Topic Models

Oier Lopez de Lacalle

April 2021

(Slides from David Blei and Mirella Lapata)

Why Topic Modeling

Information overload

- As more information becomes available, it becomes more difficult to find and discover what we need

Why Topic Modeling

Information overload

- As more information becomes available, it becomes more difficult to find and discover what we need

Main Tools: Search and Links

1. Type keyword into search engine and retrieve related documents
2. Look at the documents and navigate to the other documents

Why Topic Modeling?

Theme based search

- Imagine searching and exploring documents based on themes that run through them.
- We might "zoom-in" or "zoom-out" to find specific or broader themes
- We might look at how themes change through time, how they are connected to each other
- Find the theme first and then examine the documents pertaining to that theme

Topic Models

When we talk about topic models ...

- We want to **find themes** (or topics) in documents which are useful for e.g., search or browsing.
- We do **not** want to do **supervised topic classification** (neither fix topics in advance nor do manual annotation).
- Approach must **automatically tease out** the topics.
- Essentially a *clustering problem*: both words and documents are being clustered.
- Tackle the problem of **flat vectorial representation**.

Topic Models

When we talk about topic models ...

- We want to **find themes** (or topics) in documents which are useful for e.g., search or browsing.
- We do **not** want to do **supervised topic classification** (neither fix topics in advance nor do manual annotation).
- Approach must **automatically tease out** the topics.
- Essentially a *clustering problem*: both words and documents are being clustered.
- Tackle the problem of **flat vectorial representation**.

Provide methods for automatically organizing, understanding, searching and summarizing large electronic archives without any prior annotation or labeling.

Applications: Organize your documents

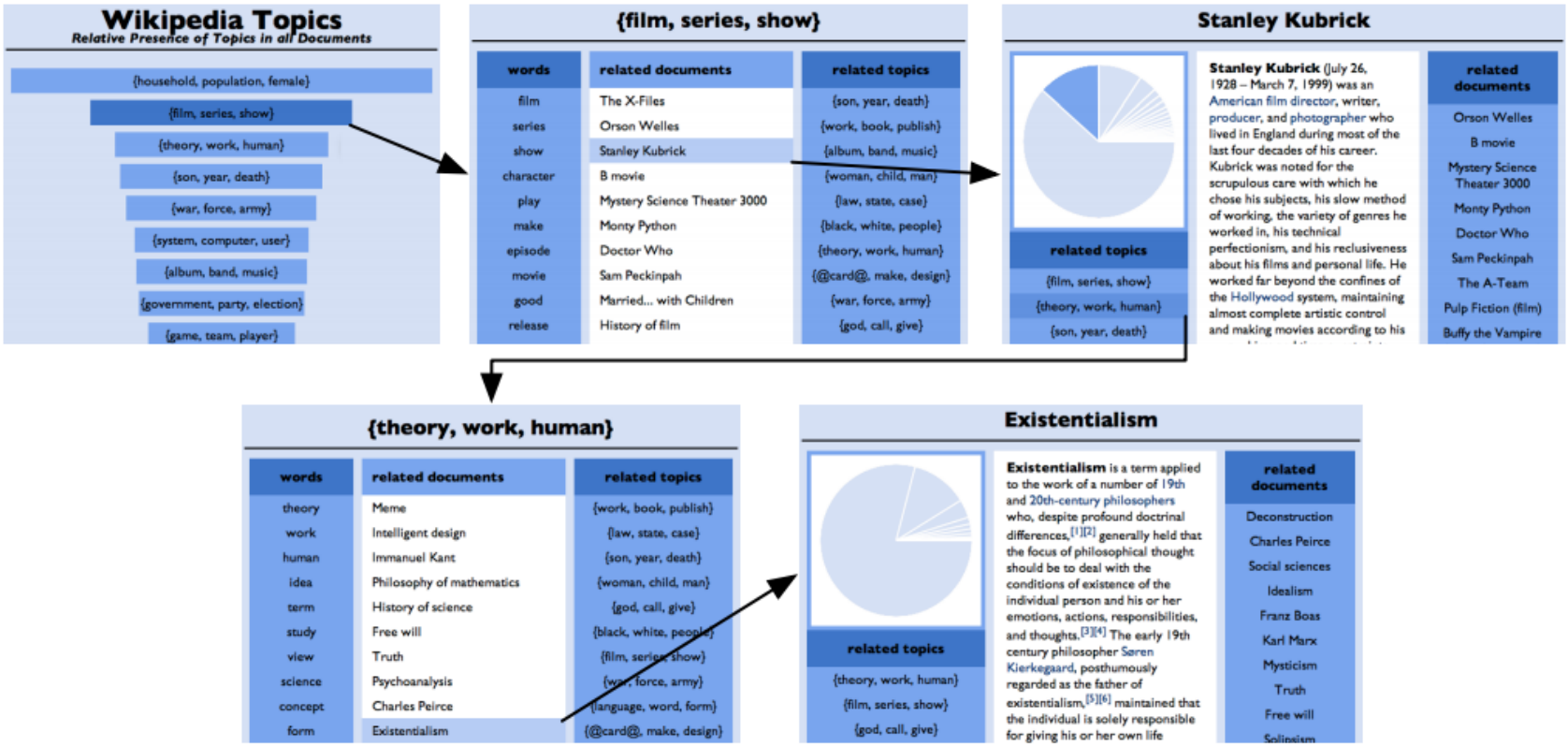
There are great visualization tool based on Topic Models that help **organizing** data and **discovering** things:

- A browser of 100K Wikipedia articles:
  - Code+tutorial: `https://github.com/ajbc/tmve-original`
  - Visualization: `http://www.princeton.edu/~achaney/tmve/wiki100k/browse/topic-presence.html`

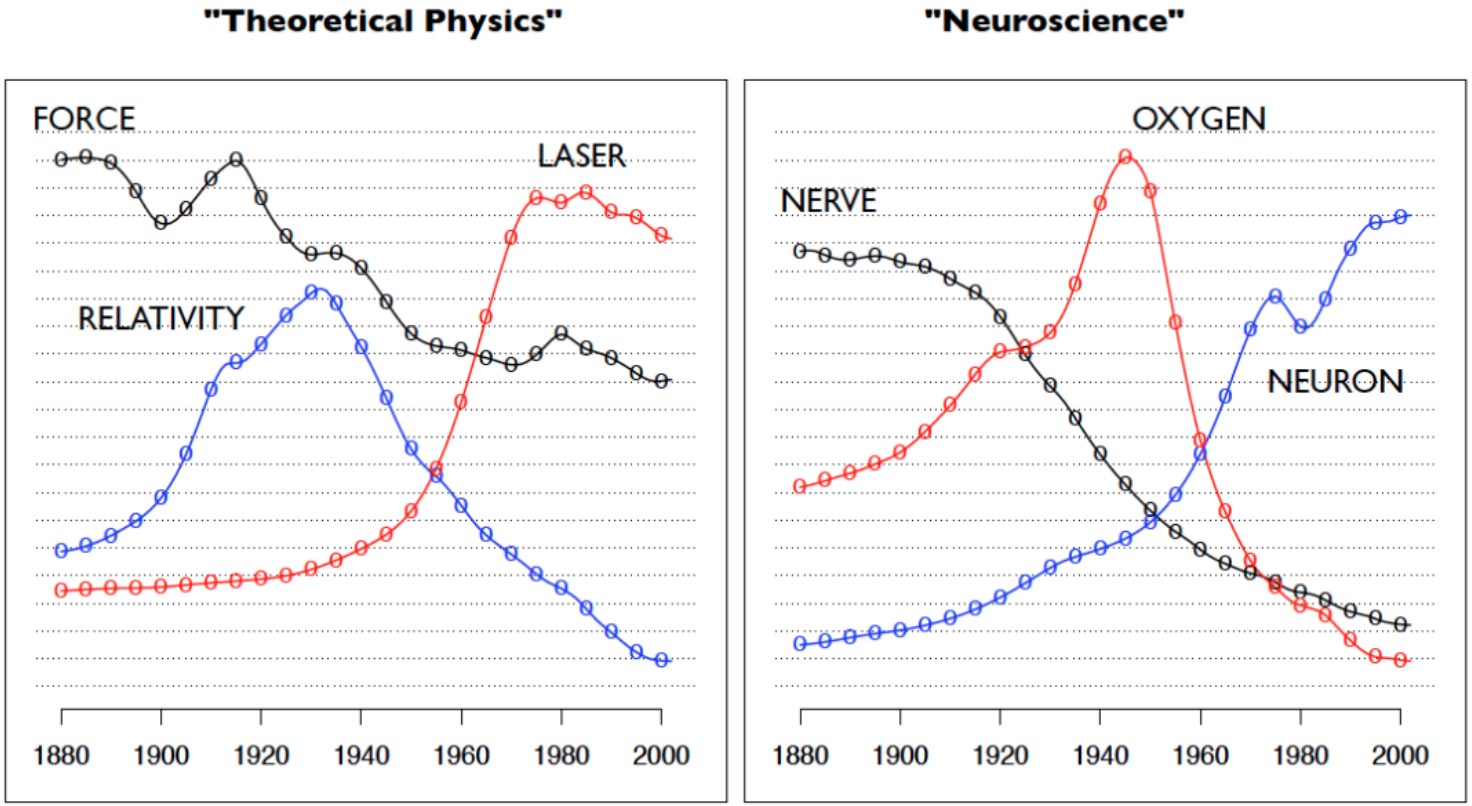
- Discover biology related topics (from wikipedia articles):
  - Code+tutorial: `https://github.com/cpsievert/LDAvis`
  - `http://bl.ocks.org/oierldl/raw/d93656d1683ad278cedf2a1e8c9a9590/`

Applications: Organize your documents

- A browser of 100K Wikipedia articles



Applications: Evolution of topics over time



Reading

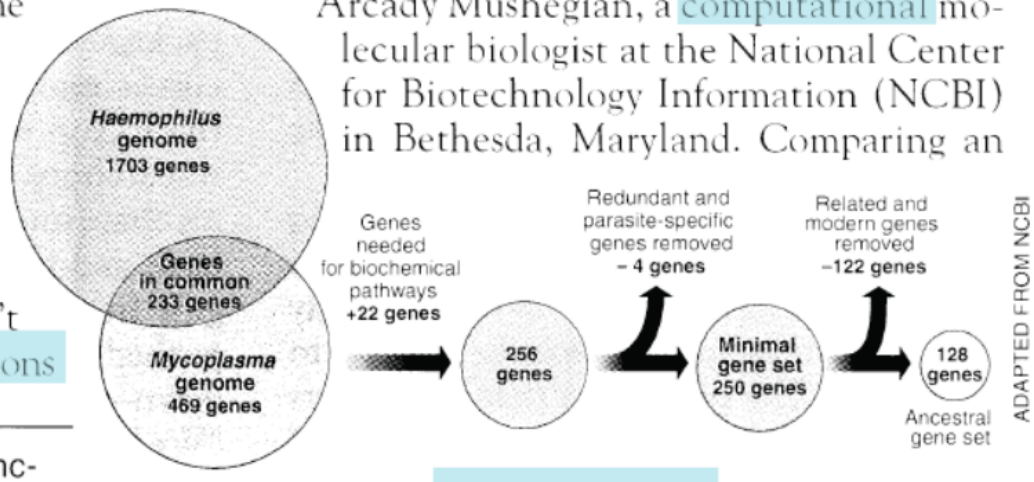
- All about topic modeling: `http://www.cs.columbia.edu/~blei/topicmodeling.html`
- **Introduction to Topic Modeling**
  David M. Blei. 2012. Probabilistic topic models. Communications of the ACM 54(5): 77–84
- **Application of Topic Models**
  Jordan Boyd-Graber, Yuening Hu, and David Mimno. Applications of Topic Models. Foundations and Trends in Information Retrieval, Now Publishers. 2017.
  `https://mimno.infosci.cornell.edu/papers/2017_fntir_tm_applications.pdf`

