Lecture 3: Topic Models

Mirella Lapata

School of Informatics
University of Edinburgh
mlap@inf.ed.ac.uk

January 24, 2016
Slides based on Blei 2011

Reading: Griffiths et al. (2007).

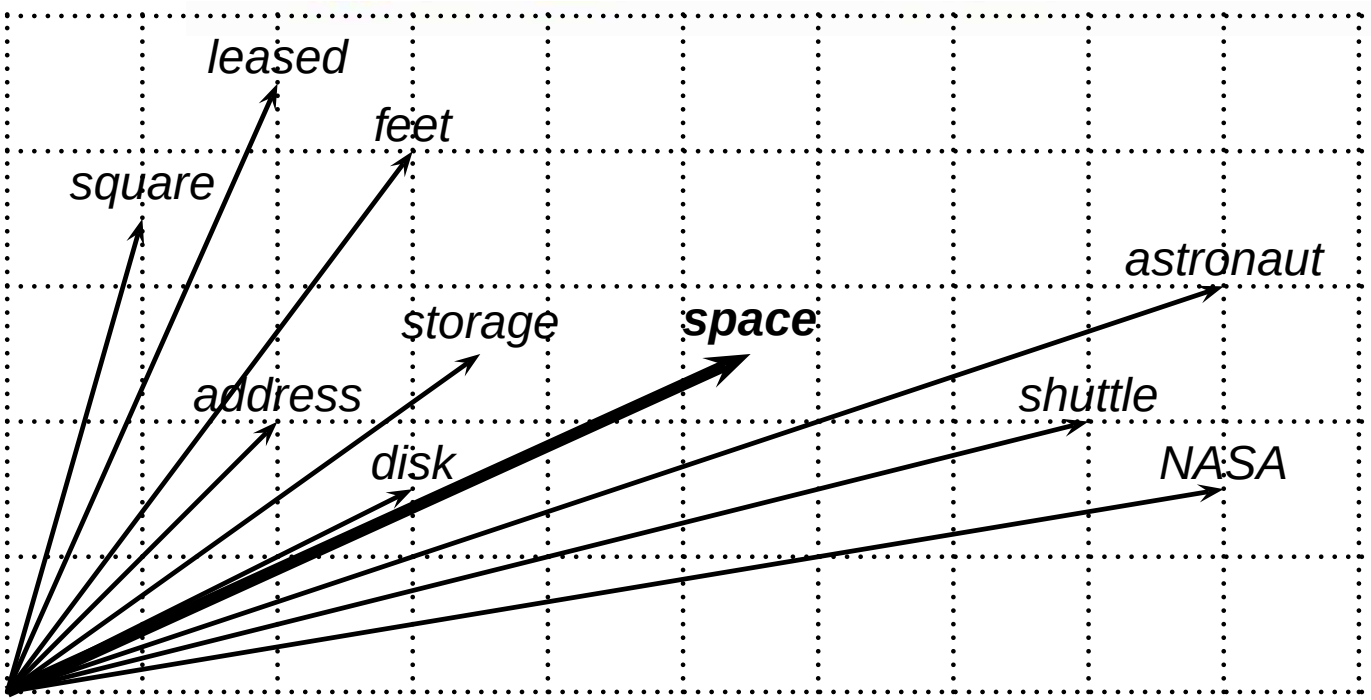
Motivation

We'd like to represent the meaning of words and documents, to help us do things like:

- Retrieve documents relevant to a query.
- Extract features for downstream applications (e.g., sentiment classification, entailment recognition, MT)
- Explain human learning and processing of words (word associations, speed of acquisition, etc).
- Analyze large document collections.

Vector-based Models

Represent the meaning of words in high-dimensional space:



Multiple word meanings are conflated, no probability model.

Topic Models

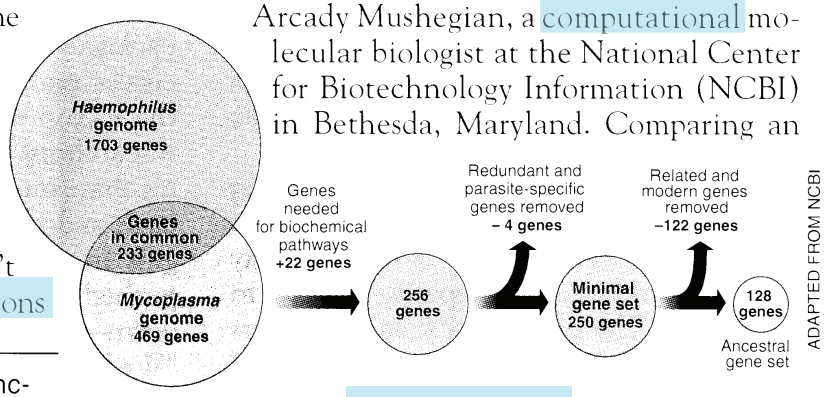
- We want to find themes (or topics) in documents which is useful for e.g., search or browsing.
- We don't want to do supervised topic classification (neither fix topics in advance nor do manual annotation).
- Approach must automatically tease out the topics.
- Essentially a clustering problem: both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

Topic Models

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



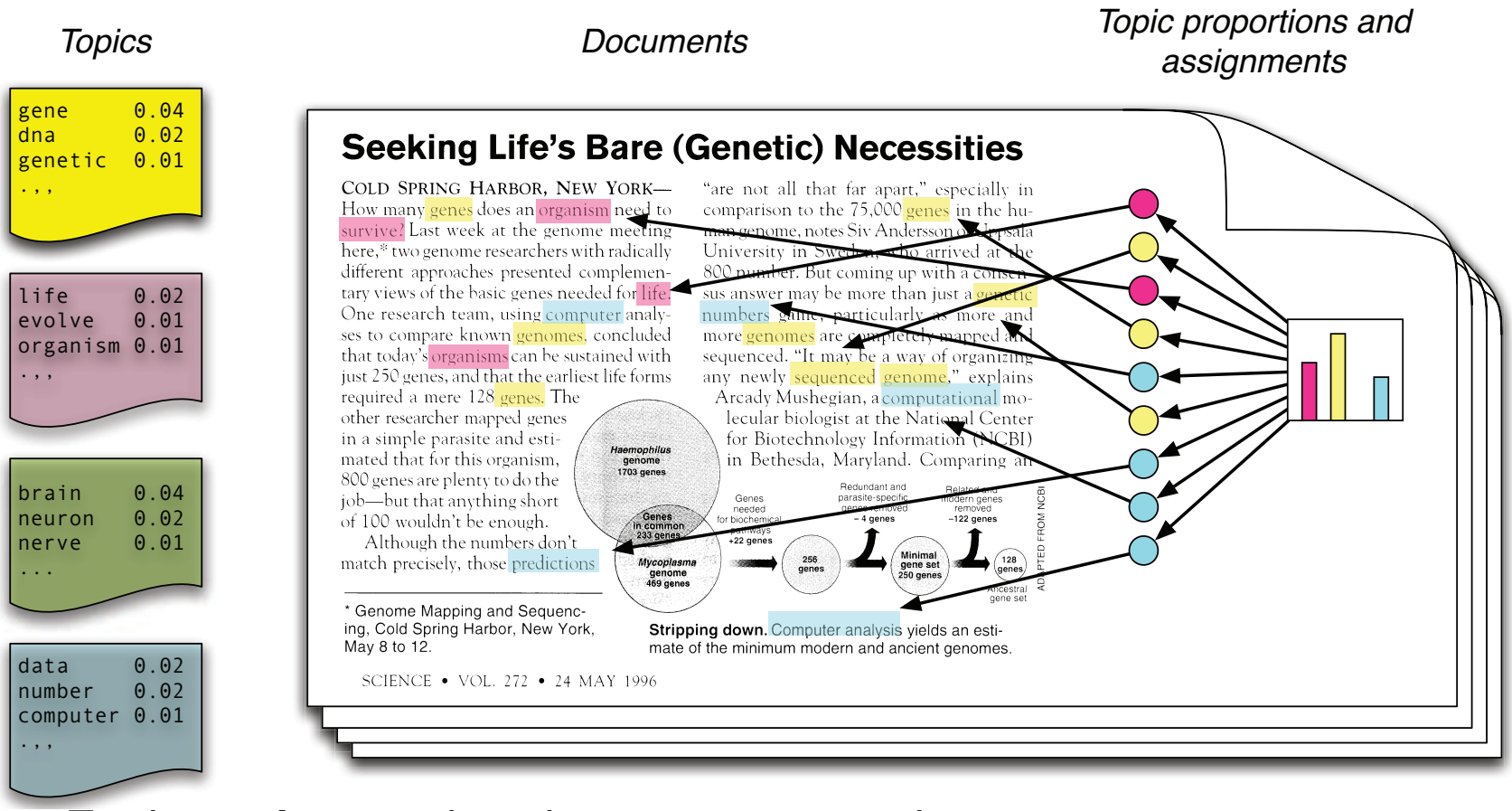
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

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

SCIENCE • VOL. 272 • 24 MAY 1996

Simple intuition: documents exhibit multiple topics.

