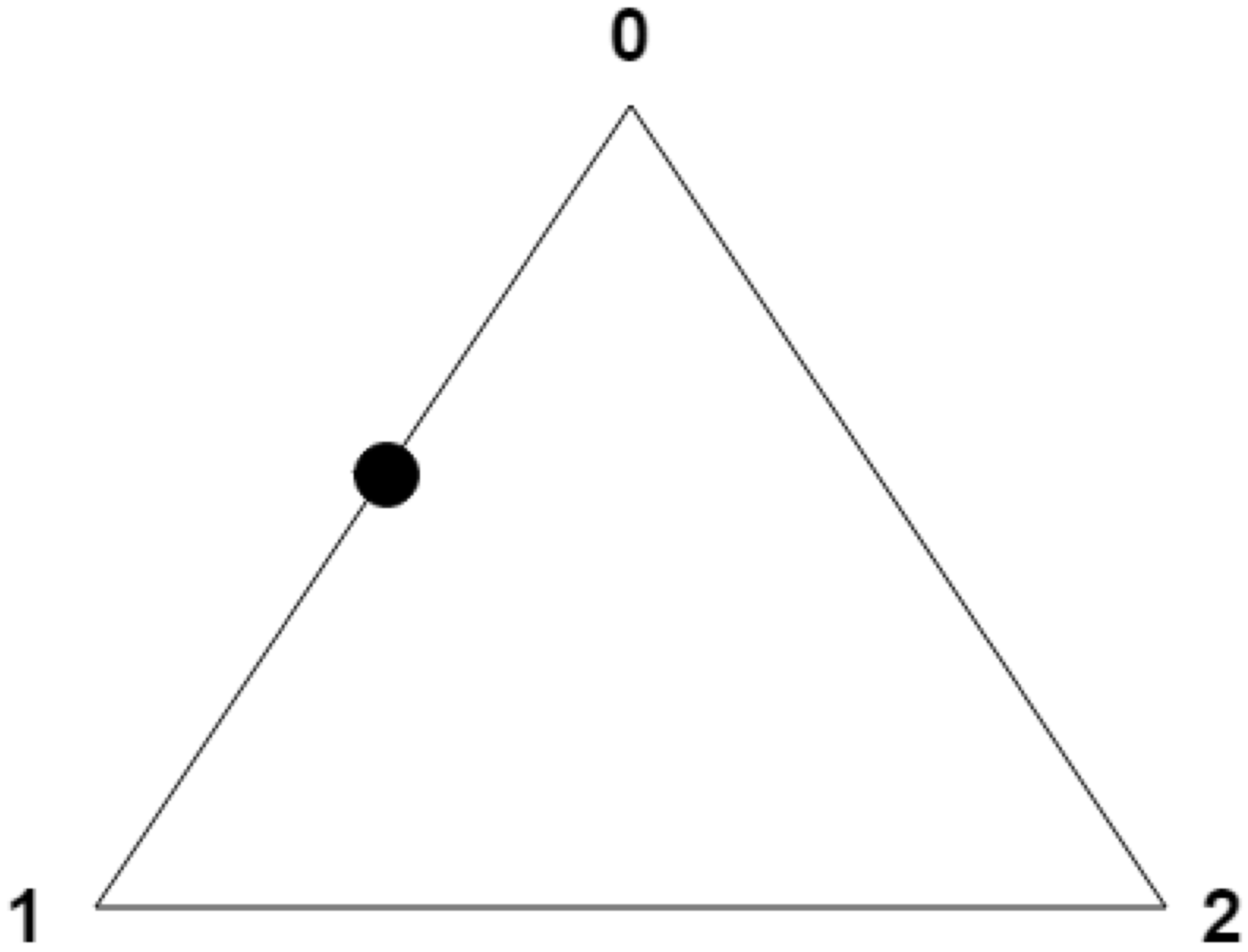
- Each point in the triangle is a probability distribution over 3 topics