Intuition

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

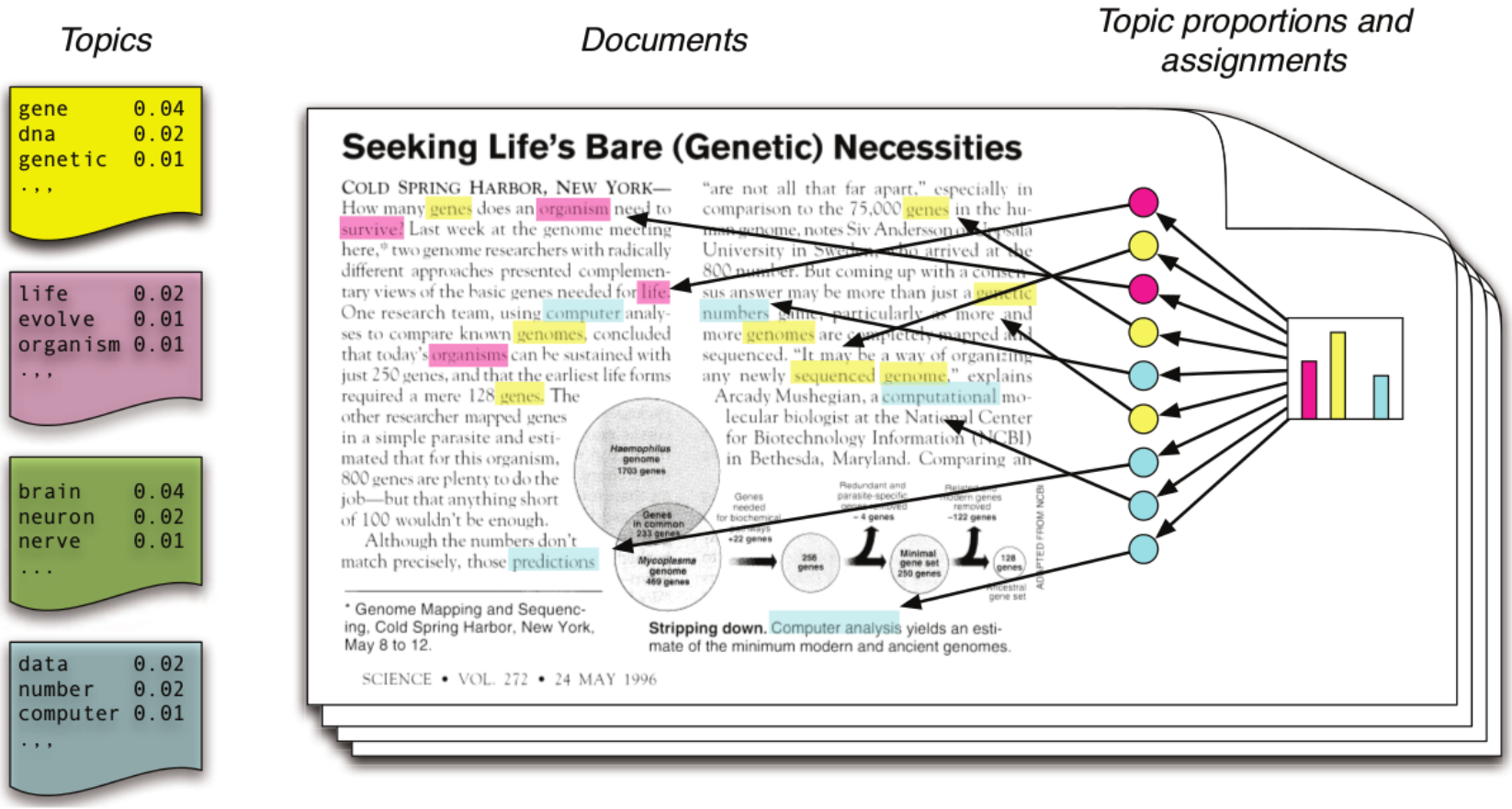


Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

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

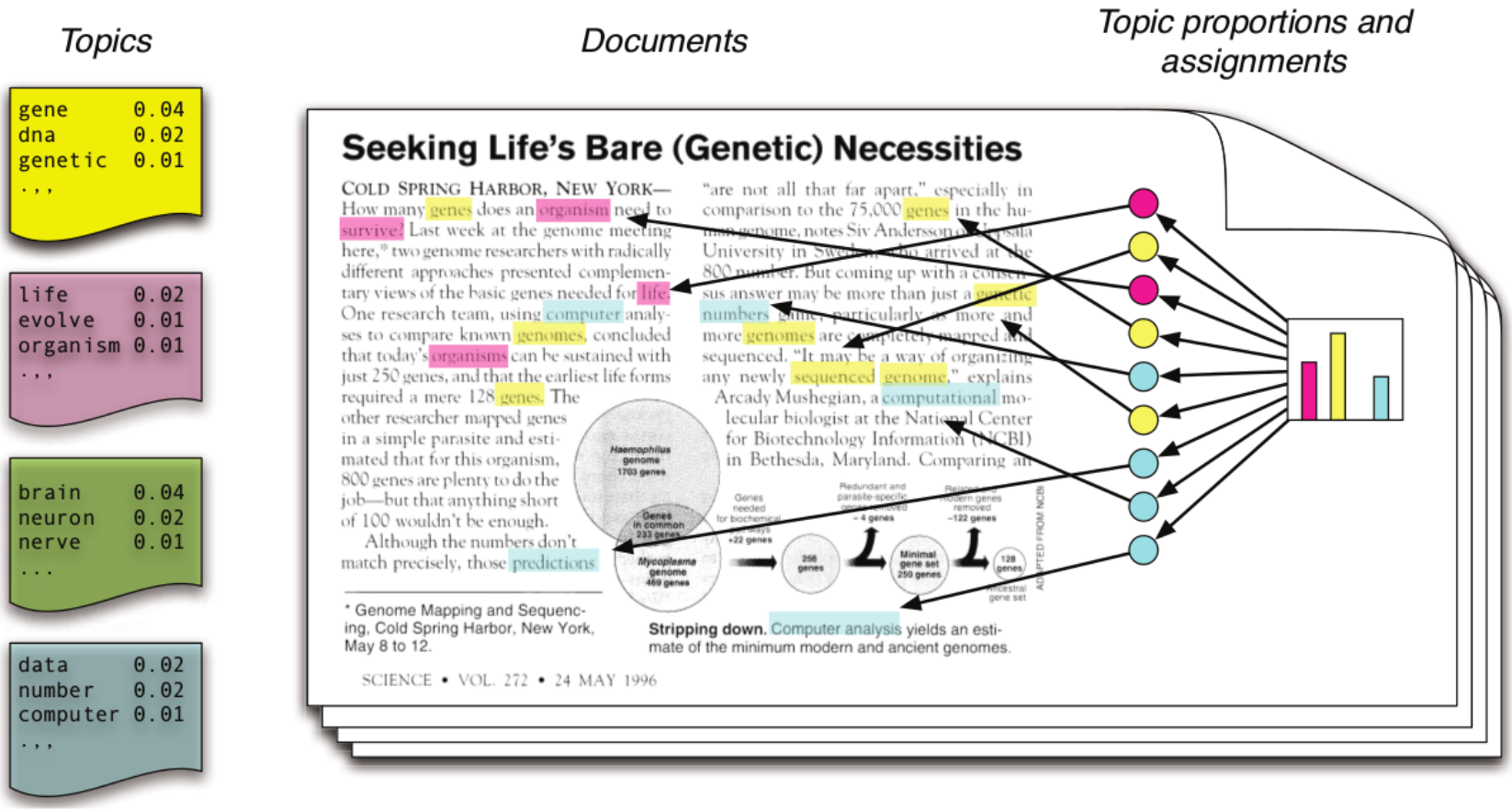
SCIENCE • VOL. 272 • 24 MAY 1996

Topic Models: Basic idea



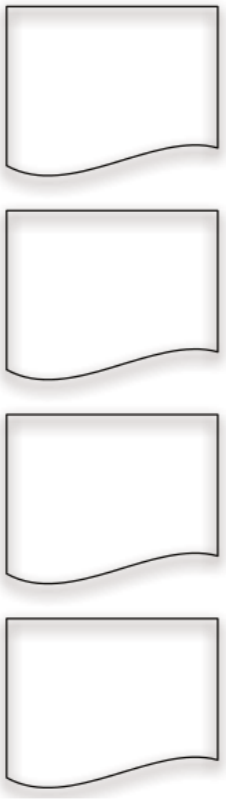
Simple intuition: documents exhibit multiple topics.

Latent Dirichlet Allocation



- Each topic is a distribution over words
- Each document is a mixture of corpus-wide topics

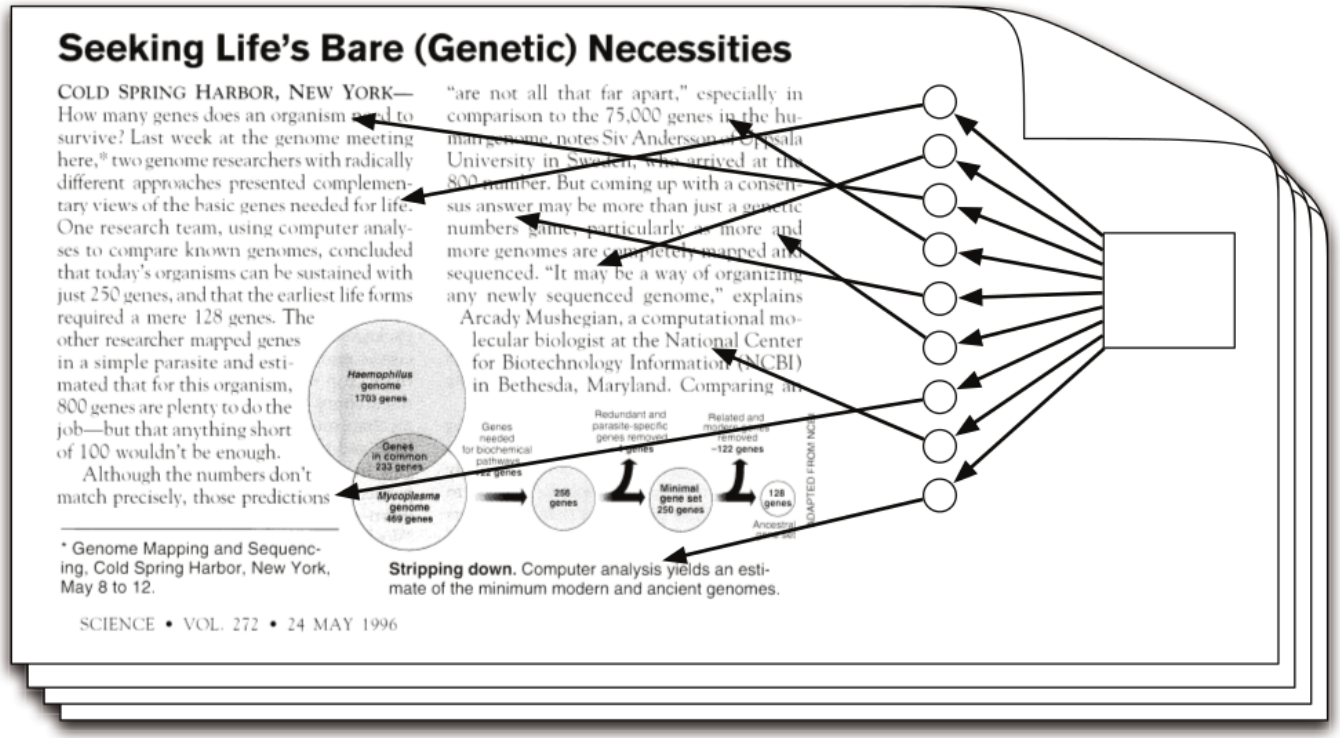
The Posterior Distribution



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

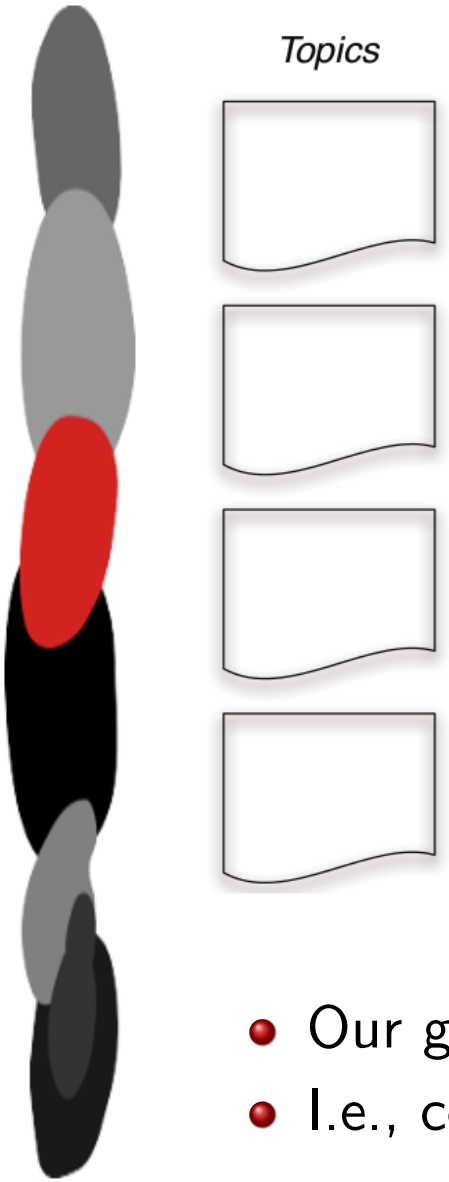
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