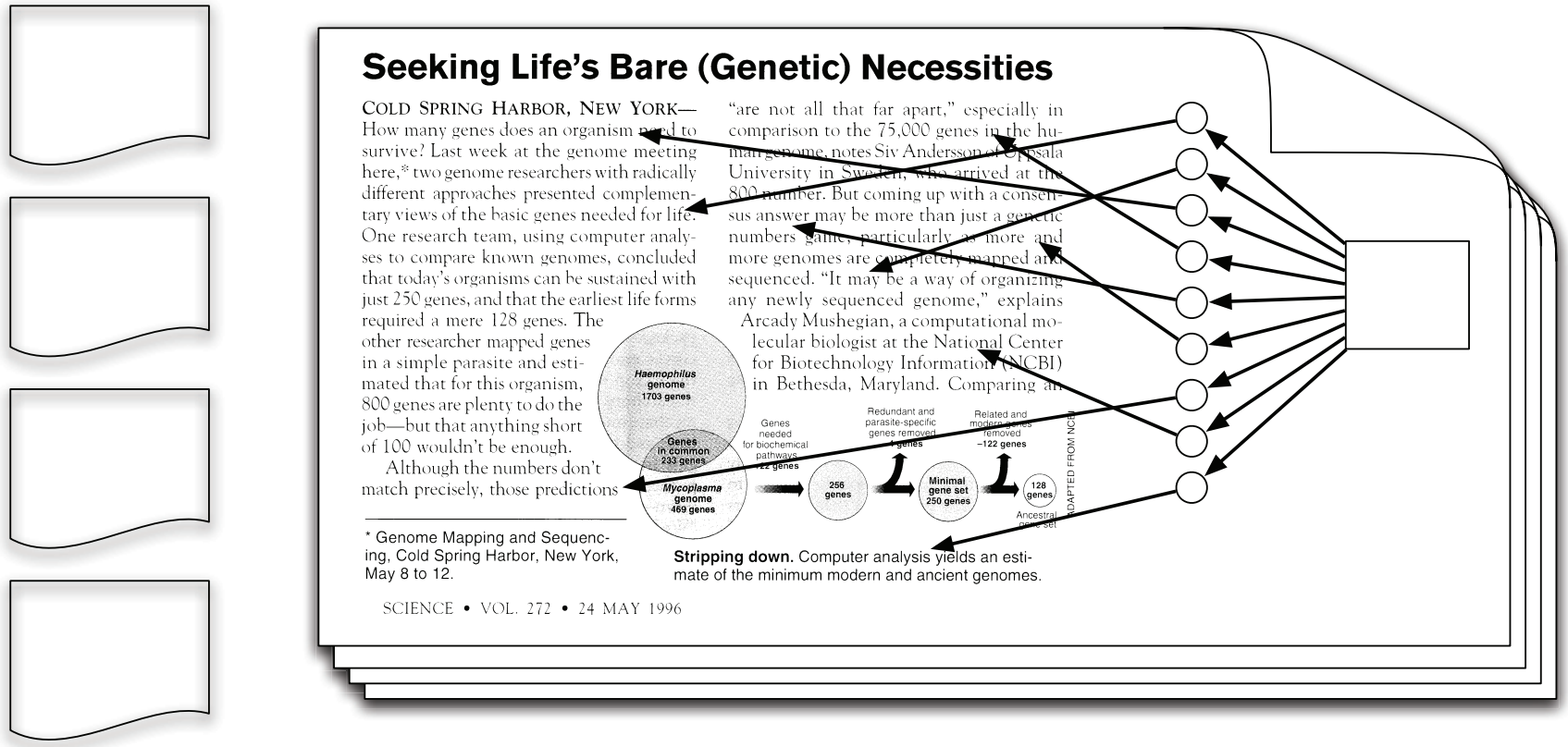
Latent Dirichlet Allocation



- Each **topic** is a distribution over words
- Each **document** is a mixture of corpus-wide topics
- Each **word** is drawn from one of those topics

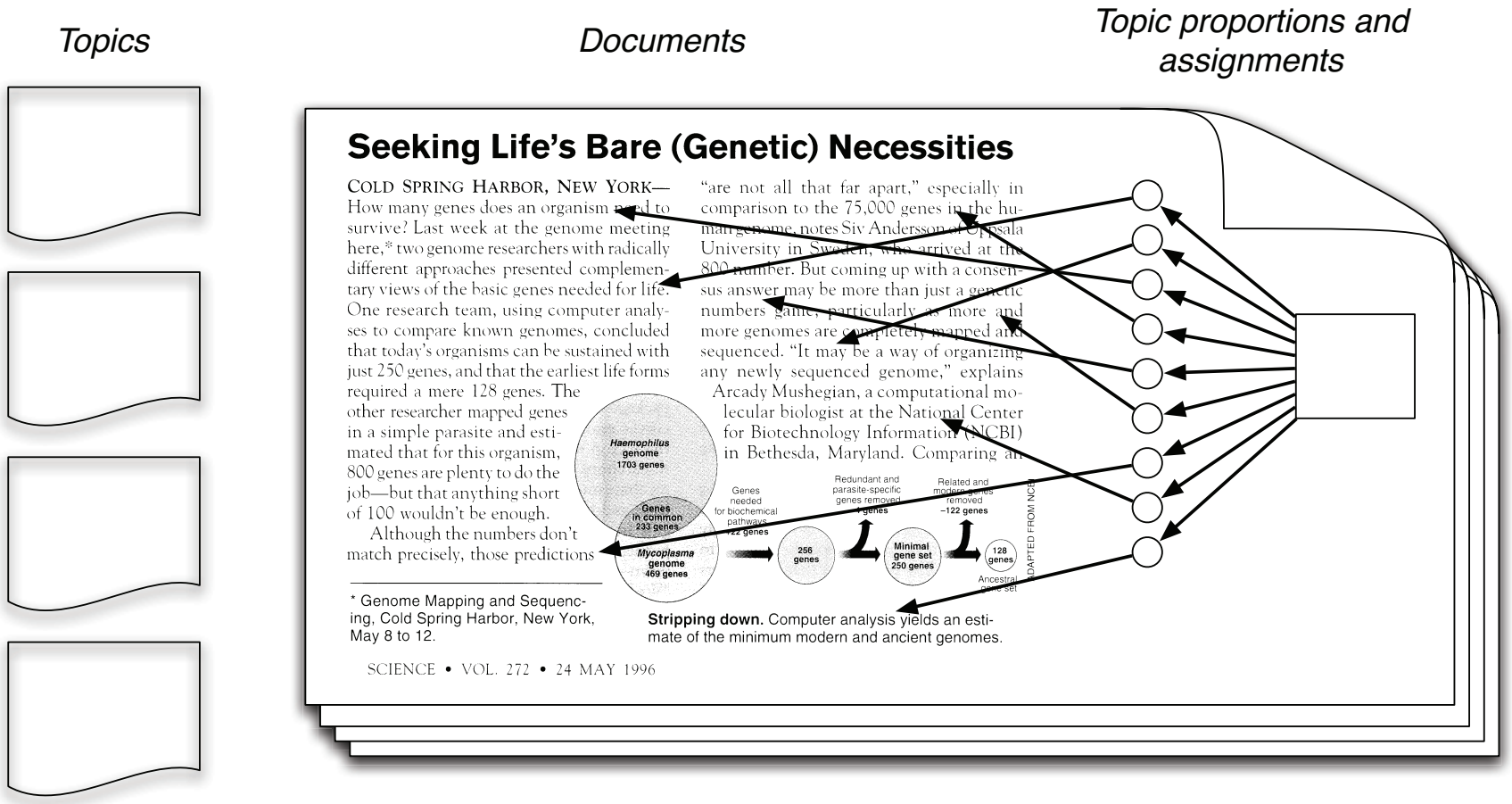
The Posterior Distribution

Topics *Documents* *Topic proportions and assignments*



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

The Posterior Distribution



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

$$p(\text{topics, proportions, assignments}|\text{documents})$$

LDA: Key Assumptions

- 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

Document Generation

1. Choose a number of words N the document will have
2. Choose topic mixture for document according to Dirichlet distribution over a set of T topics

   Dirichlet distribution $\mathrm{Dir}(\alpha)$; α is prior: what kind of topic mixtures can I generally expect (determines shape of distribution)

3. Generate each word w_i in the document by:
   1. randomly choosing a topic from the distribution over topics
   2. randomly choose a word from the corresponding topic (distribution over the vocabulary)

      Dirichlet distribution $\mathrm{Dir}(\beta)$

⚠ Note we need a distribution over a distribution (for step 1)

⚠ Note that words are generated independently of other words (unigram bag-of-words model)

Document Generation: Example

1. Pick 5 to be the number of words in D.
2. Decide D will be 1/2 about food and 1/2 about cute animals.
3. Pick first word to come from the food topic: *broccoli*.
4. Pick second word to come from cute animals topic: *panda*.
5. Pick third word from the cute animals topic: *adorable*.
6. Pick fourth word to from the food topic: *cherries*.
7. Pick fifth word to come from food topic: *eating*.

What is the document generated under the LDA model?

Learning

1. Go through each document, and randomly assign each word in the document to one of T topics.
2. For each document d, go through each word w in d and for each topic t compute:
    1. $p(topic\ t|document\ d)$
    2. $p(word\ w|topic\ t)$
3. Reassign w a new topic, where we choose topic t with probability: $p(topic\ t|document\ d) * p(word\ w|topic\ t)$
4. After repeating the previous step a large number of times, youll eventually reach a roughly steady state where your assignments are pretty good.

Generative Process

For $j = 1 \ldots T$ topics,

Choose $\varphi^{(j)} \sim \text{Dirichlet}(\beta)$.

$\varphi_1^{(j)} \ldots \varphi_V^{(j)}$: prob. of each wd. in topic j

For $d = 1 \ldots D$ documents,

Choose $\theta^{(d)} \sim \text{Dirichlet}(\alpha)$.

$\theta_1^{(d)} \ldots \theta_T^{(d)}$: prob. of each topic in doc. d

For $i = 1 \ldots N_d$ words in doc d,

Choose $z_i \sim \text{Multinomial}(\theta^{(d)})$.

