Topic Models and Latent Dirichlet Allocation

Review of Probabilistic Models

Model	Inference
Simple Generative Models	Exact Bayesian
Conditional Models – Regression/Classification	Exact Bayesian – conjugacy, exp family
Latent Variable – Mixture Models	Expectation Maximization
Latent Variable – Mixed Membership/Topic Models	Approximate Inference
Graphical Models	- Variational Inference
- Directed	- Markov Chain Monte Carlo
- Undirected	- Other VI: Expectation Propagation, Loopy Belief Propagation
- HMM	- Other Sampling: Collapsed Gibbs, Langevin Dynamics, Sequential MCMC, Particle Filters

Probabilistic Topic Models

- Given a collection of objects (each object a set of discrete tokens), find topics in the collection
- Each object annotated as to how much each topic is present in that object
- Each topic is represented by a collection of token with weights

Example

- Collection of articles published in the journal Science over past century

Words

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

Topics (Themes)

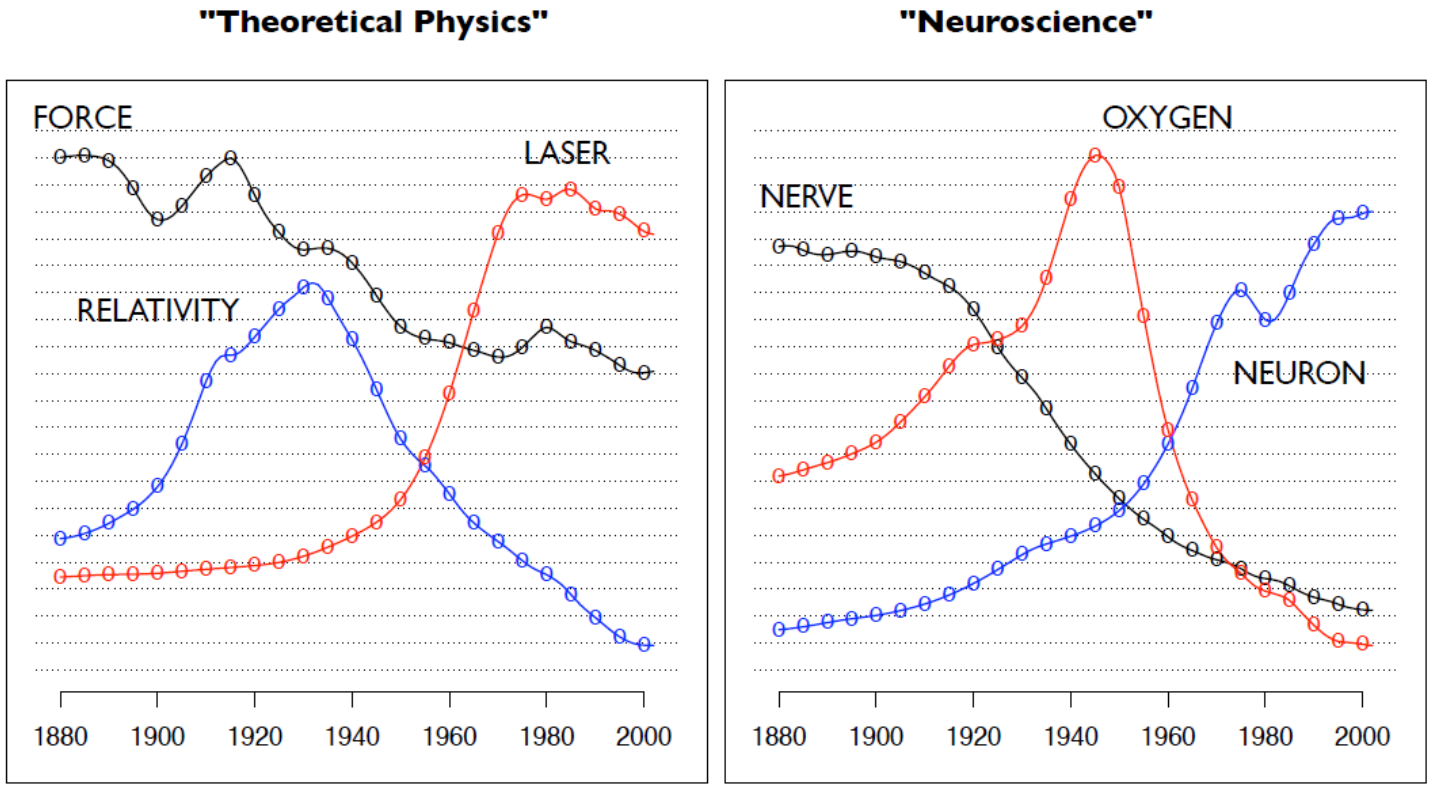
Motivation

Topic modeling provides methods for automatically organizing, understanding, searching, and summarizing large electronic archives.

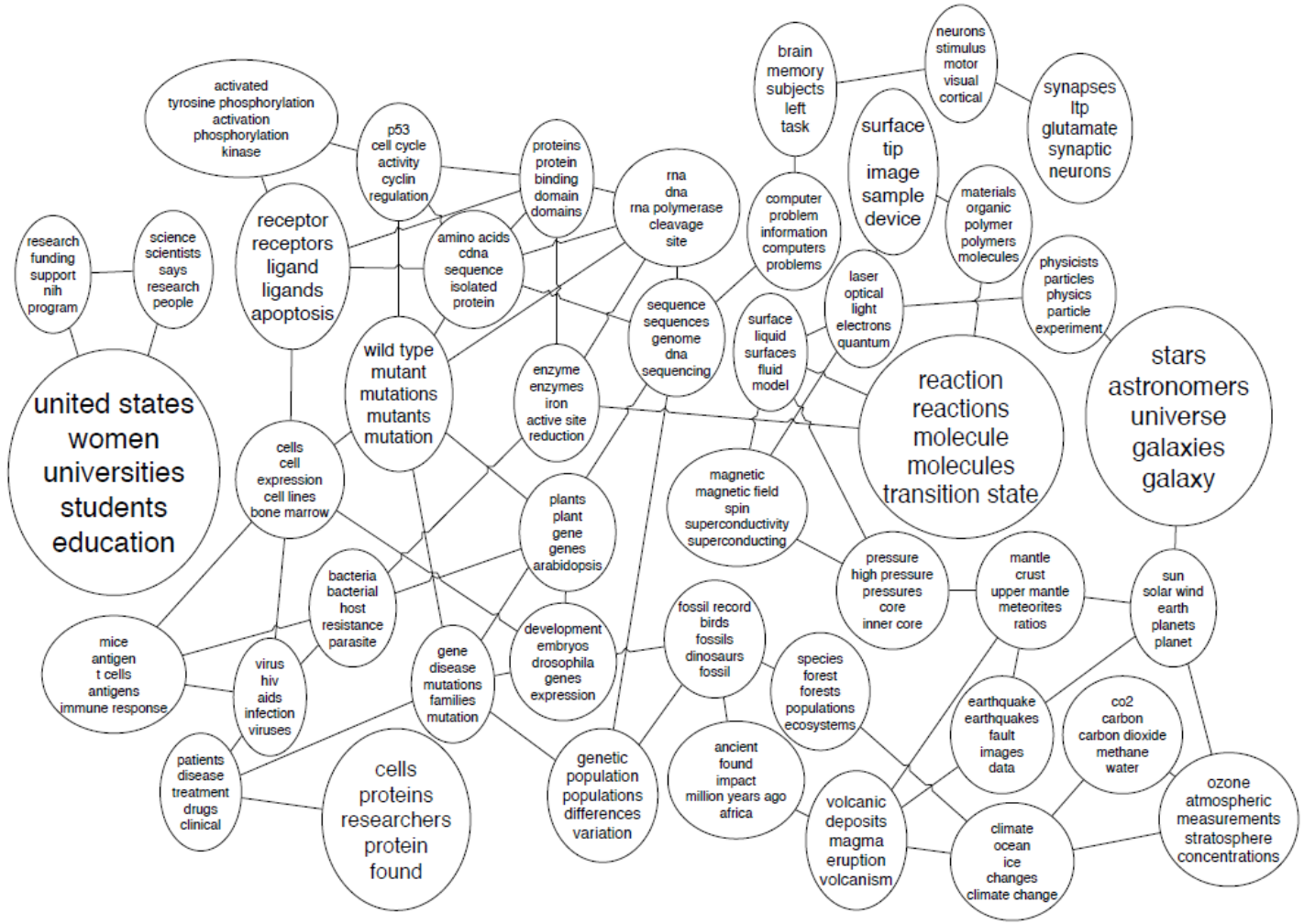
1. Discover the hidden themes that pervade the collection.
2. Annotate the documents according to those themes.
3. Use annotations to organize, summarize, and search the texts.