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

SCIENCE • VOL. 272 • 24 MAY 1996

- In reality, we only observe the documents
- The other structures are **hidden variables**

The Posterior Distribution



- Our goal is to **infer** the hidden variables
- I.e., compute their distribution conditioned on the documents

$$p(topics, proportions, assignments|documents)$$

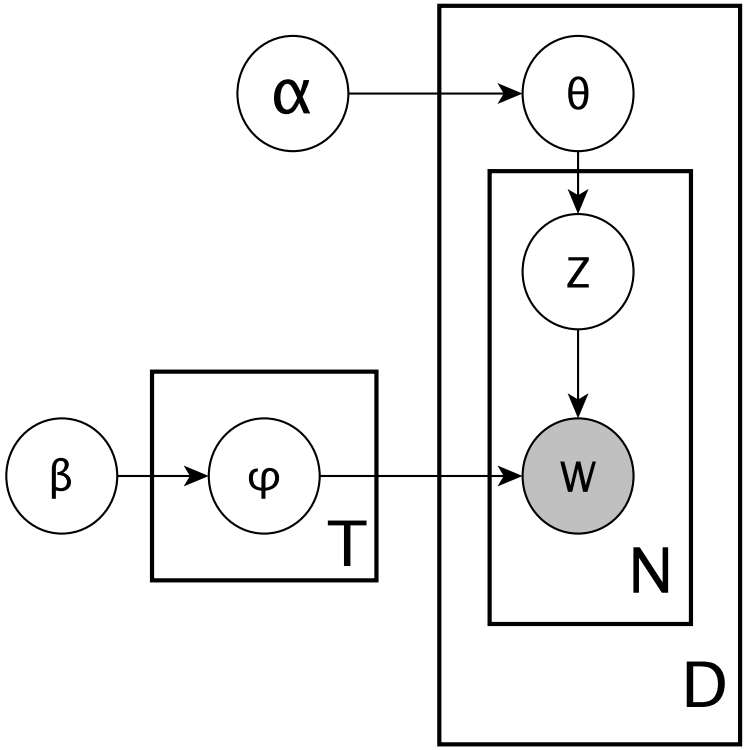
LDA: Key Assumption

- Documents exhibit multiple topics (but typically not many)
- LDA is a probabilistic model with a corresponding generative process (each document is generated by this process)
- A topic is a distribution over a fixed vocabulary (topics are assumed to be generated first, before the documents)
- Only the number of topics is specified in advance

Generative Process

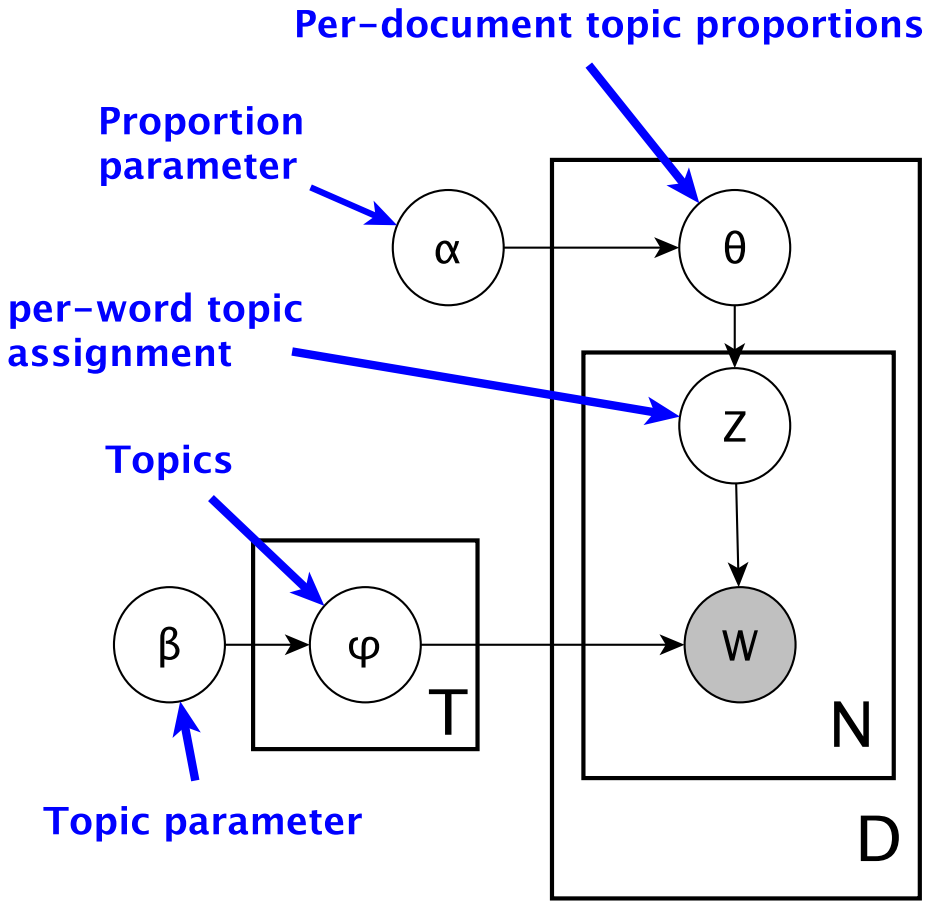
1: **for** $j = 1$ to T **do**
2: Choose $\phi^{(j)} \sim$ Dirichlet(β) $\{\phi_1^{(j)} \ldots \phi_V^{(j)}$: *prb.of each wd. in topic j*$\}$
3: **end for**
4: **for** $d = 1$ to D **do**
5: Choose $\theta^{(d)} \sim$ Dirichlet(α) $\{\theta_1^{(d)} \ldots \theta_T^{(d)}$: *prb.of each topic in doc d*$\}$
6: **for** $i = 1$ to N_d **do**
7: Choose $z_i \sim$ Multinomial(θ_j) $\{z_i$: *topic of word i*$\}$
8: Choose $w_i \sim$ Multinomial(ϕ_{z_i}) $\{w_i$: *form of word i*$\}$
9: **end for**
10: **end for**

Graphical model



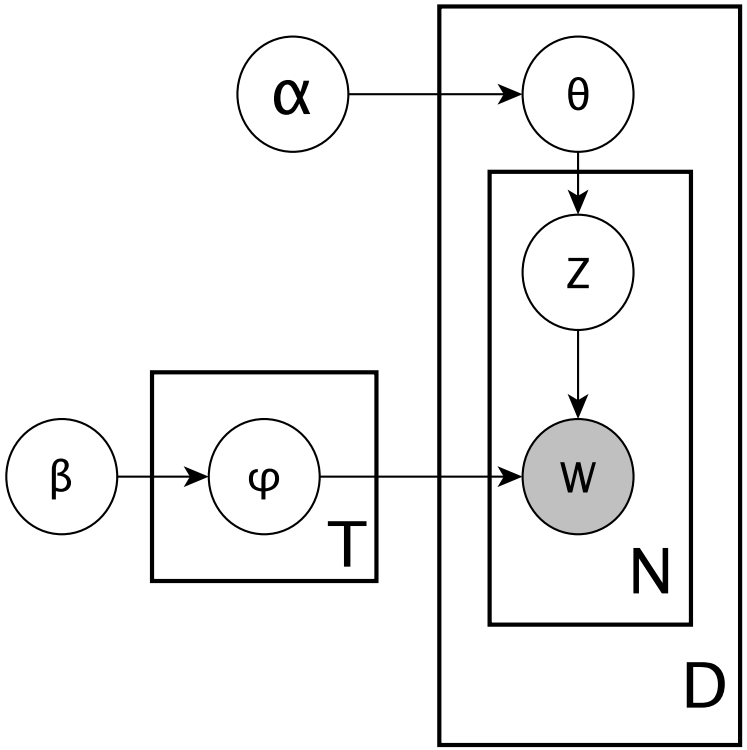
- Nodes are random variables; edges indicate dependence.
- Shaded nodes are observed; plates ≈ replicated variables.

Graphical model



- Nodes are random variables; edges indicate dependence.
- Shaded nodes are observed; plates ≈ replicated variables.

Graphical model



$$P(w_i) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_j = j)$$

Aside: Multinomial Distribution

(models the probability of counts for rolling a d sided dice n times)
For $x_i \in 0, \ldots, n$

$$P(\boldsymbol{x}|\theta) = \frac{n!}{\prod_{i=1}^{d} x_i!} \prod_{i=1}^{d} \theta_i^{x_i}, n = \sum_{i=1}^{d} x_i, \sum_{i=1}^{d} \theta = 1, \theta_i > 0$$

When n = 1 the multinomial distribution simplifies to:

$$P(\boldsymbol{x}|\theta) = \prod_{i=1}^{d} \theta_i^{x_i}, n = \sum_{i=1}^{d} x_i, \sum_{i=1}^{d} \theta = 1, \theta_i > 0$$

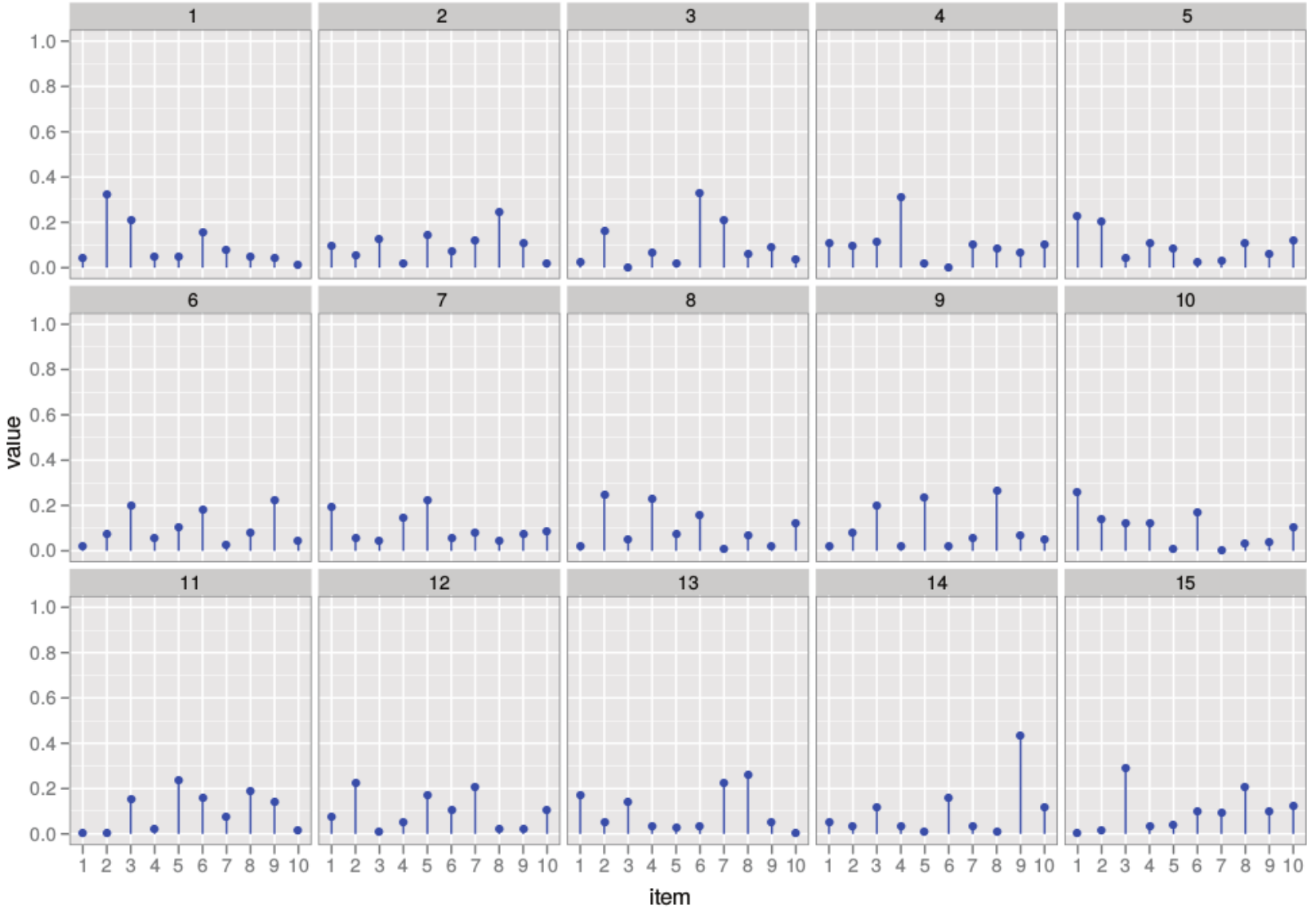
- **Unigram language model (Bag-of-Word model)**:
  - 1-of-V coding ($d = V$ vocab size)
  - x_i indicates word i of the vocabulary observed ($x_i = 1$ if word i is observed and 0 otherwise)
  - $\theta_i = P(w_i)$ the probability that word i is seen