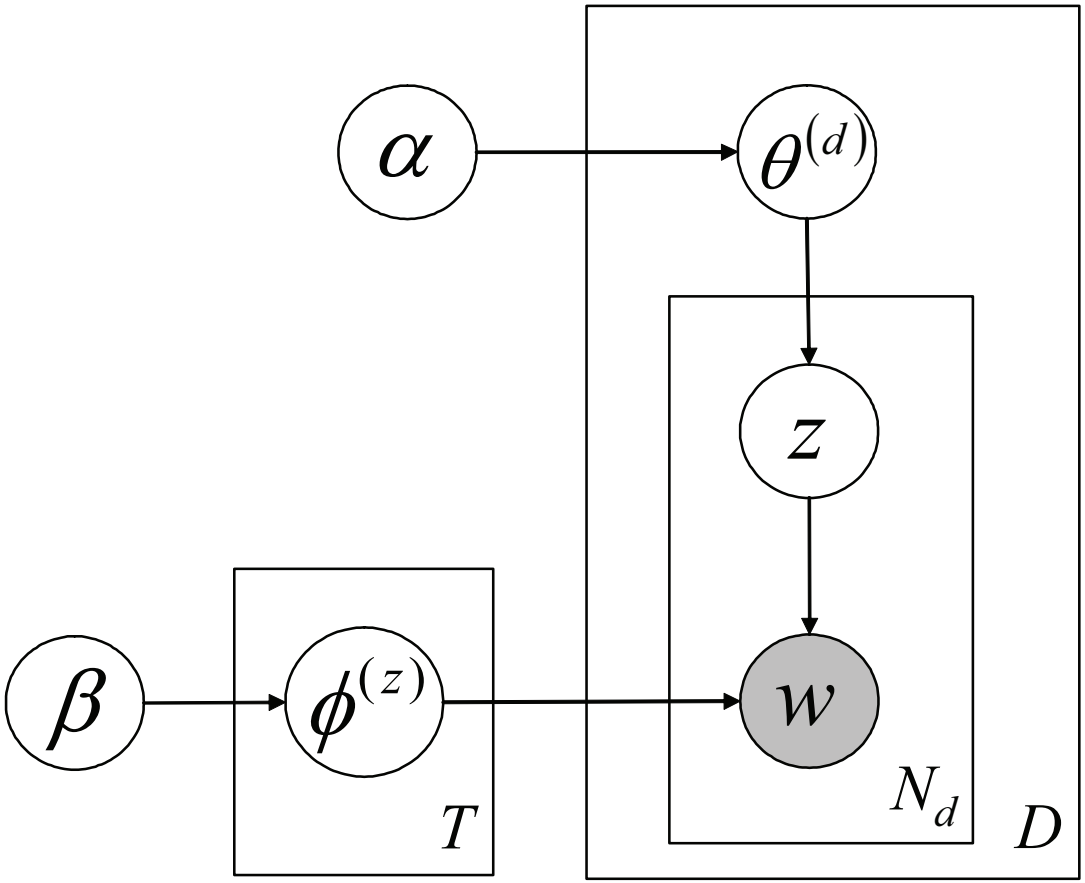
z_i : topic of word i

Choose $w_i \sim \text{Multinomial}(\varphi^{(zi)})$.

w_i : identity of word i

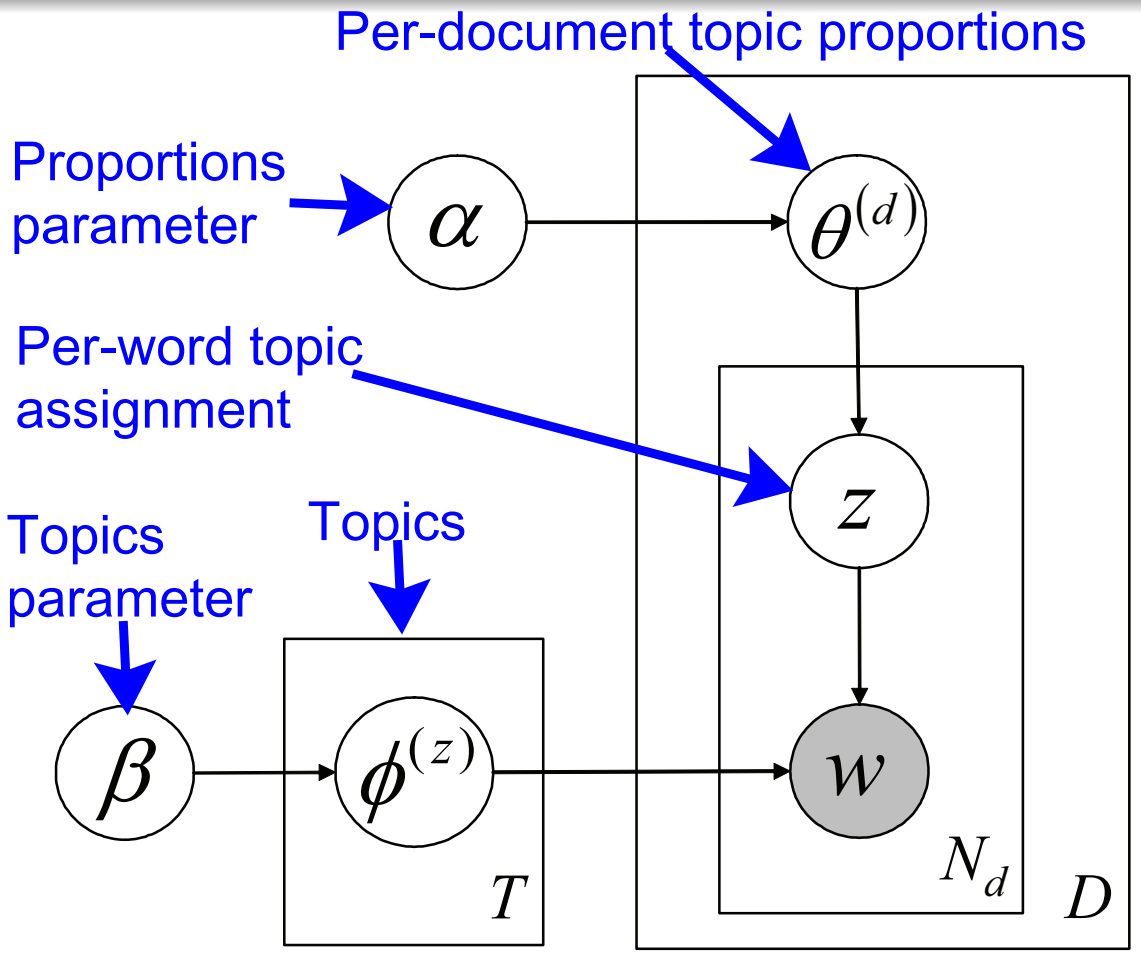
Note that, except in the appendices, Griffiths et al. (2007) use g (for gist) to refer to θ. They also use the term Discrete rather than Multinomial.

The Graphical Model



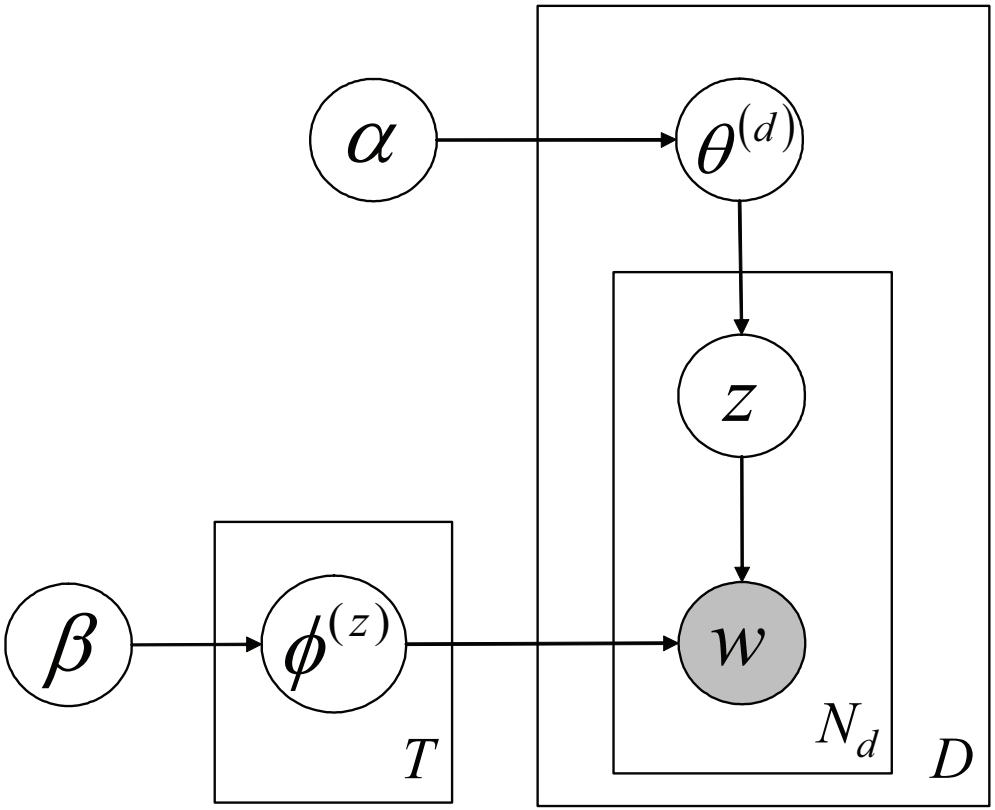
- Nodes are random variables; edges indicate dependence.
- Shaded nodes are observed; plates $\approx$ replicated variables.