THE PROBABILITY SIMPLEX



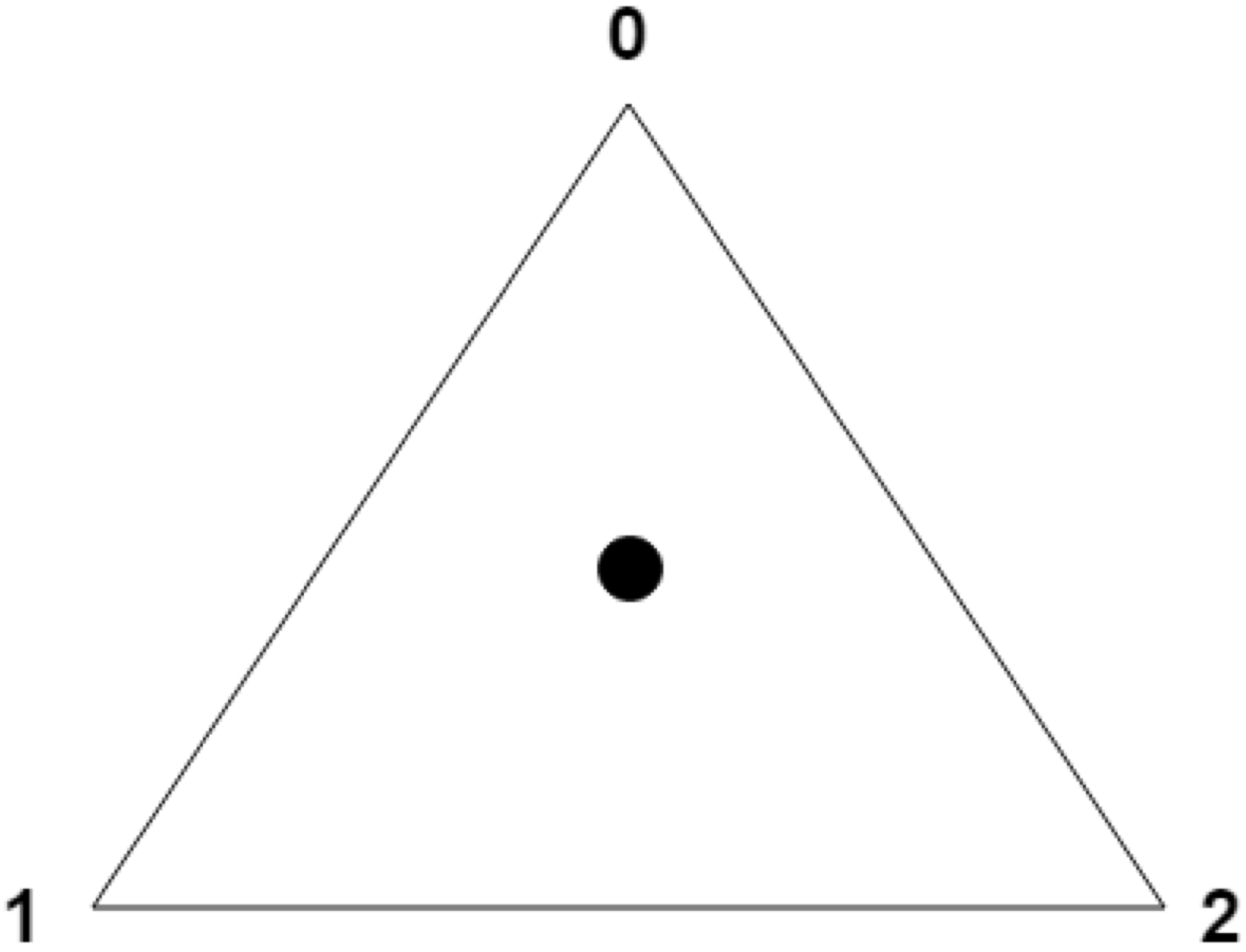
- $P(0) = 1, P(1) = 0, P(2) = 0$

THE PROBABILITY SIMPLEX



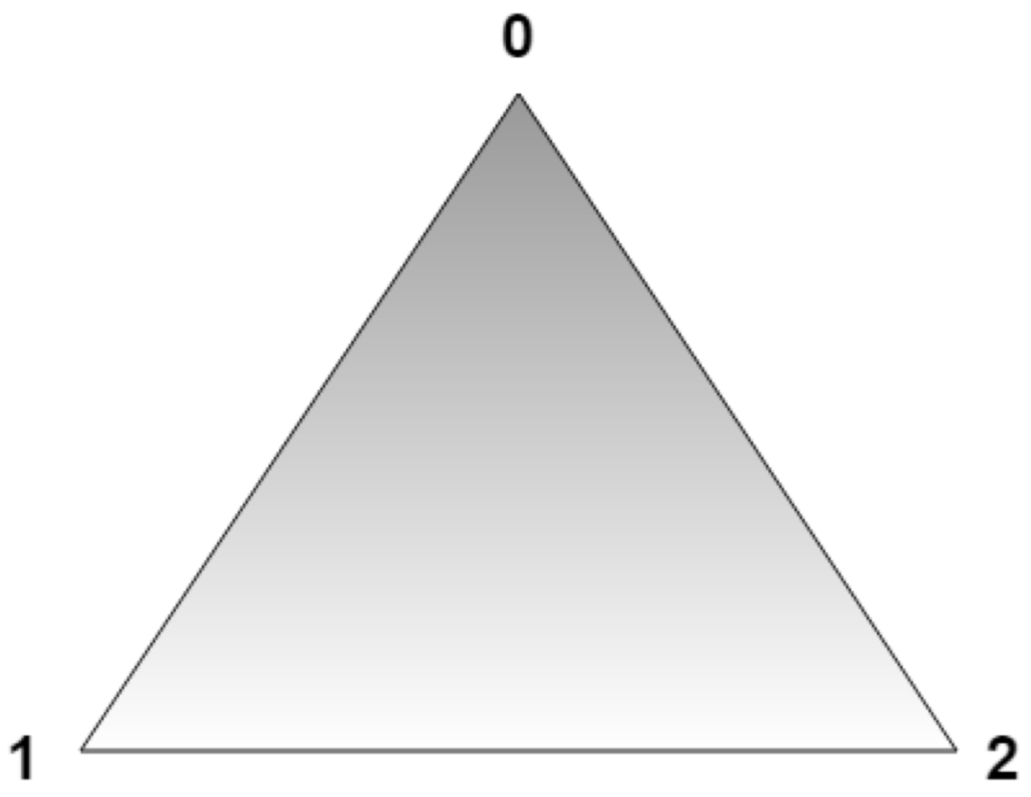
- $P(0) = 0.5,\ P(1) = 0.5,\ P(2) = 0$

THE PROBABILITY SIMPLEX



- P(0) = 0.333, P(1) = 0.333, P(2) = 0.333

THE DIRICHLET DISTRIBUTION



- The Dirichlet is a distribution over a probability simplex.
  - α defines the spread
  - Symmetric: α is a scalar
  - Asymmetric: one α per topic