Dirichlet Distribution

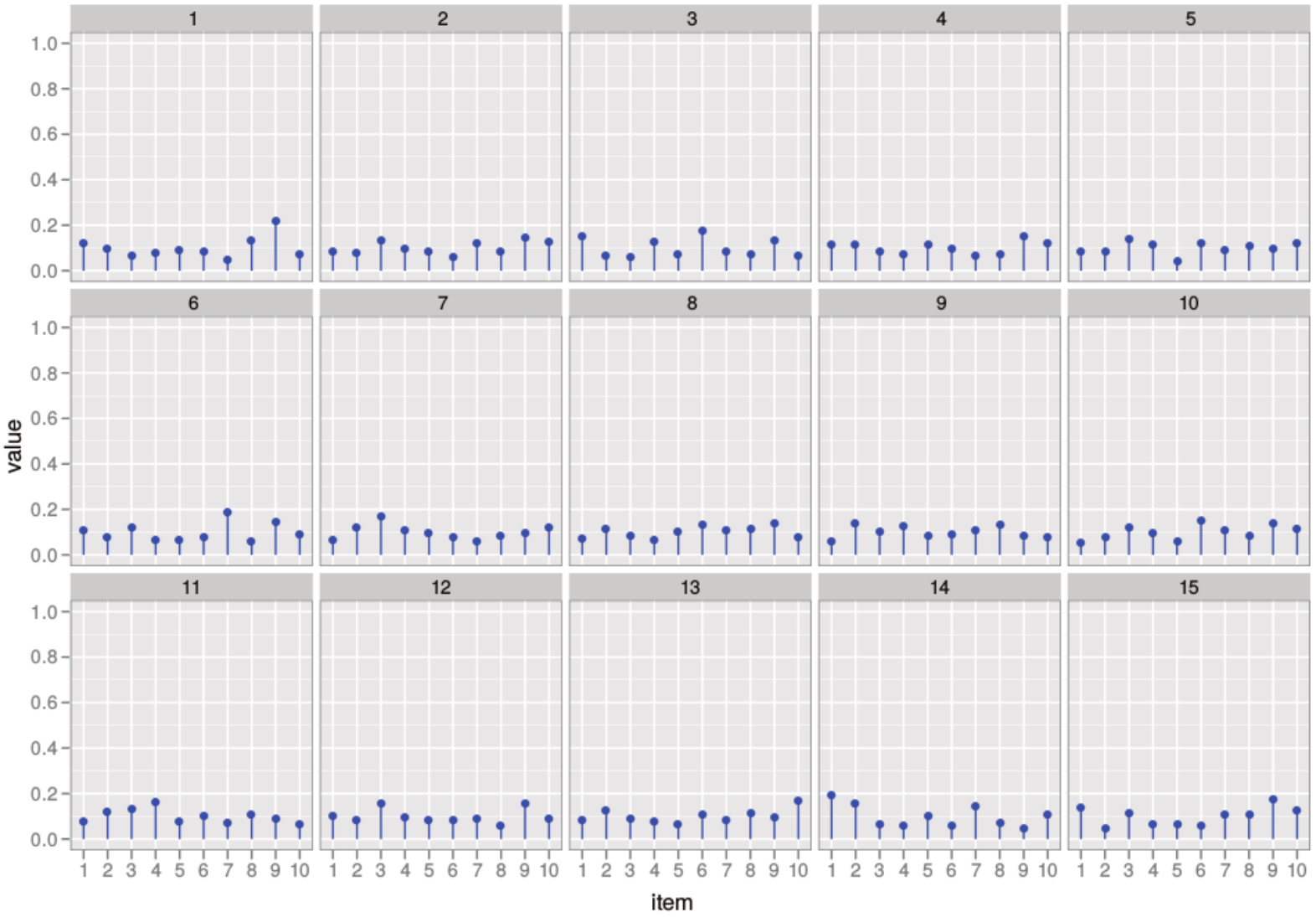
- The Dirichlet distribution is an exponential family distribution over the simplex, i.e., positive vectors that sum to one.
- A draw from a Dirichlet is a (discrete) multinomial distribution with probability one.
- That is, it is a distribution over distributions.

- It is **conjugate to the multinomial**. Given a multinomial observation θ, the posterior distribution of θ is a Dirichlet.
- The hyperparameter governs the behavior of the distribution, i.e. how the multinomial distributions drawn will look like
  - The parameter α smoothes the topic distribution in the document.
  - The parameter β smoothes the word distribution in every topic

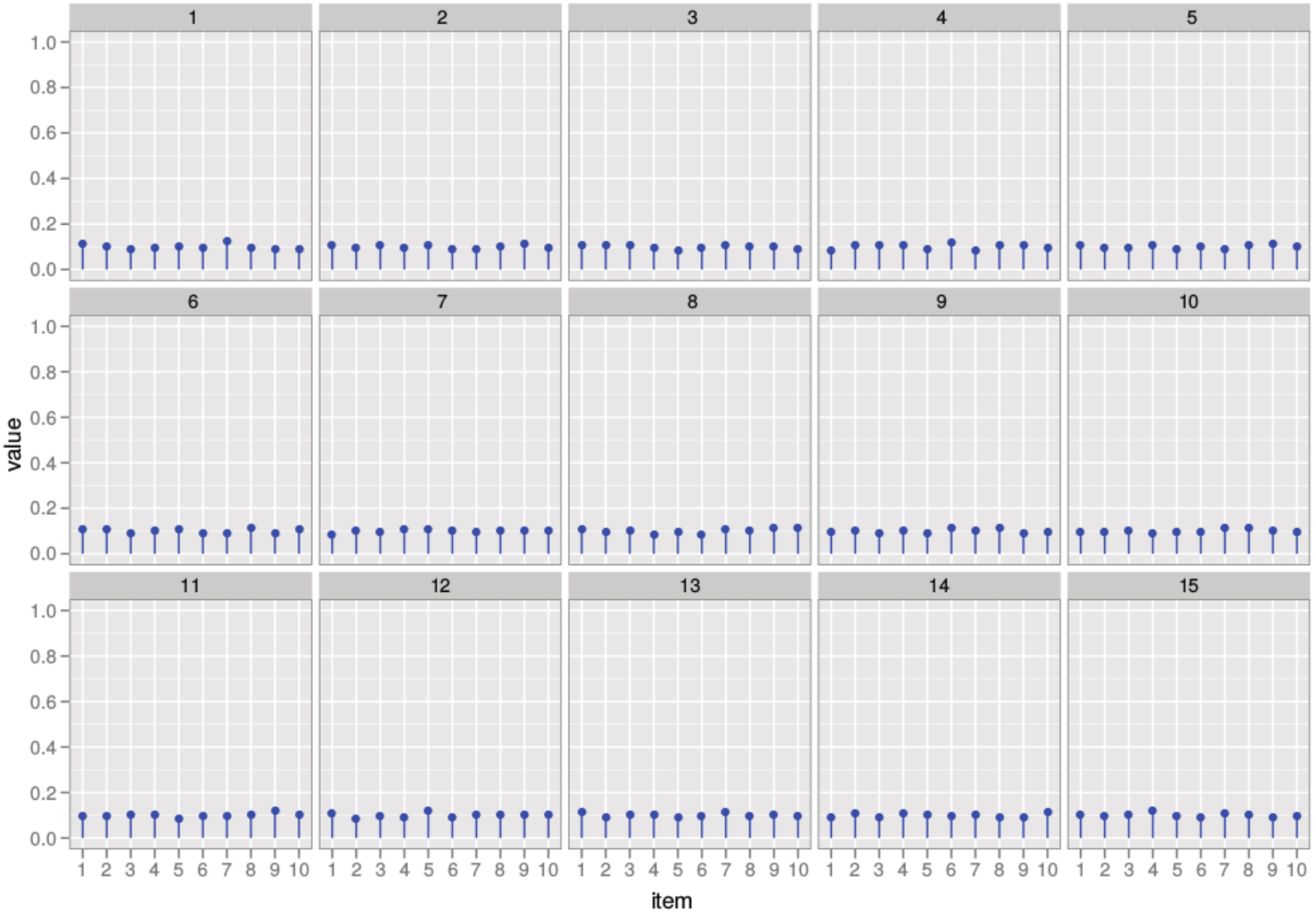
$\alpha = 1$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

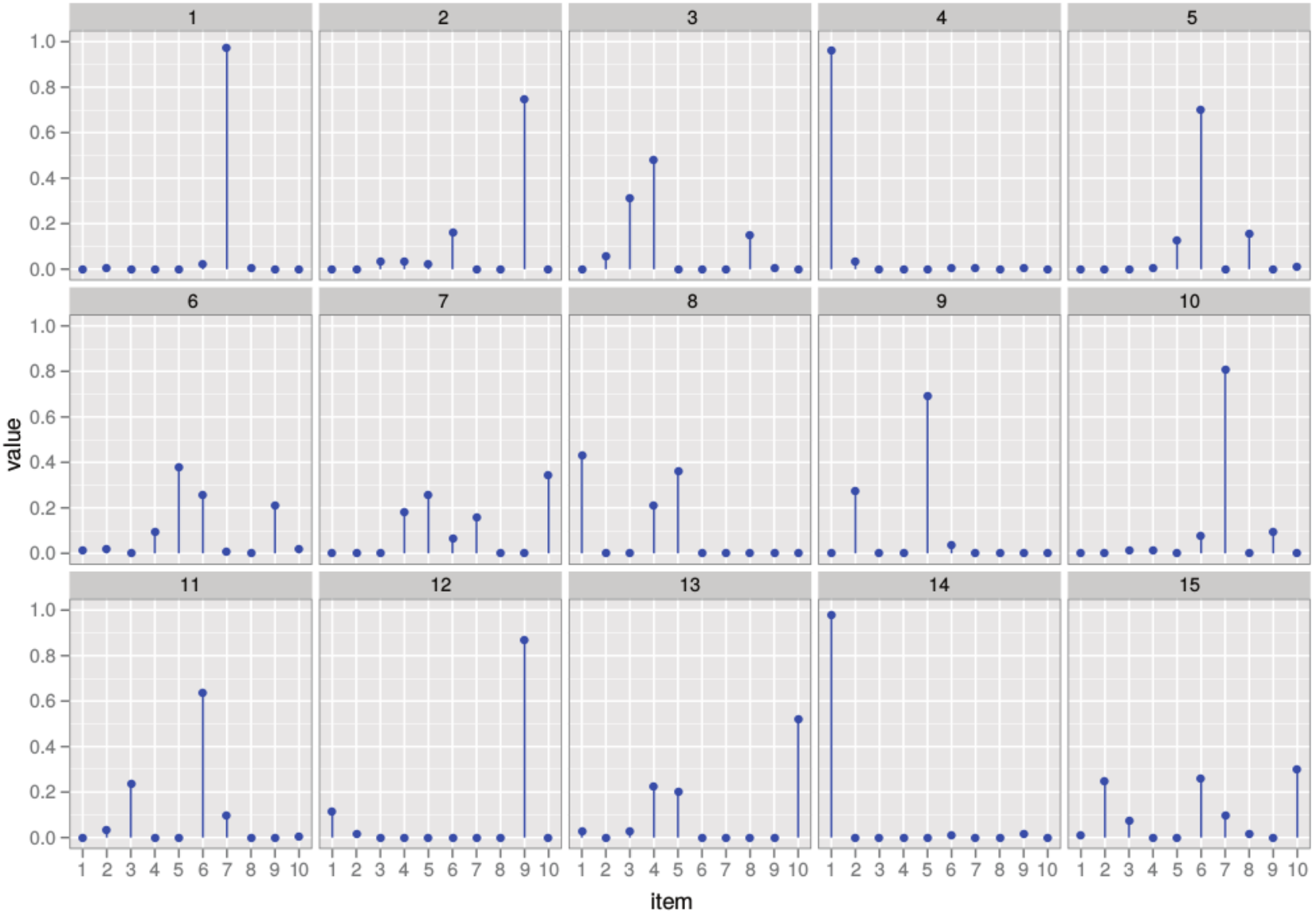
$\alpha = 10$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