The Graphical Model

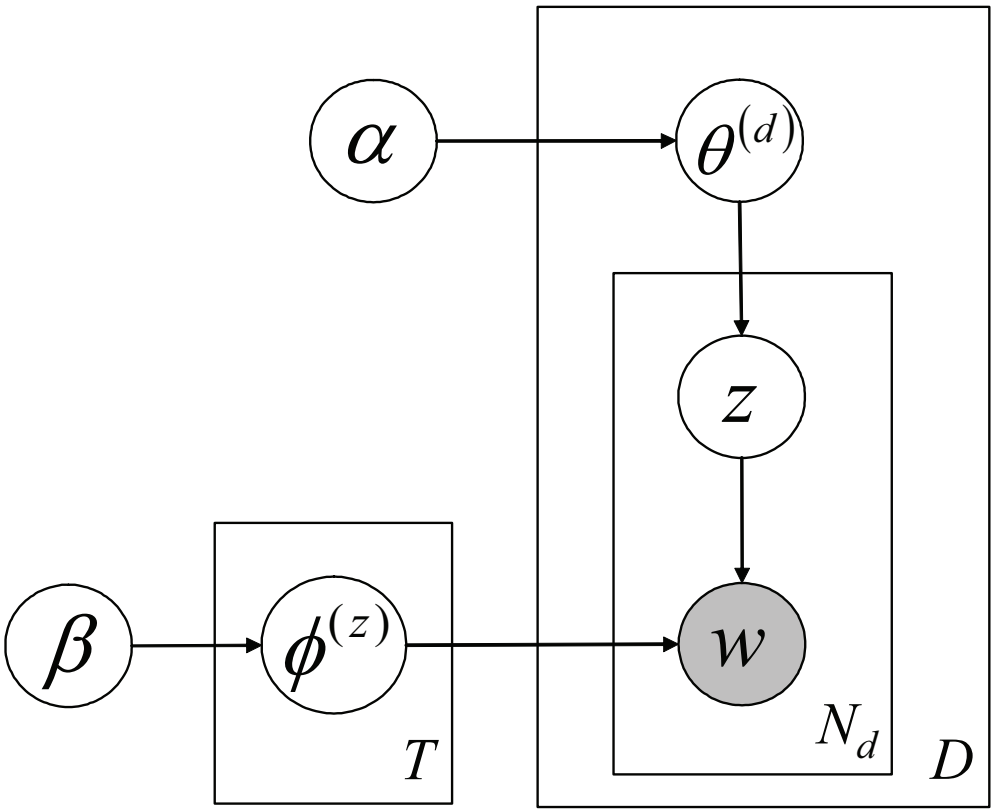


- Nodes are random variables; edges indicate dependence.
- Shaded nodes are observed; plates $\approx$ replicated variables.

The Graphical Model



The Graphical Model



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

- $P(z_i = j)$ is probability that jth topic was sampled for ith word token
- $P(w_i|z_i = j)$ is probability of word w_i under topic j
- $\phi^{(j)} = P(z_i = j)$
- $\theta^{(d)} = P(z)$

Aside: Multinomial Distribution

For $x_i \in \{0, \dots n\}$

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

When $n = 1$ the multinomial distribution simplifies to:

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

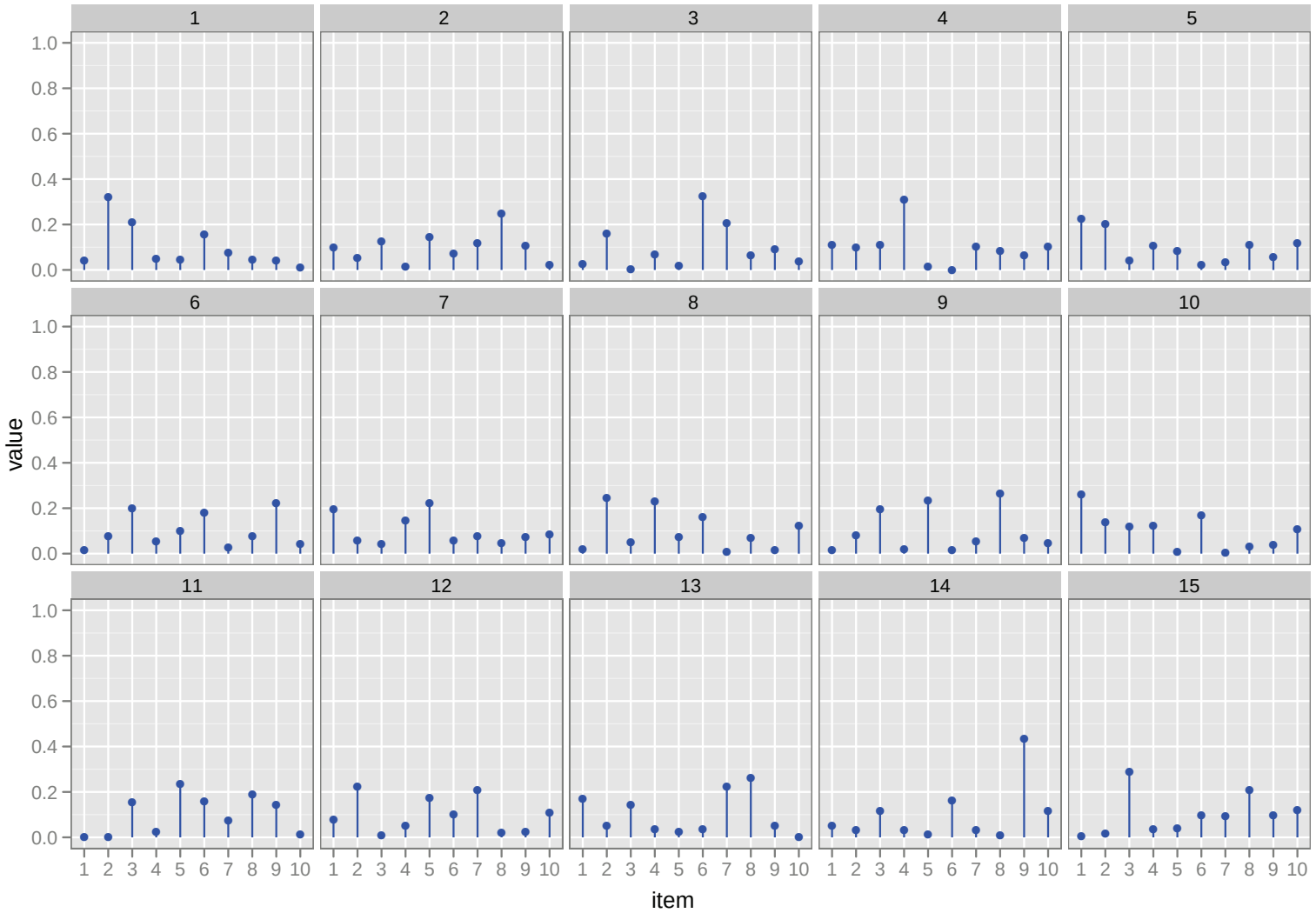
- unigram language model, 1-of-V coding ($d = V$ vocab size)
- x_i indicates word i of the vocabulary observed ($x_i = 1$ if word i is observed and 0 otherwise)
- $\theta_i = P(w_i)$ the probability that word i is seen

Dirichlet Distribution

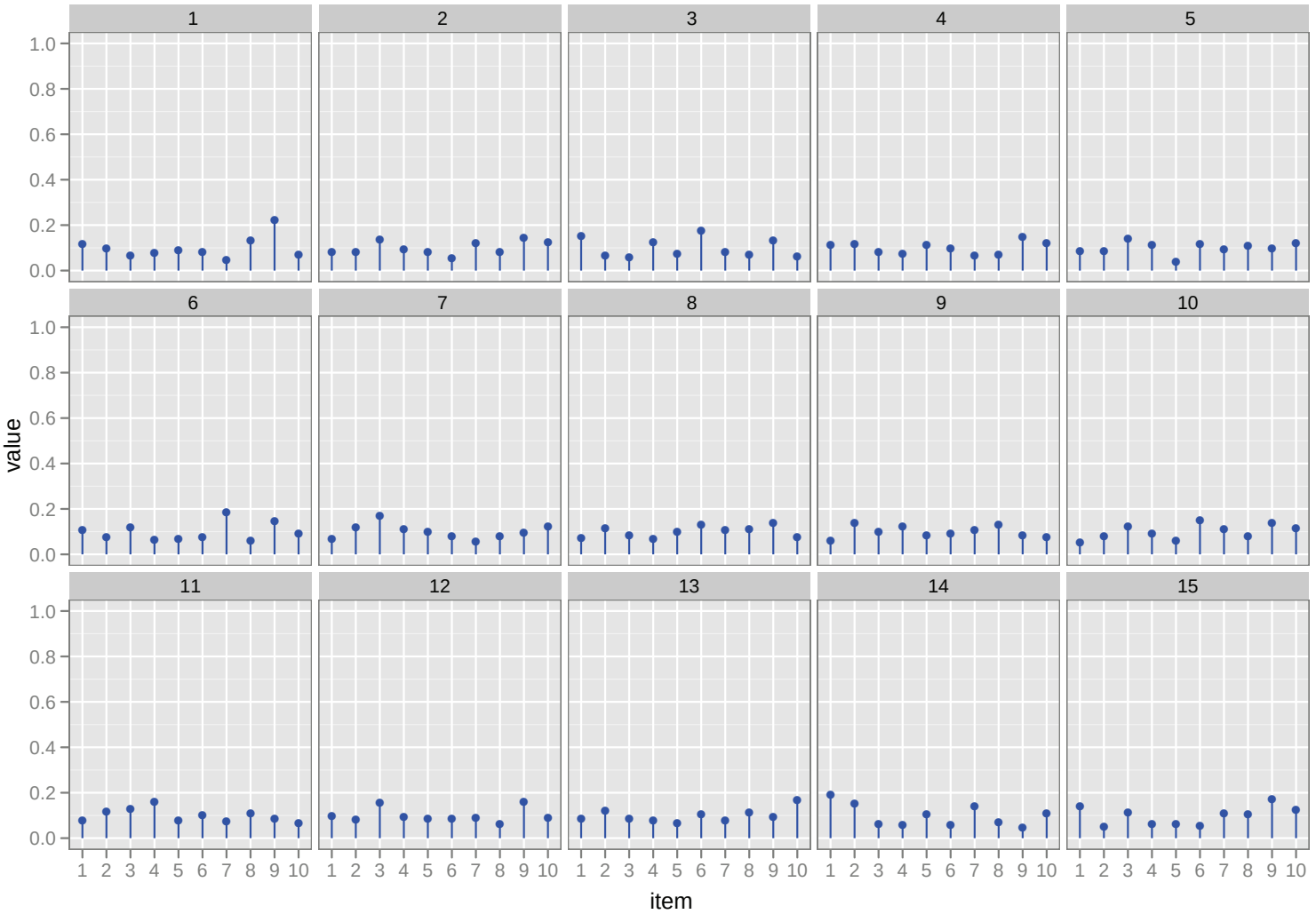
$$\text{Dir}(\alpha_1, \ldots, \alpha_T) \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_{j=1}^{T} p_j^{a_j - 1}$$

- The Dirichlet distribution is an exponential family distribution over the simplex, i.e., positive vectors that sum to one
- It is **conjugate to the multinomial**. Given a multinomial observation, the posterior distribution of θ is a Dirichlet.
- Parameter α smoothes topic distribution in the document.
- Parameter β smoothes word distribution in every topic.