LATENT DIRICHLET ALLOCATION

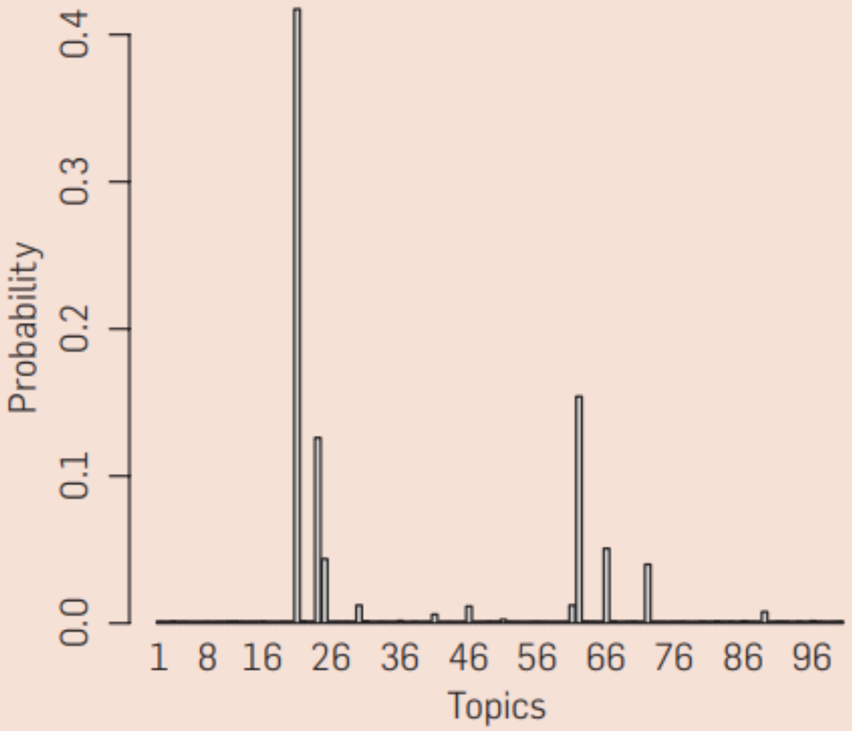
- The model we saw before is essentially LDA with symmetric Dirichlet over topics and words
  - Parameters can be estimated via EM
- In practice:
- Use asymmetric Dirichlet
- Instead of finding the maximum value for θ and β, estimate the **posterior** distribution over the parameters
  - Harder to do but prevents overfitting
  - Usually done via Gibbs sampling or variational inference: we will not see these techniques here… ☹

EVALUATING TOPIC MODELS

- Intrinsic: perplexity.
  - The model defines a distribution over each word, same as a language model.
  - We can split the corpus into a training and test set and evaluate perplexity on the test set.
- Extrinsic is harder.
  - If the topic model is used as a tool for an end task (information retrieval, machine translation, etc.), calculate the end task metric.
  - Otherwise some human intervention is required. Interpretability is key.

EVALUATING TOPIC MODELS

Figure 2. Real inference with LDA. We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.