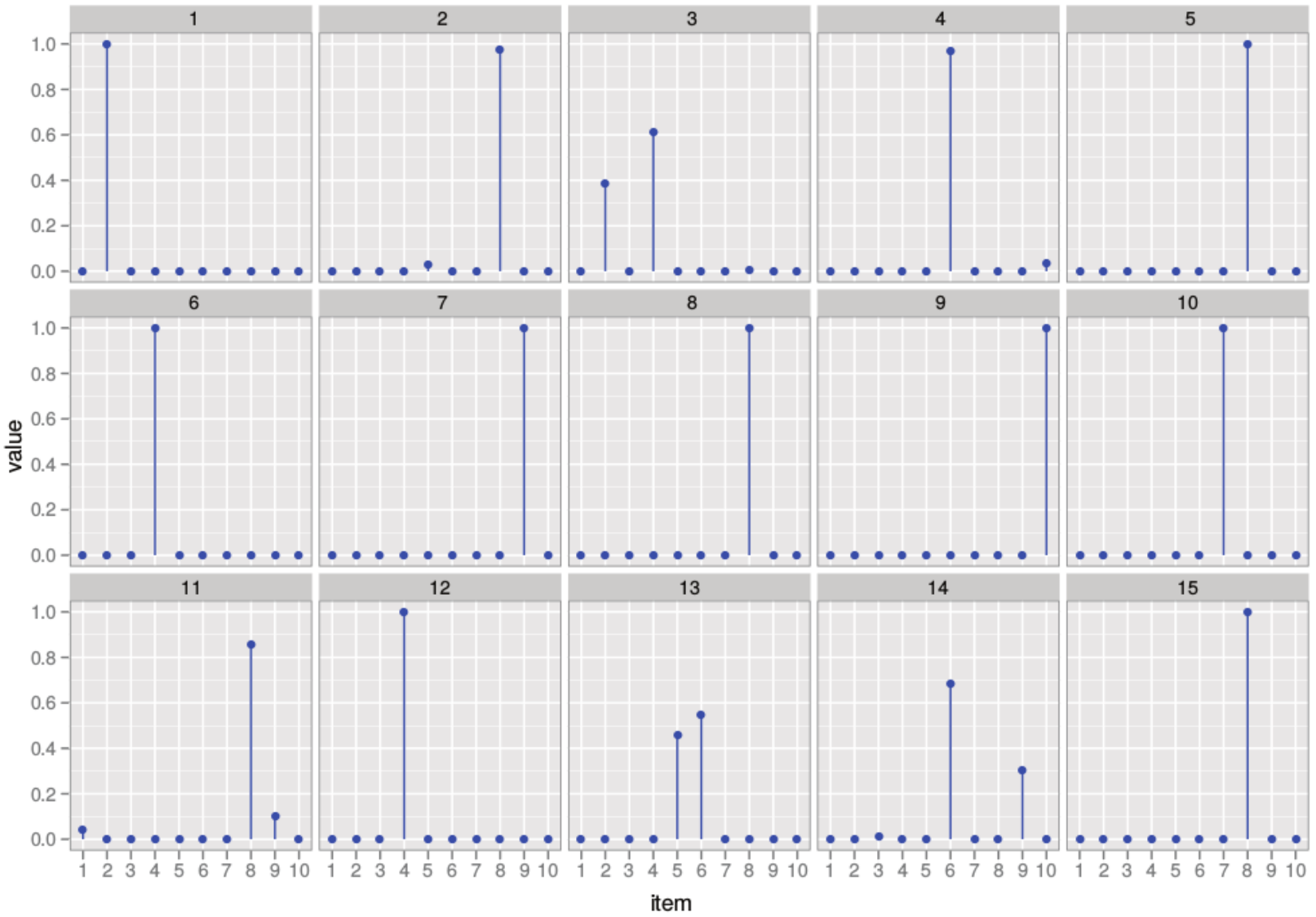
$\alpha = 100$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

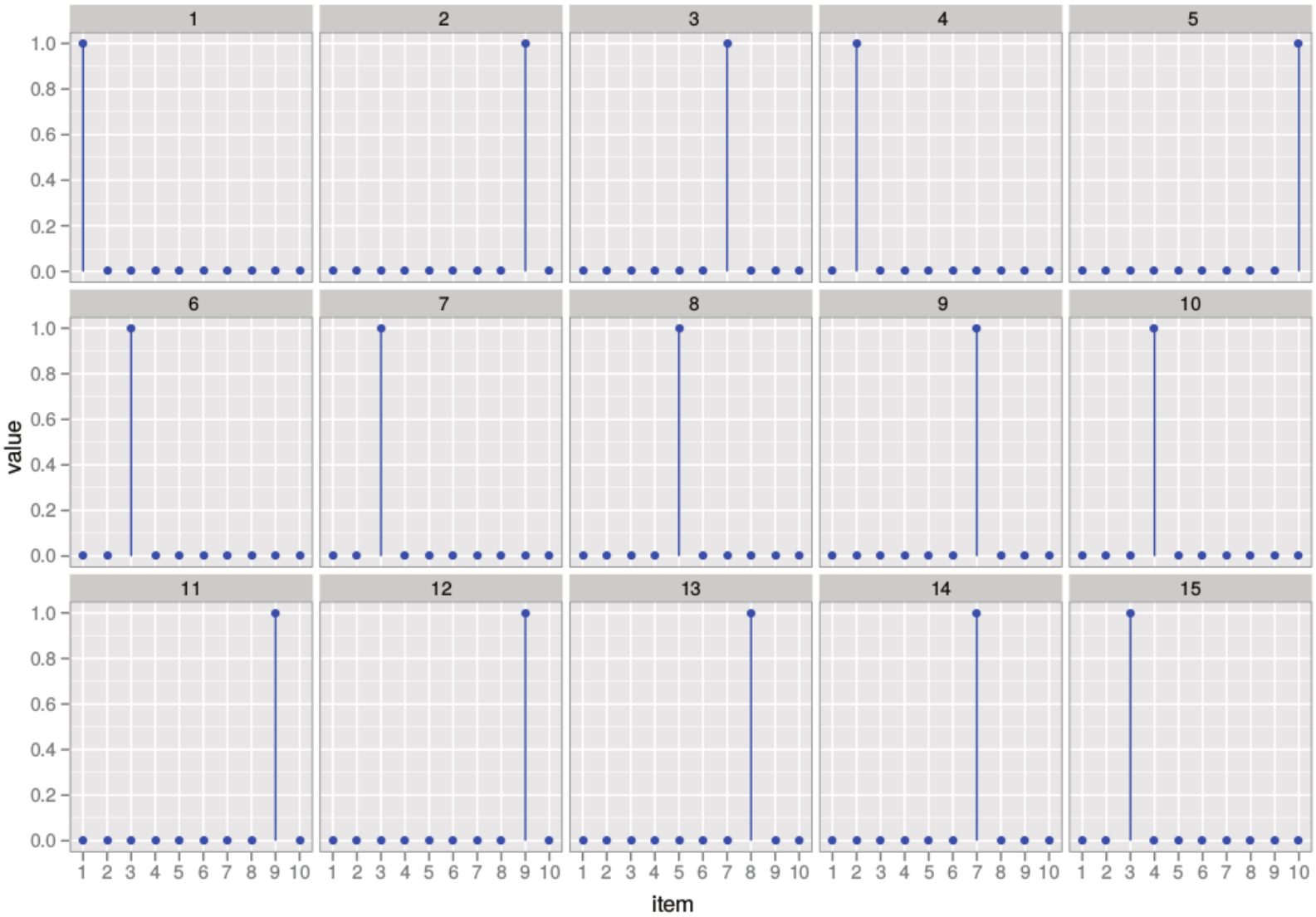
$\alpha = 0.1$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

$\alpha = 0.01$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

$\alpha = 0.001$


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
value
item
0.0
0.2
0.4
0.6
0.8
1.0
1 2 3 4 5 6 7 8 9 10

How to approximate the posterior

Two main families of algorithms:

- **Sampling based algorithms**:
  - Collect samples from the posterior to approximate it with an empirical distribution
- **Variational Methods**:
  - Deterministic alternative to sampling-based algorithms
  - The inference problem is transformed to an optimization problem

Gibbs sampling: MCMC algorithm

- A form of **Markov chain Monte Carlo** (MCMC), which simulates a high-dimensional distribution by sampling on lower-dimensional subset of variables where each subset is conditioned on the value of all others.
- Sampling is done **sequentially** and proceeds until the sampled values approximate the target distribution.
- It directly estimates the posterior distribution over z, and uses this to provide estimates for β and θ.

Idea behind Gibbs sampling

- Suppose $p(x, y)$ is a p.d.f. or p.m.f. that is difficult to sample from directly.
- Suppose, though, that we can easily sample from the conditional distributions $p(x|y)$ and $p(y|x)$.
- The Gibbs sampler proceeds as follows:
  1. set x and y to some initial starting values
  2. then sample $x|y$, then sample $y|x$, then $x|y$, and so on.

Idea behind Gibbs sampling

1. Set (x_0, y_0) to some starting value.
2. Sample $x_1 \sim p(x|y_0)$, that is, from the conditional distribution $X|Y = y_0$.

   Current state: (x_1, y_0)

   Sample $y_1 \sim p(y|x_1)$, that is, from the conditional distribution $Y|X = x_1$.

   Current state: (x_1, y_1)
3. Sample $x_2 \sim p(x|y_1)$, that is, from the conditional distribution $X|Y = y_1$.

   Current state: (x_2, y_1)

   Sample $y_2 \sim p(y|x_2)$, that is, from the conditional distribution $Y|X = x_2$.

   Current state: (x_2, y_2)

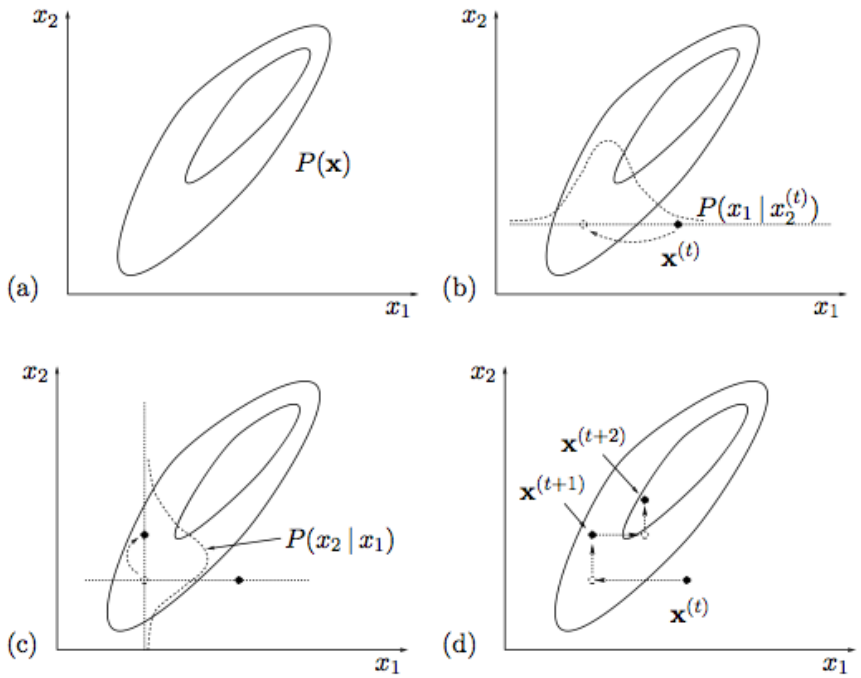
   $\vdots$

Repeat iterations 1 and 2, M times.