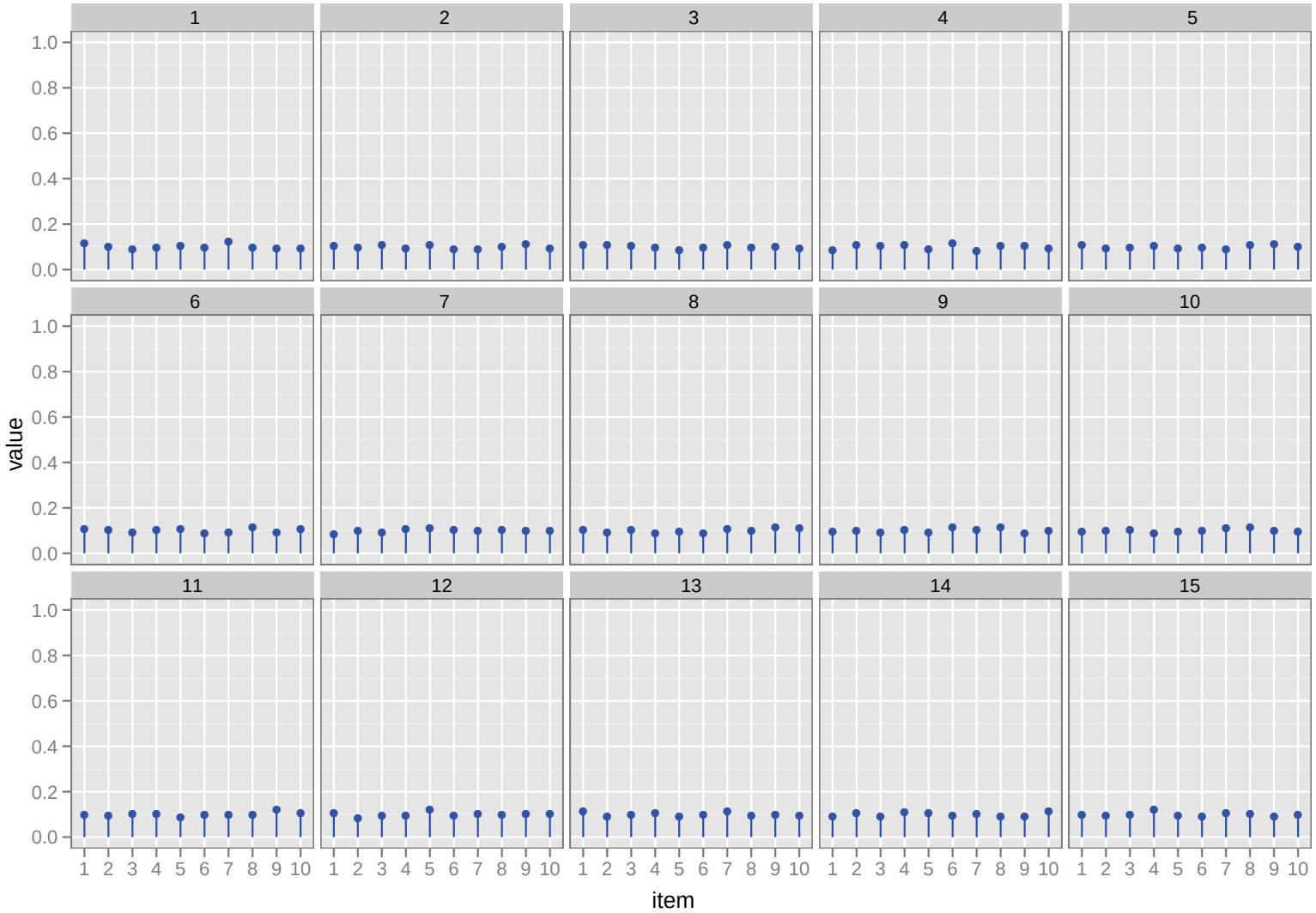
$\alpha = 1$


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

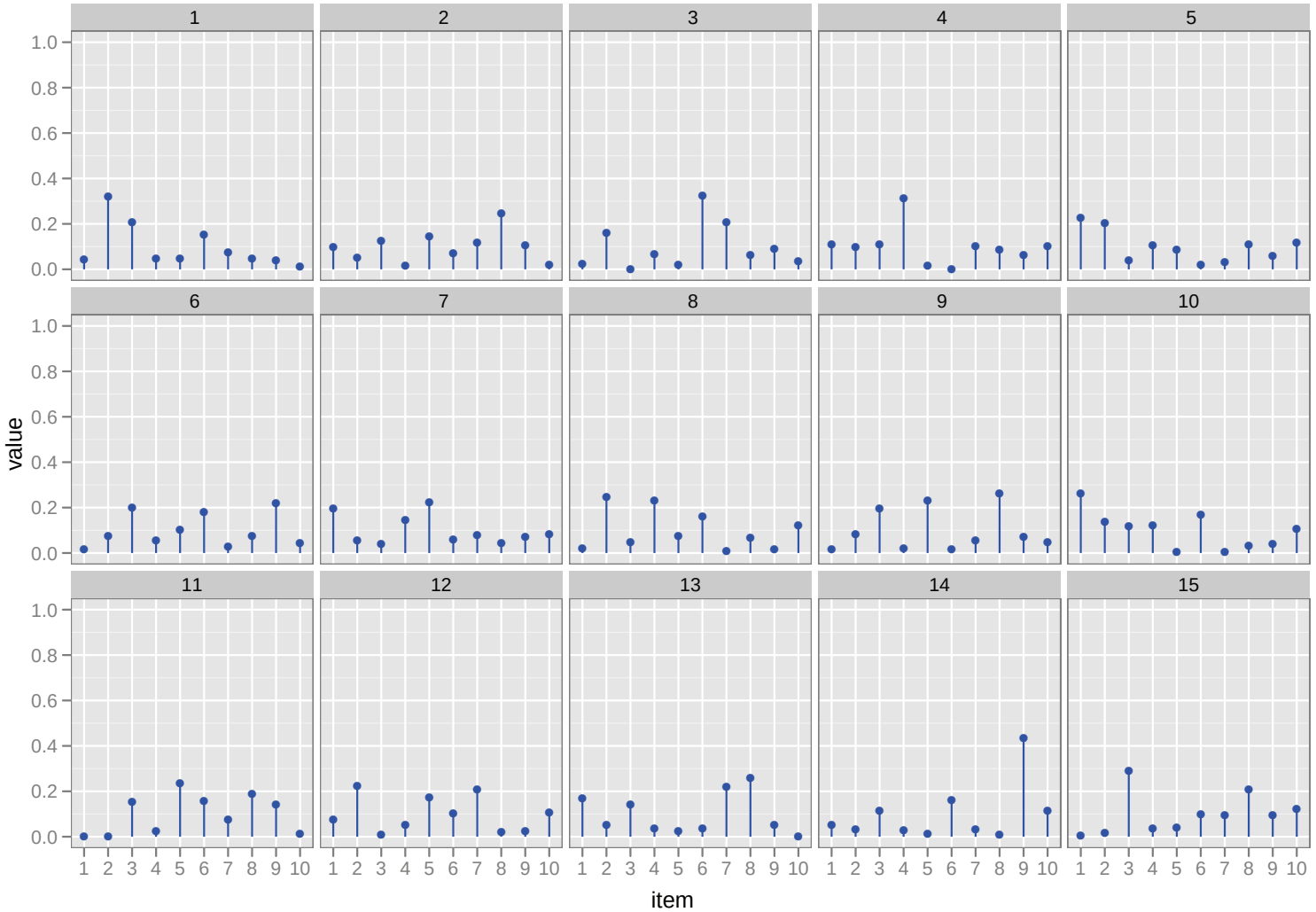
$\alpha = 10$


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

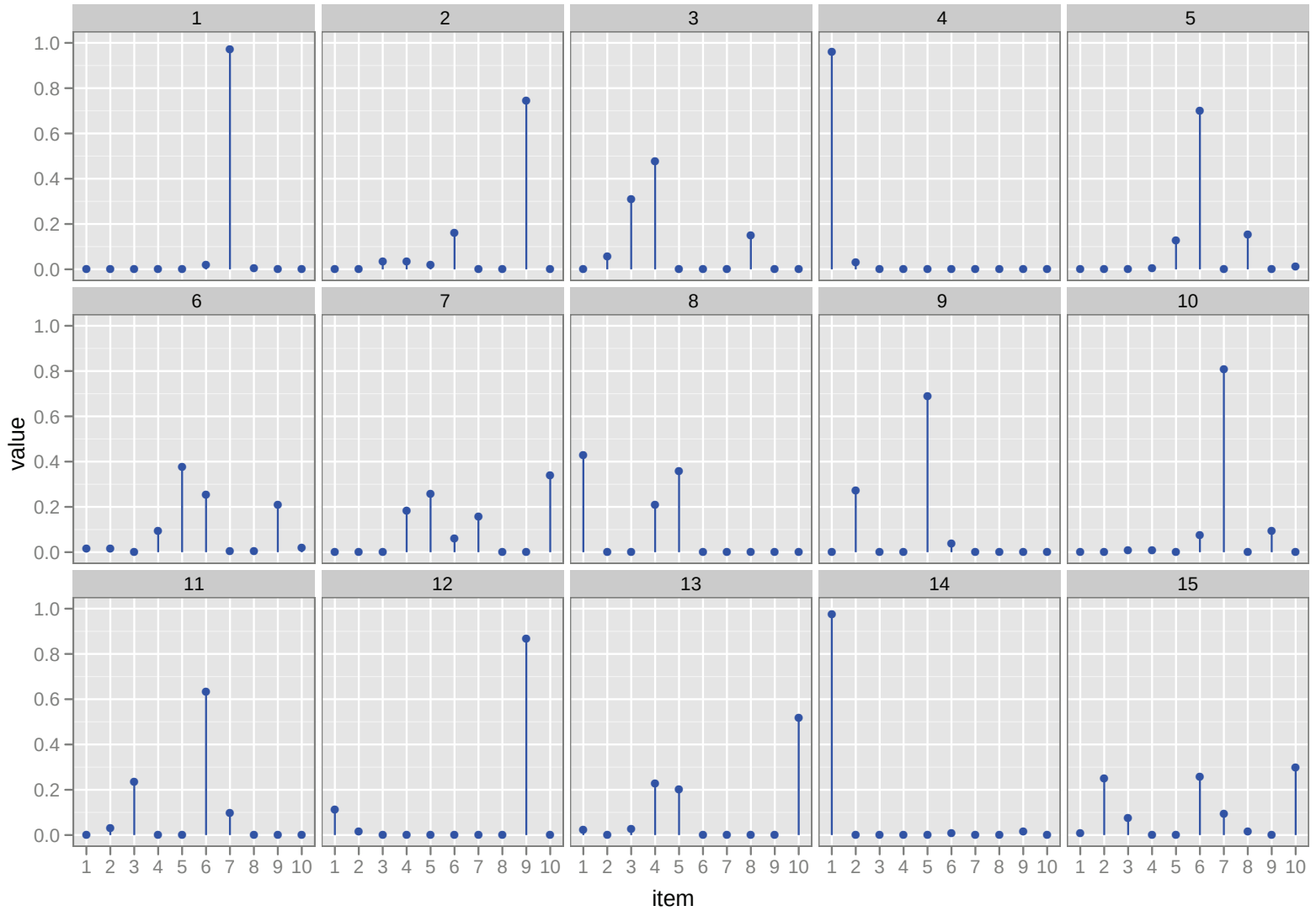
$\alpha = 100$


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

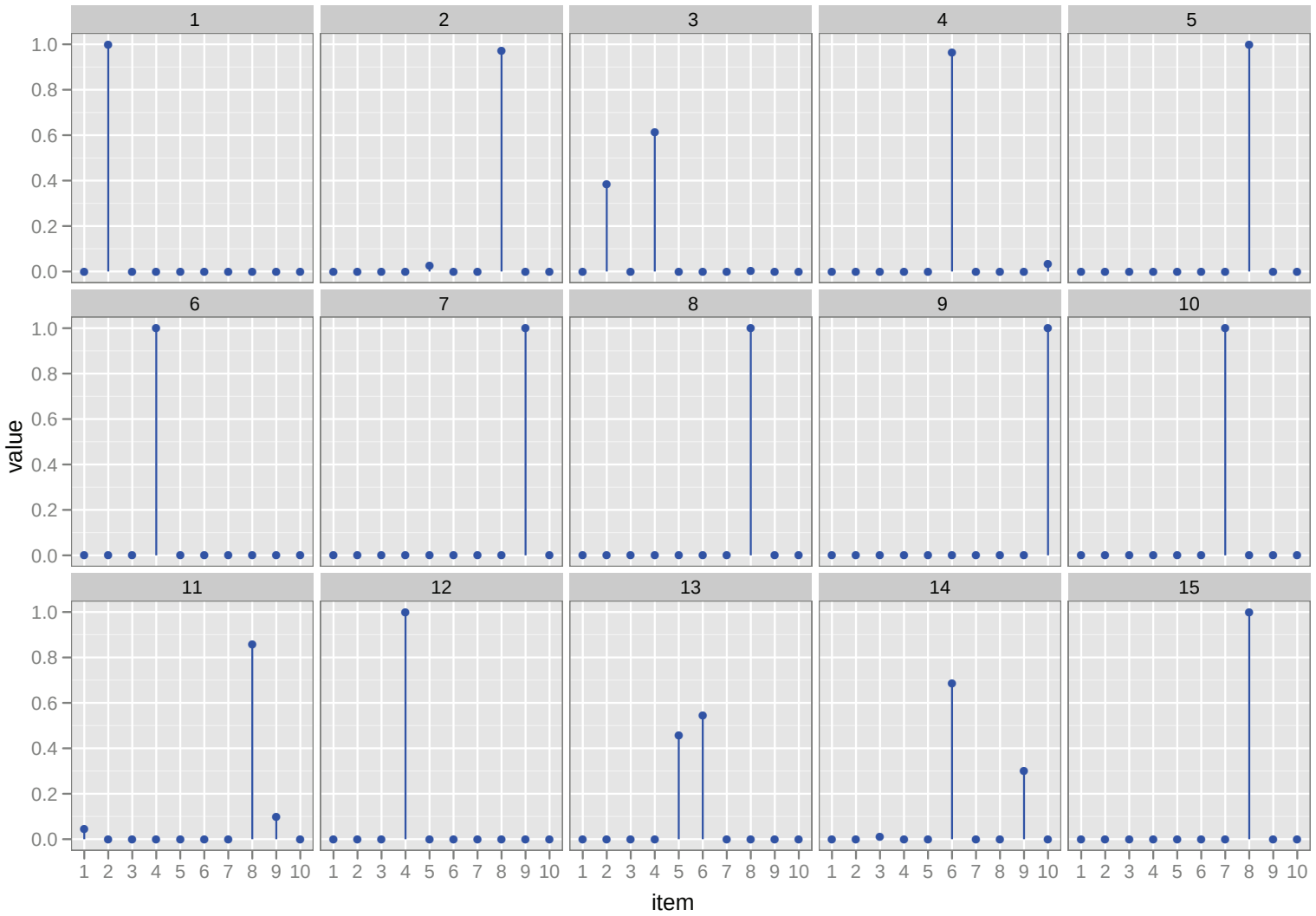