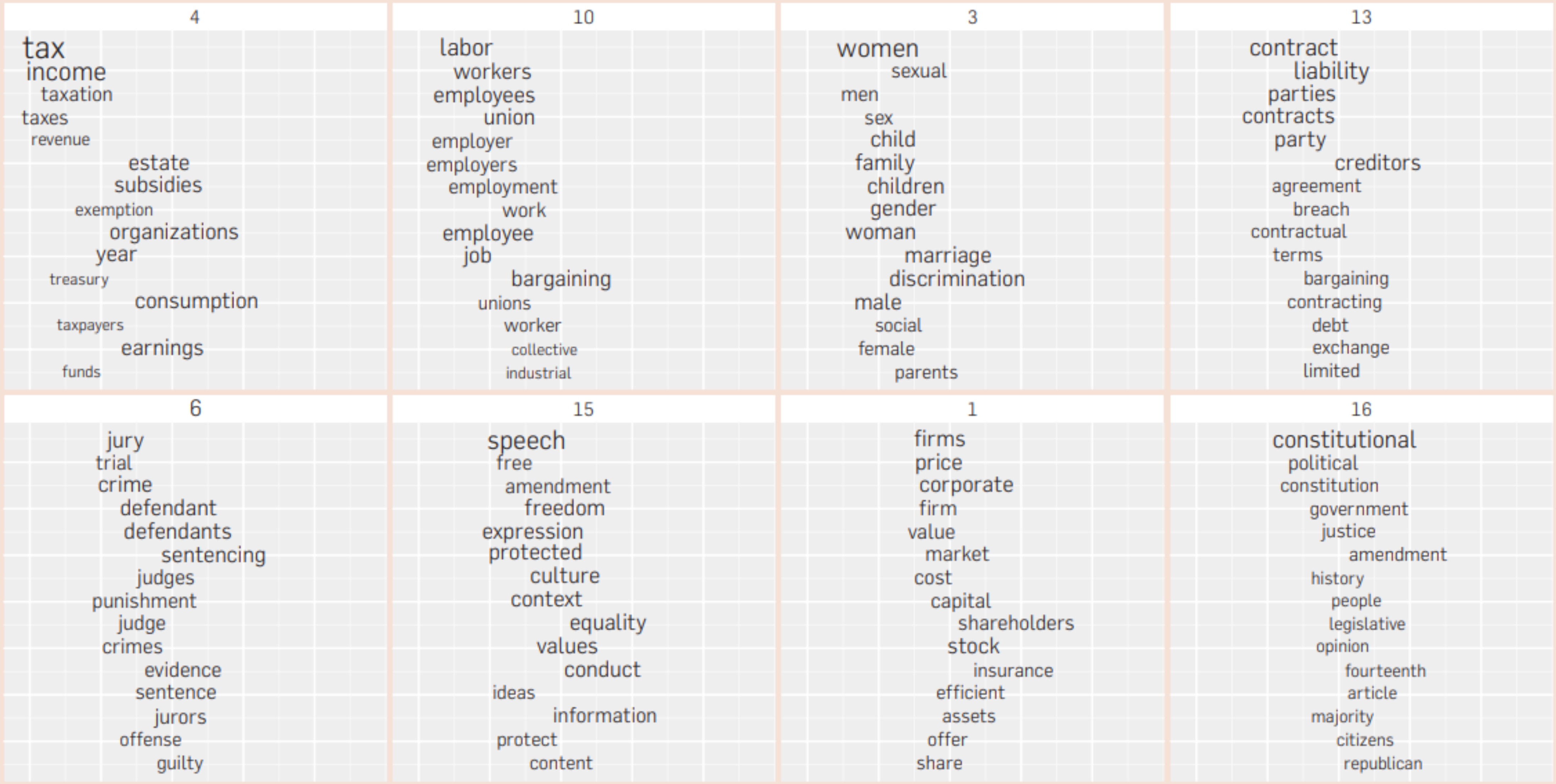


“Genetics”	**“Evolution”**	**“Disease”**	**“Computers”**
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	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

Blei
(2012)

EVALUATING TOPIC MODELS

Figure 3. A topic model fit to the *Yale Law Journal*. Here, there are 20 topics (the top eight are plotted). Each topic is illustrated with its top-most frequent words. Each word's position along the *x*-axis denotes its specificity to the documents. For example "estate" in the first topic is more specific than "tax."



Blei
(2012)

EVALUATING TOPIC MODELS

- Many other visualisation options available
  - Word clouds
  - Word graphs
- Labelling techniques
  - Distant supervision (Wikipedia, knowledge bases, etc.)
  - Article names, pictures.
- Combination of all above
  - Goal: enhance interpretability

LDA EXTENSIONS

- LDA is the standard method for topic modelling. It has been vastly studied in academia and many extensions were proposed.
- LDA-HMM: nearby words have similar topics
- Hierarchical LDA: assumes a topic hierarchy
- Correlated LDA: some topics are more similar to others (“football” and “rugby” vs. “football” and “genetics”)
- Dynamic LDA: topics change over time, need to know document ordering (from timestamps, for instance)
- Non-parametric LDA: does not assume a fixed number of topics (harder to train)

APPLICATIONS

- Information retrieval: useful for query expansion.
- Analysis of historical documents (newspapers, books).
- Making sense of scientific publications: emergence of new fields and multidisciplinary ones.
- Literary analysis: stylometry, comparative literature.
- Computational social science: text on social media (Twitter), stance detection.
- Translation: multilingual topic models for mining parallel data.
- …

A FINAL WORD

- Topic models aim at making sense of large collections of documents, combining two ideas:
  - Unsupervised learning (clustering)
  - Documents can have multiple topics
- LDA is the standard method
  - Can use EM to train via MAP
  - In practice, estimate posteriors using sampling or other techniques
- Perplexity can give evidence of performance but ultimate goal in clustering is interpretability

ADDITIONAL READING

- David Blei's ACM paper (http://www.cs.columbia.edu/~blei/papers/Blei2012.pdf)
- (Optional) "Applications of Topic Models"
  - https://mimno.infosci.cornell.edu/papers/2017_fntir_tm_applications.pdf