Idea behind Gibbs sampling

This procedure defines a sequence of pairs of random variables

$$(X_0, Y_0), (X_1, Y_1), (X_2, Y_2), (X_3, Y_3), \ldots$$



Gibbs sampling

- Suppose we have a word token i for which we want to find the **topic assignment probability**: $p(z_i = j)$
- Represent the collection of documents by a set of **word indices** w_i and **document indices** d_i for this token i
- Gibbs sampling considers each word token in turn and estimates the probability of assigning the current word token to each topic, conditioned on the topic assignment to all other word tokens.
- From this conditional distribution, a **topic is sampled** and stored as the new topic assignment for this word token
- This conditional is written as $P(z_i = j|z_{-i}, w_i, d_i, .)$

Gibbs sampling in LDA

- Collection of documents is a set of word indices w_i and document indices d_i, for each word token i.
- Let us define two matrices C^{WT} and C^{DT} of dimensions $W \times T$ and $D \times T$ respectively.
- C^{WT}_{wj} contains the number of times word w is assigned to topic j, not including the current instance
- C^{WT}_{dj} contains the number of times topic j is assigned to some word token in document d, not including the current instance

Gibbs Sampling in LDA

- For each token i and estimate:

$$P(z_i = j | z_{-i}, w_i, d_i, \cdot) = \frac{C^{WT}_{w_{ij}} + \beta}{\sum_{w=1}^{W} C^{WT}_{w_j} + W\beta} \frac{C^{DT}_{d_{ij}} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_{it}} + T\alpha}$$

- From this conditional distribution, a topic is sampled and stored as the new topic assignment for this word token.

- $z_i = j$: topic assignment of token i to topic j
- z_{-i}: the topic assignments of all other word tokens
- $\cdot$: all other known information (w_{-i} , d_{-i} , α, β).
- C^{WT} : matrix of counts with dimensions $W \times T$
- C^{DT} : matrix of counts with dimensions $D \times T$

Gibbs Sampling in LDA

$$P(z_i = j | z_{-i}, w_i, d_i, \cdot) = \frac{C^{WT}_{w_{ij}} + \beta}{\sum_{w=1}^{W} C^{WT}_{w_j} + W\beta} \frac{C^{DT}_{d_{ij}} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_{it}} + T\alpha}$$

- The left part is the probability of word w under topic j (How likely a word is for a topic) whereas
- the right part is the probability of topic j under the current topic distribution for document d (How dominant a topic is in a document)

LDA Algorithm

- An iterative process.
- Start with random topic assignments for each word.
- In each iteration, for each word in the data:
  - Assume you know (from the prev. iteration) the topics of all other words. (pretend they are correct)
  - Determine the probabilities of each topic-assignment given the rest of the data.
  - Sample topic from the determined probability distribution.
  - Update counts.
- Iterate until converges.

Posterior Estimates of β and θ

$$\beta_{ij} = \frac{C_{w_{ij}}^{WT} + \beta}{\sum_{w=1}^{W} C_{w_j}^{WT} + W\beta}, \theta_{dj} = \frac{C_{d_{ij}}^{DT} + \alpha}{\sum_{t=1}^{T} C_{d_{it}}^{DT} + T\alpha}$$

Using the count matrices as before:

- β_{ij} is the probability of word type i for topic j
- θ_{dj} is the proportion of topic j in document d.

Toy Example

Suppose we artificially generate data from known topic model.

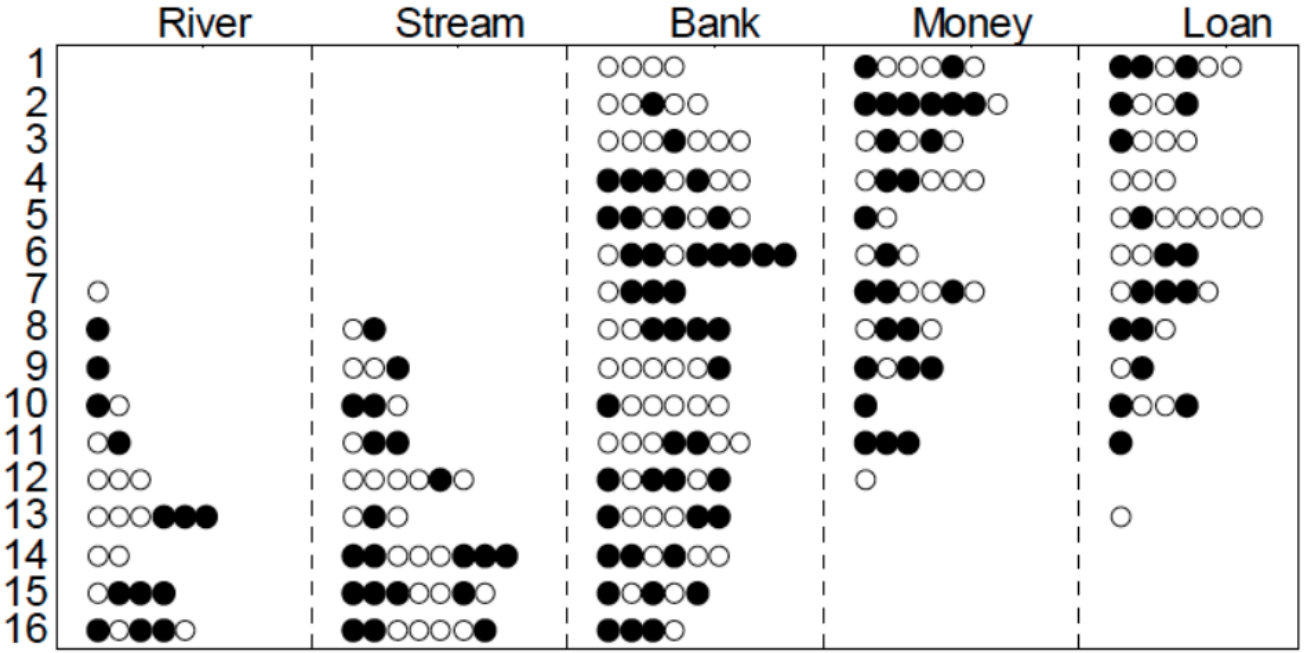
- Let topic 1 give equal probability to MONEY, LOAN, BANK and topic 2 give equal probability to words RIVER, STREAM, and BANK

$$\beta^{(1)}_{MONEY} = \beta^{(1)}_{LOAN} = \beta^{(1)}_{BANK} = 1/3$$

$$\beta^{(2)}_{RIVER} = \beta^{(2)}_{STREAM} = \beta^{(2)}_{BANK} = 1/3$$

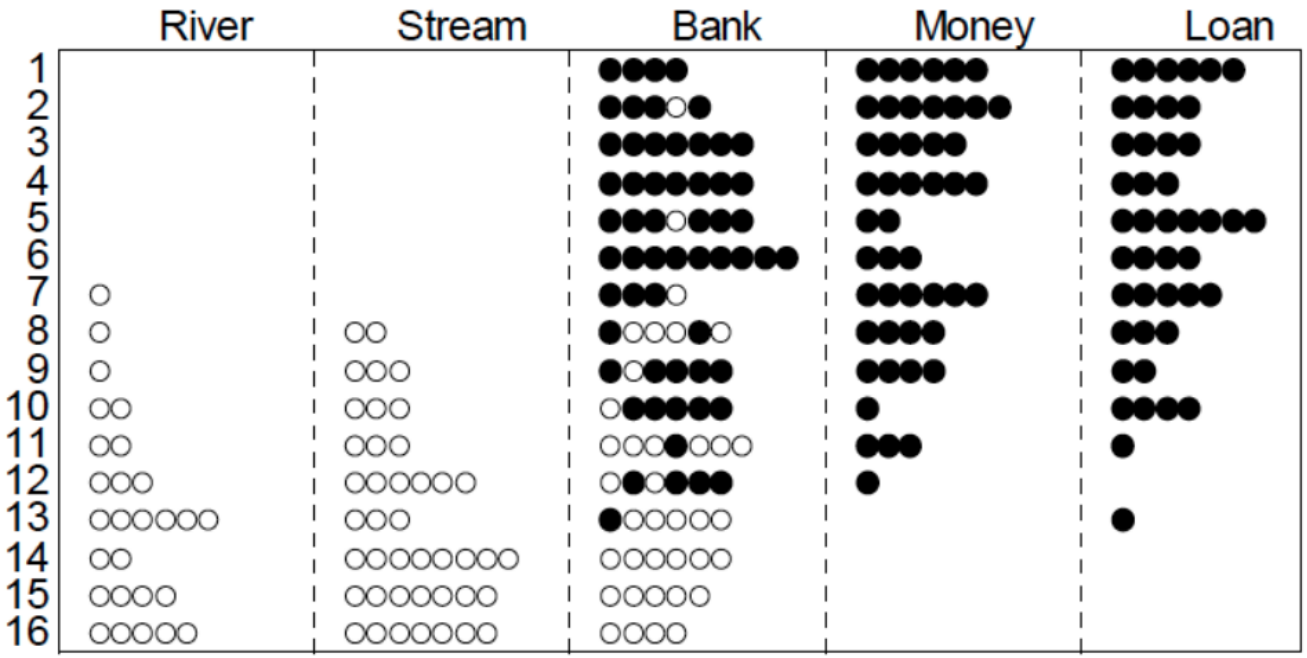
- Generate 16 documents mixing topics.

Toy Example: Initial state



- black = topic 1
- white = topic 2

Toy Example: After some iterations



$$\beta^{(1)}_{MONEY} = 0.32, \beta^{(1)}_{LOAN} = 0.29, \beta^{(1)}_{BANK} = 0.39$$

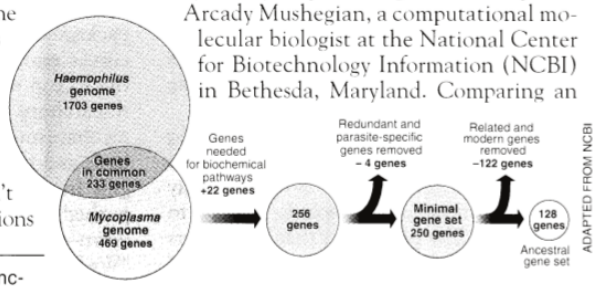
$$\beta^{(2)}_{RIVER} = 0.25, \beta^{(2)}_{STREAM} = 0.4, \beta^{(2)}_{BANK} = 0.35$$

Example Inference

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.

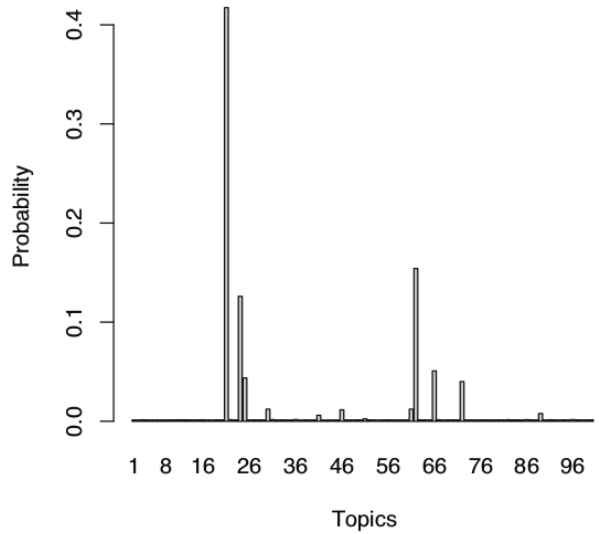
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just 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



Example Inference

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

Example Inference

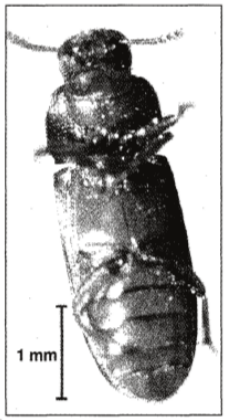
Chaotic Beetles

Charles Godfray and Michael Hassell

Ecologists have known since the pioneering work of May in the mid-1970s (*1*) that the population dynamics of animals and plants can be exceedingly complex. This complexity arises from two sources: The tangled web of interactions that constitute any natural community provide a myriad of different pathways for species to interact, both directly and indirectly. And even in isolated populations the nonlinear feedback processes present in all natural populations can result in complex dynamic behavior. Natural populations can show persistent oscillatory dynamics and chaos, the latter characterized by extreme sensitivity to initial conditions. If such chaotic dynamics were common in nature, then this would have important ramifications for the management and conservation of natural resources. On page 389 of this issue, Costantino *et al.* (*2*) provide the most convincing evidence to date of complex dynamics and chaos in a biological population—of the flour beetle, *Tribolium castaneum* (see figure).