$\alpha = 1$


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

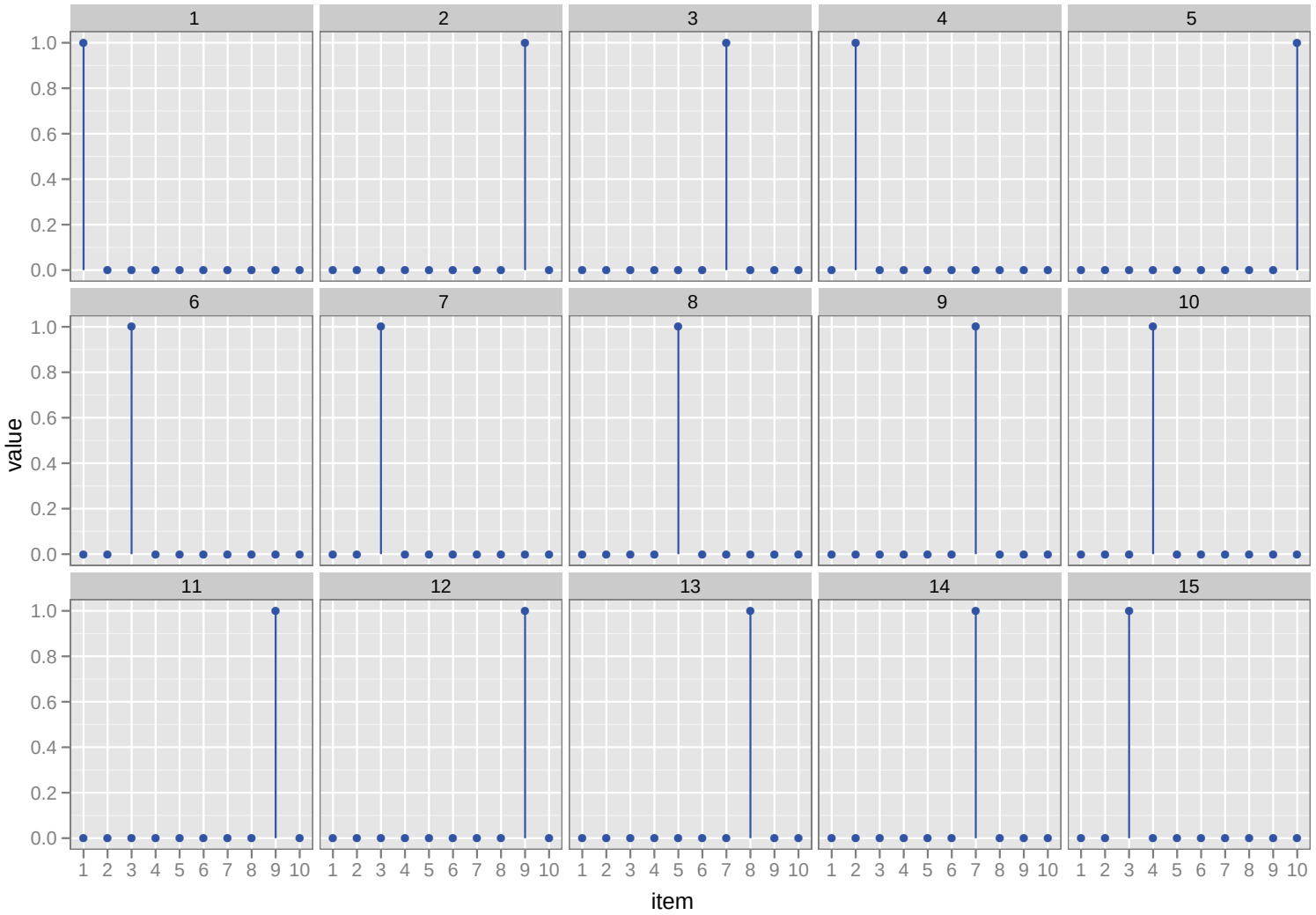
$\alpha = 0.1$


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

Inference with Gibbs Sampling

- 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.
  - Choose the most probable assignment.
- Iterate until convergeance.