It has proven extremely difficult to demonstrate complex dynamics in populations in the field. By its very nature, a chaotically fluctuating population will superficially resemble a stable or cyclic population buffeted by the normal random perturbations experienced by all species. Given a long enough time series, diagnostic tools from nonlinear mathematics can be used to identify the telltale signatures of chaos. In phase space, chaotic trajectories come to lie on "strange attractors," curious geometric objects with fractal structure and hence noninteger dimension. As they move over the surface of the attractor, sets of adjacent trajectories are pulled apart, then stretched and folded, so that it becomes impossible to predict exact population densities into the future. The strength of the mixing that gives rise to the extreme sensitivity to initial conditions can be measured mathematically estimating the Liapunov exponent, which is positive for chaotic dynamics and nonpositive otherwise. There have been many attempts to estimate attractor dimension and Liapunov exponents from time series data, and some candidate chaotic population have been identified (some insects, rodents, and most convincingly, human childhood diseases), but the statistical difficulties preclude any broad generalization (*3*).

An alternative approach is to parameterize population models with data from natural populations and then compare their predictions with the dynamics in the field. This technique has been gaining popularity in recent years, helped by statistical advances in parameter estimation. Good ex-



Cannibalism and chaos. The flour beetle, *Tribolium castaneum*, exhibits chaotic population dynamics when the amount of cannibalism is altered in a mathematical model.

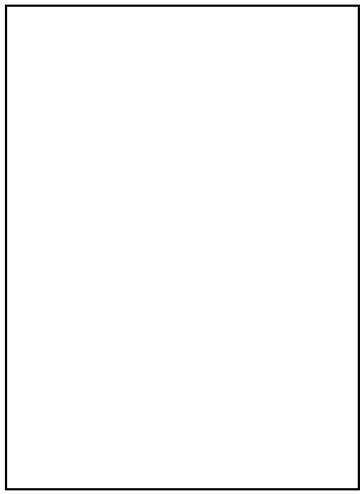
The authors are in the Department of Biology, Imperial College at Silwood Park, Ascot, Berks, SL5 7PZ UK. E-mail: m.hassell@ic.ac.uk

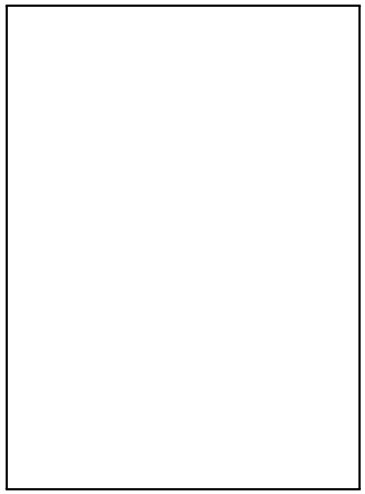
Example Inference

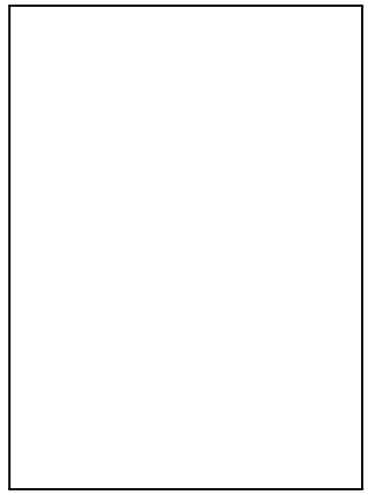
problem	model	selection	species
problems	rate	male	forest
mathematical	constant	males	ecology
number	distribution	females	fish
new	time	sex	ecological
mathematics	number	species	conservation
university	size	female	diversity
two	values	evolution	population
first	value	populations	natural
numbers	average	population	ecosystems
work	rates	sexual	populations
time	data	behavior	endangered
mathematicians	density	evolutionary	tropical
chaos	measured	genetic	forests
chaotic	models	reproductive	ecosystem

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, free association, ambiguity, semantic priming, reading time, free recall.

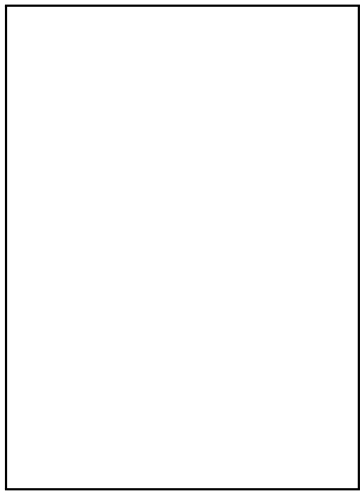
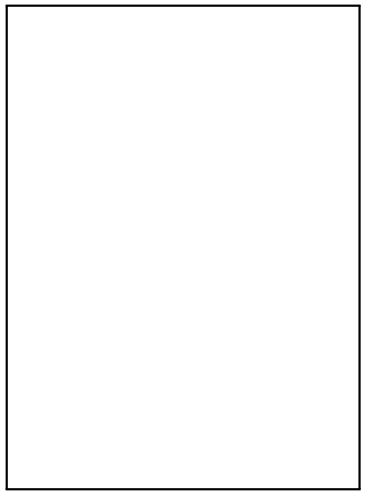
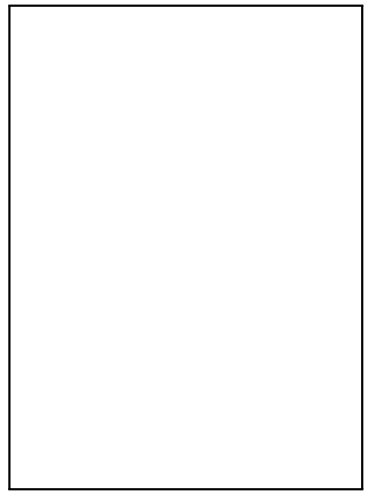






Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

SPOTS

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

SPOTS
DOG
DIRTY
DIRT
STRIPES
DARK

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

SPOTS
DOG
DIRTY
DIRT
STRIPES
DARK

RICE

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

SPOTS
DOG
DIRTY
DIRT
STRIPES
DARK

RICE
CHINESE
WEDDING
FOOD
WHITE
CHINA

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

SPOTS
DOG
DIRTY
DIRT
STRIPES
DARK

RICE
CHINESE
WEDDING
FOOD
WHITE
CHINA

SEPARATE

Model Evaluation

- Griffiths et al. (2007) present a range of evaluations and modeling tasks comparing LDA and LSA.
- Synonym tests from TOEFL, **free association**, ambiguity, semantic priming, reading time, free recall.
- **Free-association**: produce the first word that comes to mind in response to a cue word.

SPOTS
DOG
DIRTY
DIRT
STRIPES
DARK

RICE
CHINESE
WEDDING
FOOD
WHITE
CHINA

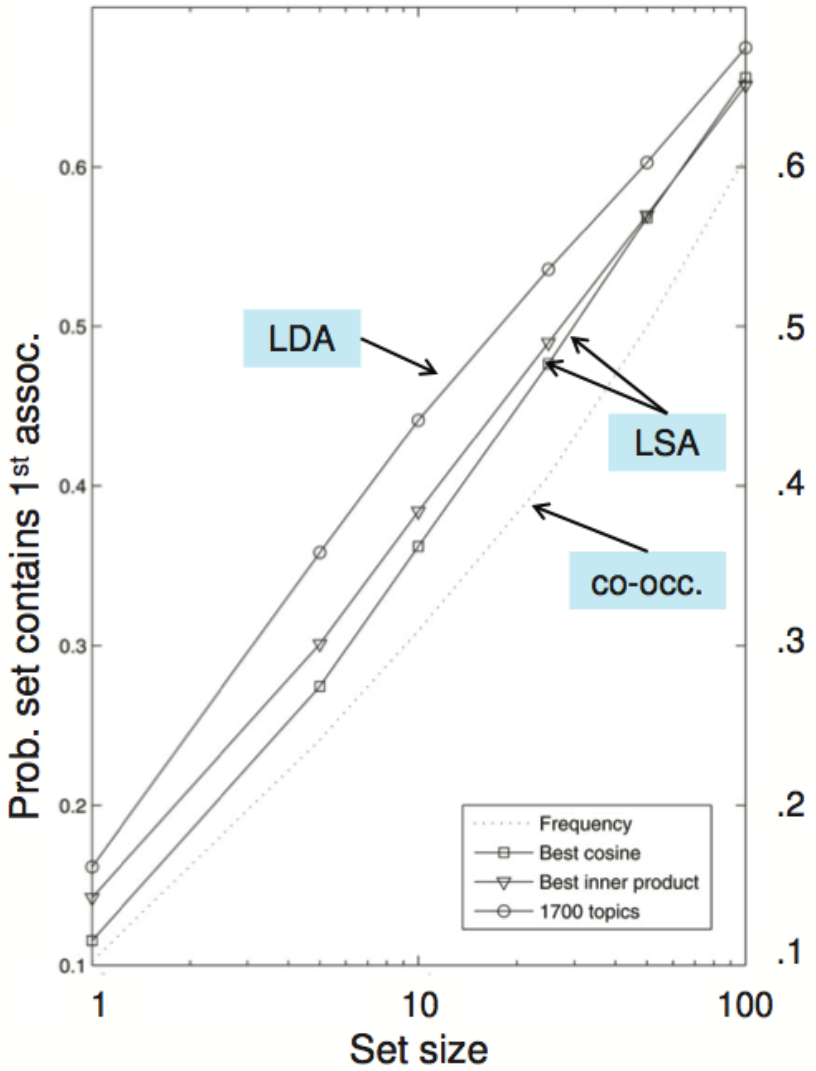
SEPARATE
DIVIDE
DIVORCE
PART
SPLIT
REMOVE

Computing Word Similarity

- Build LSA and LDA representations from corpus of educational materials.
- Compute associates predicted by each model.
- LSA: top associate of w_1 is word with closest cosine (or dot product) similarity.
- LDA: top associate of w_1 is word with highest $P(w_2|w_1)$.

$$P(w_2|w_1) = \sum_z P(w_2|z)P(z|w_1)$$

Results



If we take the top n associates returned by the model (set size), what is the probability that this set contains the true first associate?

Discussion

. . . but we can do more:

- Use LDA as a basic building block when building more complex models
- by now, we only include one kind of stucture into the model and call it "topics"
- by using the Dirichlet distribution, we assume that topics are more or less independent of each other

What can we change?

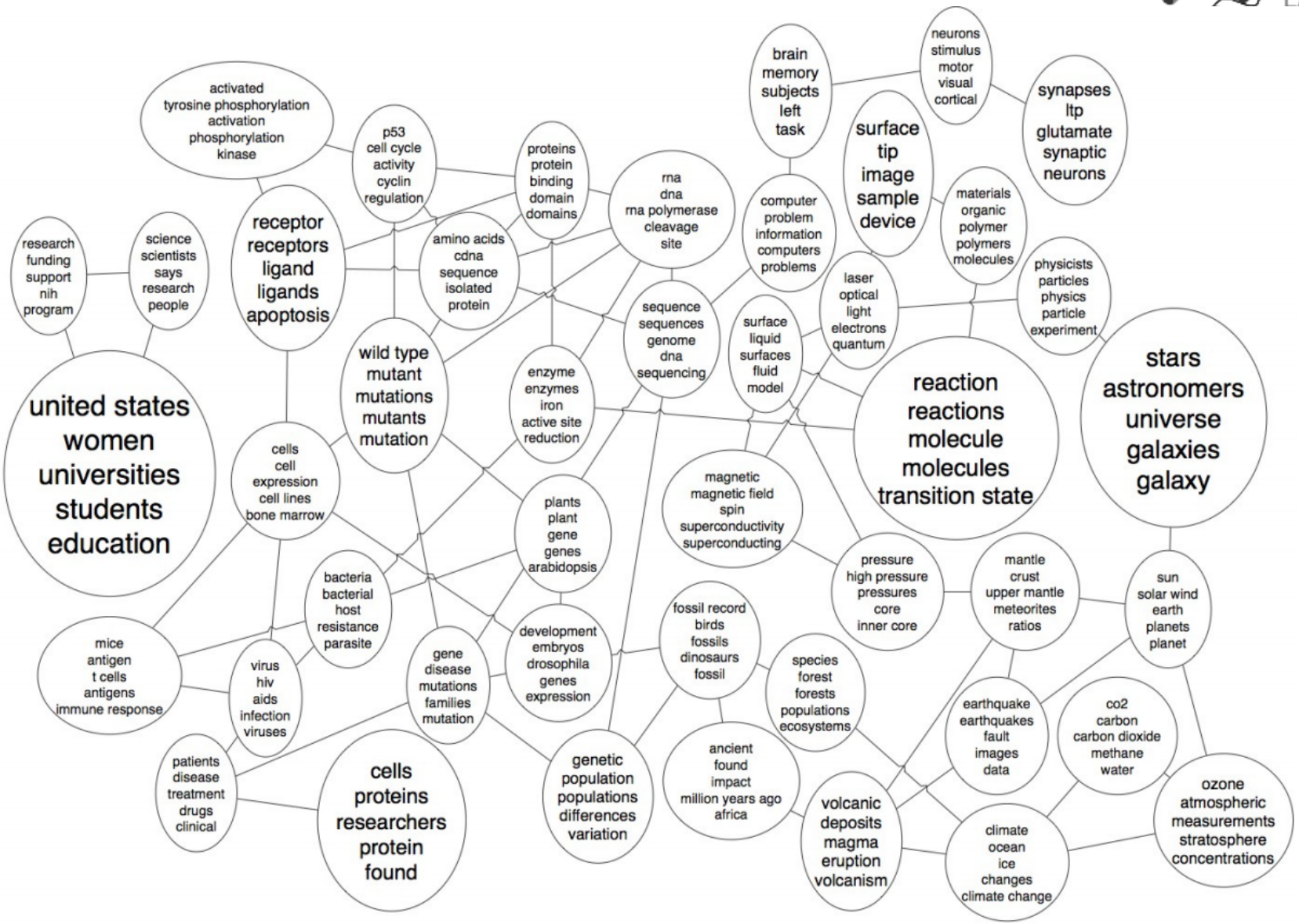
- We can change/add structure in the model.
  - Model represents our assumptions about the data.
- Or we can change prior distributions.