Gibbs Sampling

- Collection of documents is a set of word indices w_i and document indices d_i, for each word token i.

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

- 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 WxT
C^{DT}: matrix of counts with dimensions DxT

Posterior Estimates of β and θ

$$\phi_{ij} = \frac{C_{ij}^{WT} + \beta}{\sum_{t=1}^{W} C_{w_i j}^{WT} + W\beta} \qquad \theta_{dj} = \frac{C_{d_i j}^{DT} + \alpha}{\sum_{t=1}^{T} C_{d_i t}^{DT} + T\alpha}$$

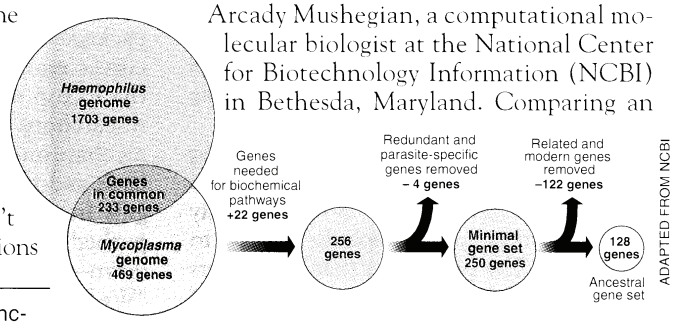
- ϕ_{ij}: normalizes word-topic count matrix (probability of word type i for topic j)
- θ_{dj}: normalizes the counts in the document-topic count matrix (probability that document d belongs to topic j).
- technically the posterior mean of the paramets ϕ and θ

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.

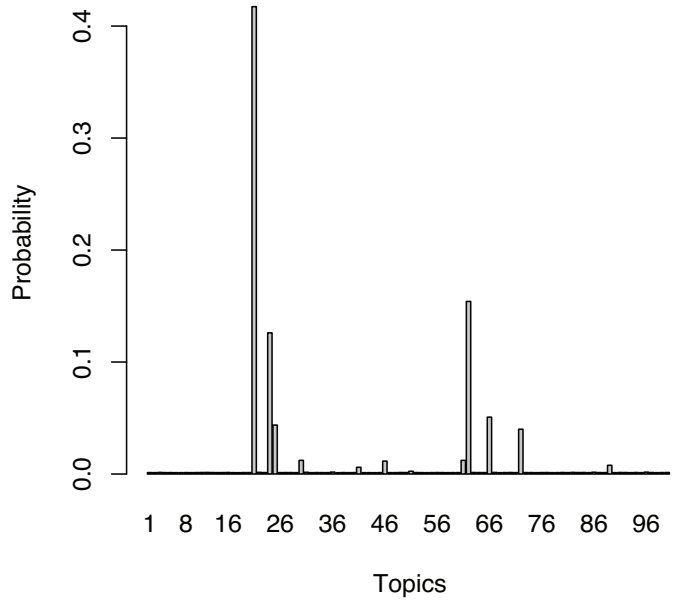
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just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

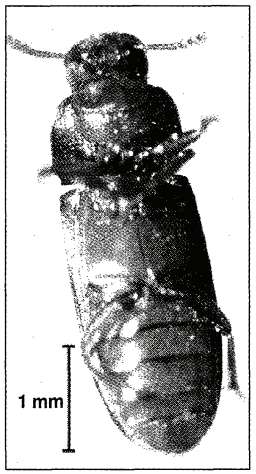
Example Inference

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*).



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

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-

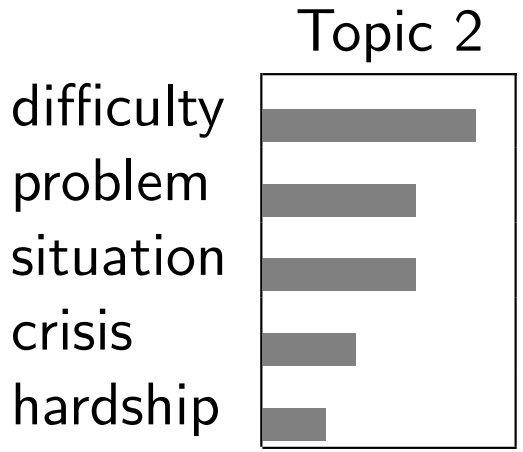
The authors are in the Department of Biology, Imperial College at Silwood Park, Ascot, Berks, SL5 7PZ UK. E-mail: m.hassell@ic.ac.uk

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

What does this have to do with semantics?

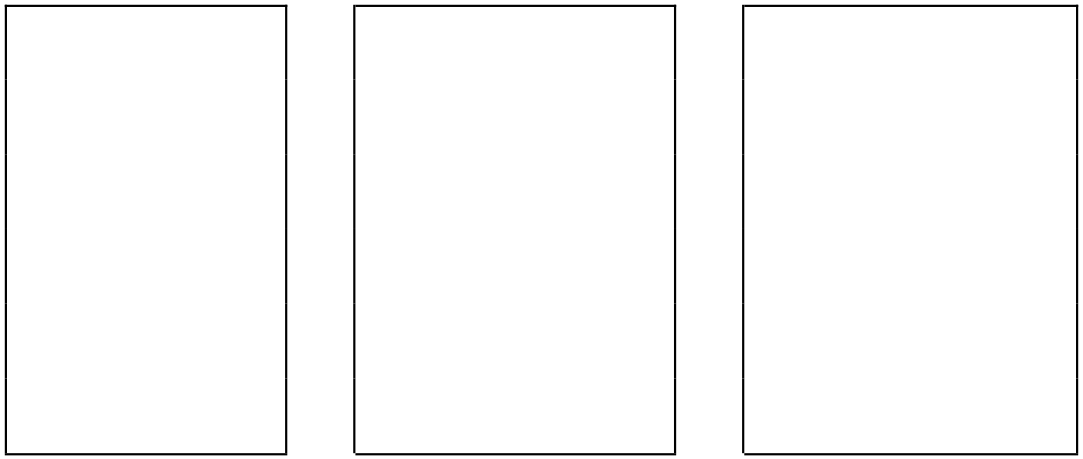
	Topic 1	Topic 2	Topic *n*
practical	0.39	0.02	...
difficulty	0.03	0.44	...
produce	0.06	0.17	...



- Topics are the dimensions of the space (500, 1000)
- Vector components: probability of word given topic
- Topics correspond to coarse-grained sense distinctions
- Cosine similarity can be used (probabilistic alternatives)

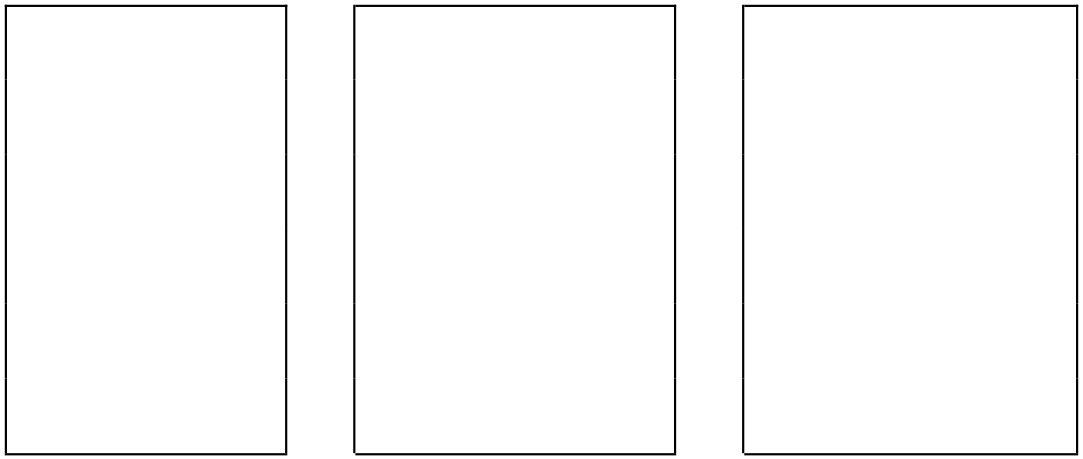
Model Evaluation

- Griffiths et al. 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



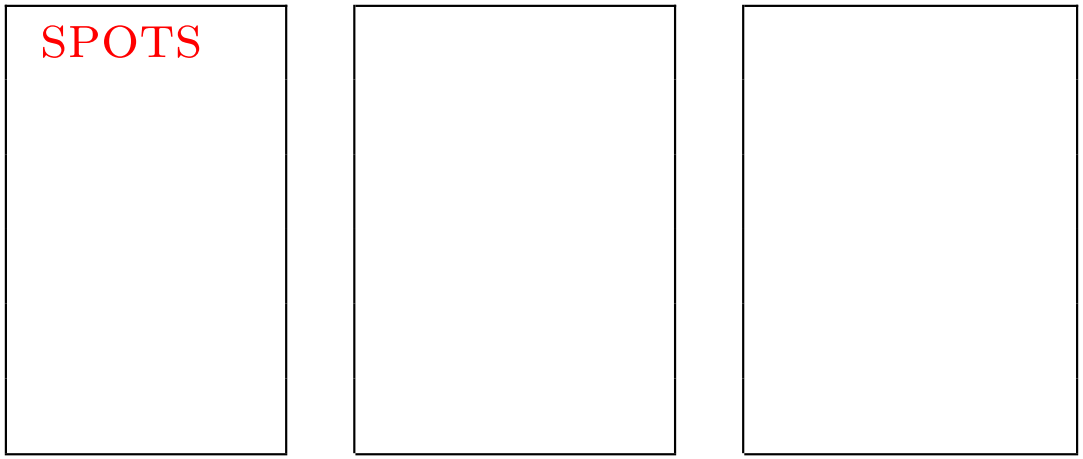
Model Evaluation

- Griffiths et al. 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. 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. 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. 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	RICE	
DOG		
DIRTY		
DIRT		
STRIPES		
DARK		

Model Evaluation

- Griffiths et al. 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	RICE	
DOG	CHINESE	
DIRTY	WEDDING	
DIRT	FOOD	
STRIPES	WHITE	
DARK	CHINA	

Model Evaluation

- Griffiths et al. 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	RICE	SEPARATE
DOG	CHINESE	
DIRTY	WEDDING	
DIRT	FOOD	
STRIPES	WHITE	
DARK	CHINA	

Model Evaluation

- Griffiths et al. 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.

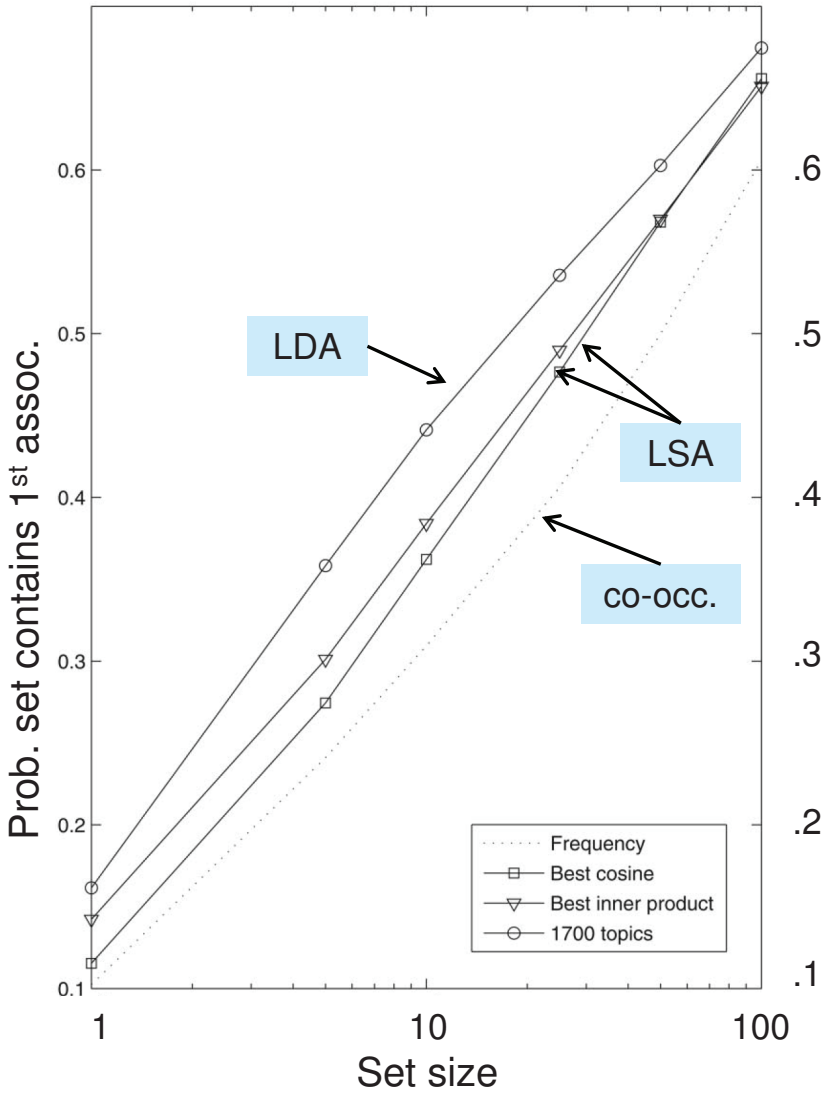
SPOTS	RICE	SEPARATE
DOG	CHINESE	DIVIDE
DIRTY	WEDDING	DIVORCE
DIRT	FOOD	PART
STRIPES	WHITE	SPLIT
DARK	CHINA	REMOVE

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

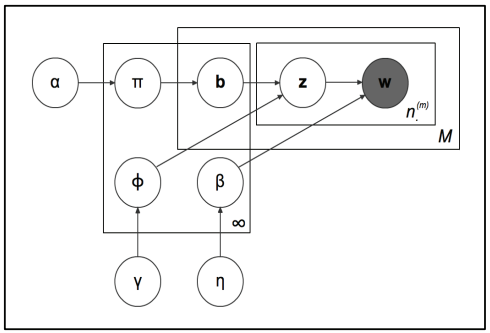


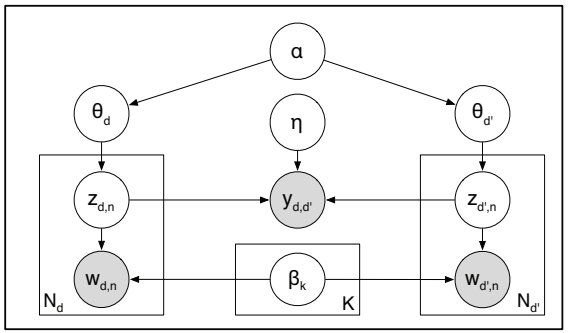
If we take the top n associates returned by the model (set size), what is the probability that this set contains the true first associate?

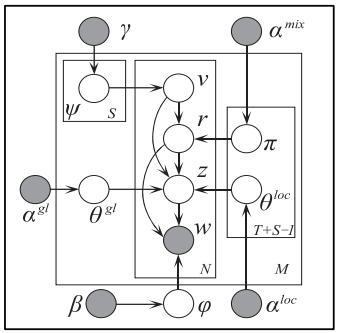
Discussion

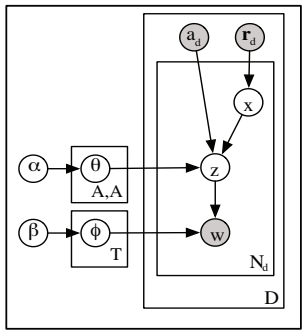
- LSA is useful in practice: IR, unsupervised morphology, essay scoring, text coherence, language modeling, summarization.
- LDA gave rise to many machine learning papers: Hierarchical LDA, nested LDA, author-topic model, topics over time, topics + syntax, etc.
- So far, limited work showing these models useful in NLP.
- LDA for discourse segmentation, extensions for text classification, WSD, tracking social phenomena.
- In comparisons, LSA often performs as well or better.

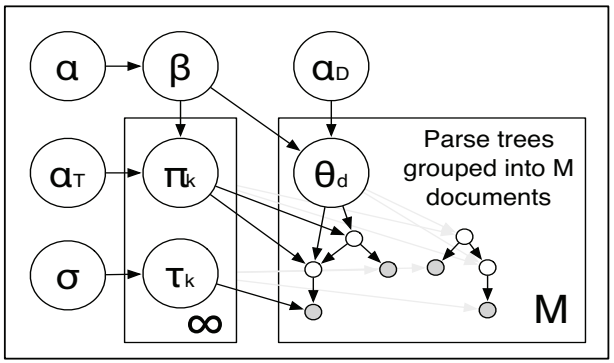
Discussion

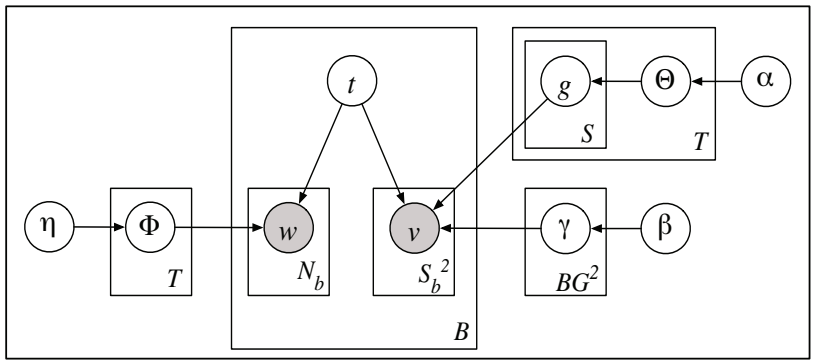

α
π
b
z
w
$n_{\cdot}^{(m)}$
M
ϕ
β
∞
γ
η

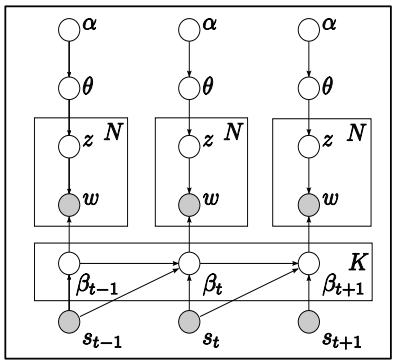

α
θ_d
η
$\theta_{d'}$
$z_{d,n}$
$y_{d,d'}$
$z_{d',n}$
$w_{d,n}$
β_k
$w_{d',n}$
N_d
K
$N_{d'}$


γ
α^{mix}
ψ
S
ν
r
z
w
N
π
θ^{loc}
T+S−1
M
α^{gl}
θ^{gl}
β
φ
α^{loc}


a_d
r_d
x
α
θ
A,A
z
β
ϕ
T
w
N_d
D


α
β
α_D
α_T
π_k
θ_d
Parse trees
grouped into M
documents
σ
τ_k
∞
M


t
g
Θ
α
S
T
η
Φ
w
v
γ
β
T
N_b
S_b^2
BG^2
B


α
θ
z
N
w
K
β_{t-1}
β_t
β_{t+1}
s_{t-1}
s_t
s_{t+1}

References

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