New models

Correlated Topic Models:

- Dirichlet prior assumes independence in the cooccurrence of topics in documents.
- e.g. if an document is about climate change, it is more likely that topics like politics and economy coocur rather than sports
- Change prior distribution: Logistic Normal can handle those interdependencies.

CTM output


activated tyrosine phosphorylation activation phosphorylation kinase
p53 cell cycle activity cyclin regulation
proteins protein binding domain domains
rna dna rna polymerase cleavage site
brain memory subjects left task
neurons stimulus motor visual cortical
synapses ltp glutamate synaptic neurons
surface tip image sample device
materials organic polymer polymers molecules
computer problem information computers problems
research funding support nih program
science scientists says research people
receptor receptors ligand ligands apoptosis
amino acids cdna sequence isolated protein
sequence sequences genome dna sequencing
surface liquid surfaces fluid model
laser optical light electrons quantum
physicists particles physics particle experiment
wild type mutant mutations mutants mutation
enzyme enzymes iron active site reduction
reaction reactions molecule molecules transition state
stars astronomers universe galaxies galaxy
united states women universities students education
cells cell expression cell lines bone marrow
magnetic magnetic field spin superconductivity superconducting
plants plant gene genes arabidopsis
pressure high pressure pressures core inner core
mantle crust upper mantle meteorites ratios
sun solar wind earth planets planet
bacteria bacterial host resistance parasite
fossil record birds fossils dinosaurs fossil
development embryos drosophila genes expression
mice antigen t cells antigens immune response
virus hiv aids infection viruses
gene disease mutations families mutation
species forest forests populations ecosystems
earthquake earthquakes fault images data
co2 carbon carbon dioxide methane water
patients disease treatment drugs clinical
cells proteins researchers protein found
genetic population populations differences variation
ancient found impact million years ago africa
volcanic deposits magma eruption volcanism
climate ocean ice changes climate change
ozone atmospheric measurements stratosphere concentrations

New models

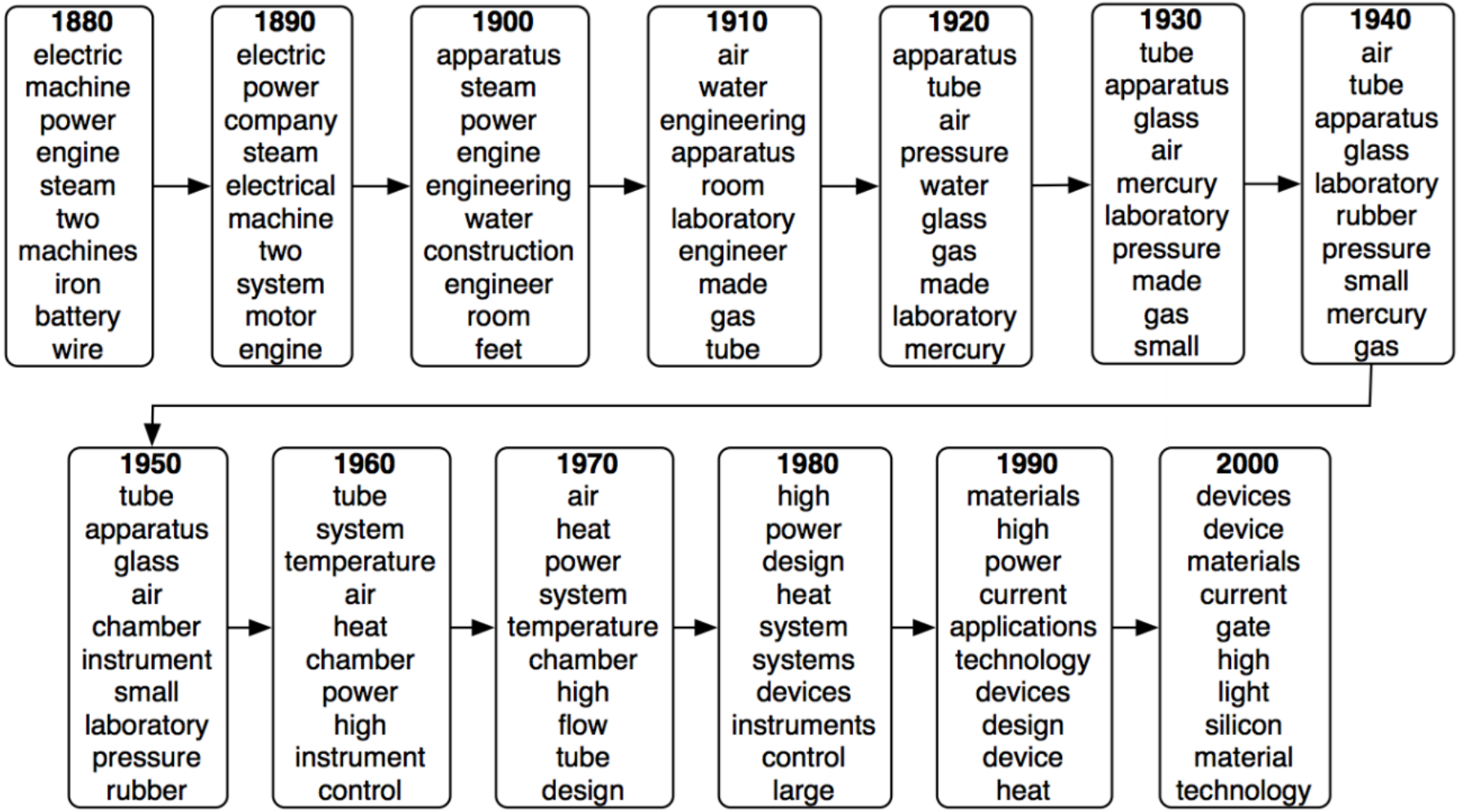
Dynamic Topic Models

LDA assumes exchangeability for both documents and words (no matter if a documents is computed before or later)

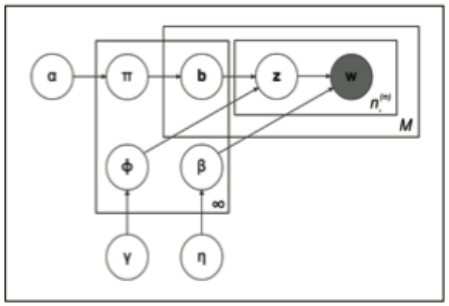
- Working with large corpora spanning over some decades or even centuries,
- we can track language, its uses, and changes over time
- Dynamic topic models allow a topic drift in a sequence

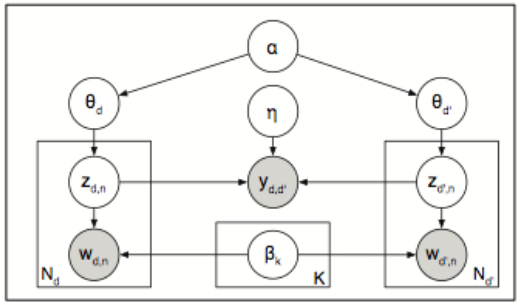
- Simply take several LDA models, one for each time slice
- and use a logistic normal to model the topics evolving over time

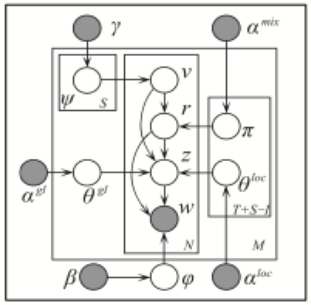
DTM output

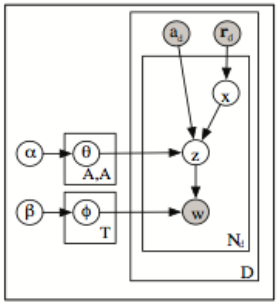

1880
electric
machine
power
engine
steam
two
machines
iron
battery
wire
1890
electric
power
company
steam
electrical
machine
two
system
motor
engine
1900
apparatus
steam
power
engine
engineering
water
construction
engineer
room
feet
1910
air
water
engineering
apparatus
room
laboratory
engineer
made
gas
tube
1920
apparatus
tube
air
pressure
water
glass
gas
made
laboratory
mercury
1930
tube
apparatus
glass
air
mercury
laboratory
pressure
made
gas
small
1940
air
tube
apparatus
glass
laboratory
rubber
pressure
small
mercury
gas
1950
tube
apparatus
glass
air
chamber
instrument
small
laboratory
pressure
rubber
1960
tube
system
temperature
air
heat
chamber
power
high
instrument
control
1970
air
heat
power
system
temperature
chamber
high
flow
tube
design
1980
high
power
design
heat
system
systems
devices
instruments
control
large
1990
materials
high
power
current
applications
technology
devices
design
device
heat
2000
devices
device
materials
current
gate
high
light
silicon
material
technology

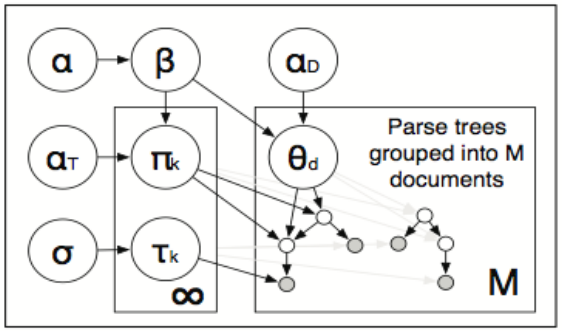
Discussion

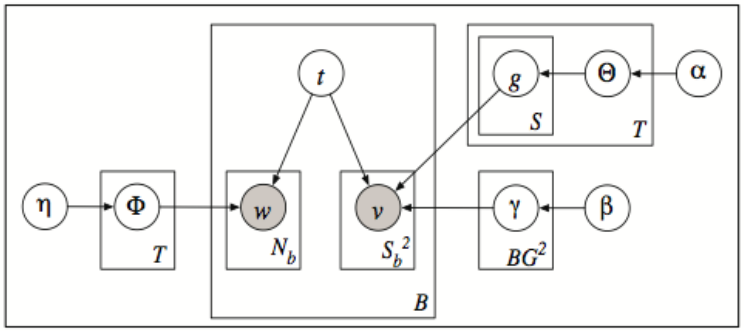


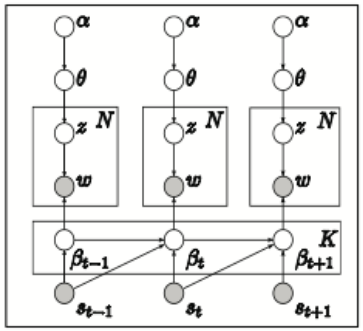












References

Griffiths, Thomas L., Mark Steyvers, and Joshua B. Tenenbaum. 2007. Topics in semantic representation. Psychological Review 114(2):211–244.