- Also applicable to other types of data, beyond text documents, e.g.,
  - Image collection: Each image is a "document" which is a bag of "visual words"

Application: Topic Tracking



Application: Topic Maps



Application: Image Annotation

SKY WATER TREE
MOUNTAIN PEOPLE

SCOTLAND WATER
FLOWER HILLS TREE

SKY WATER BUILDING
PEOPLE WATER

FISH WATER OCEAN
TREE CORAL

PEOPLE MARKET PATTERN
TEXTILE DISPLAY

BIRDS NEST TREE
BRANCH LEAVES

Simpler Topic Models

- Naïve Bayes – a document has a single topic – mixture of unigram models

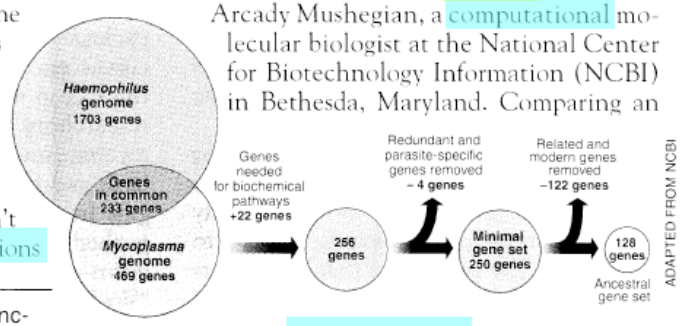
- Latent Semantic Indexing – topic as a low dimensional projection of word vectors – not a Bayesian model

Latent Dirichlet Allocation

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.

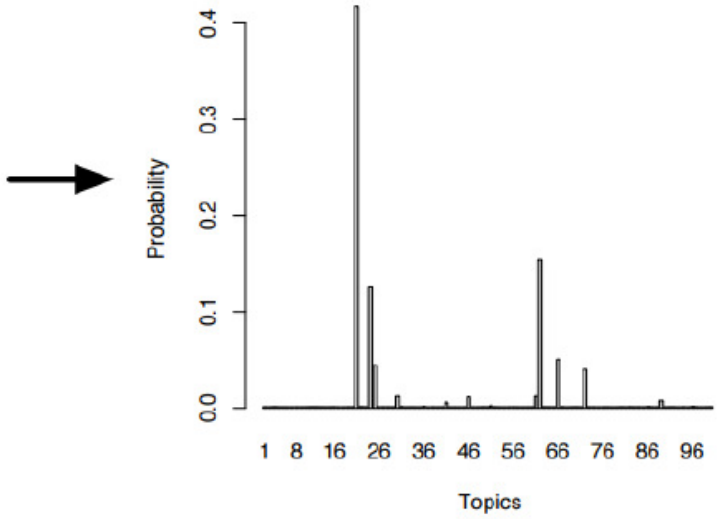
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just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

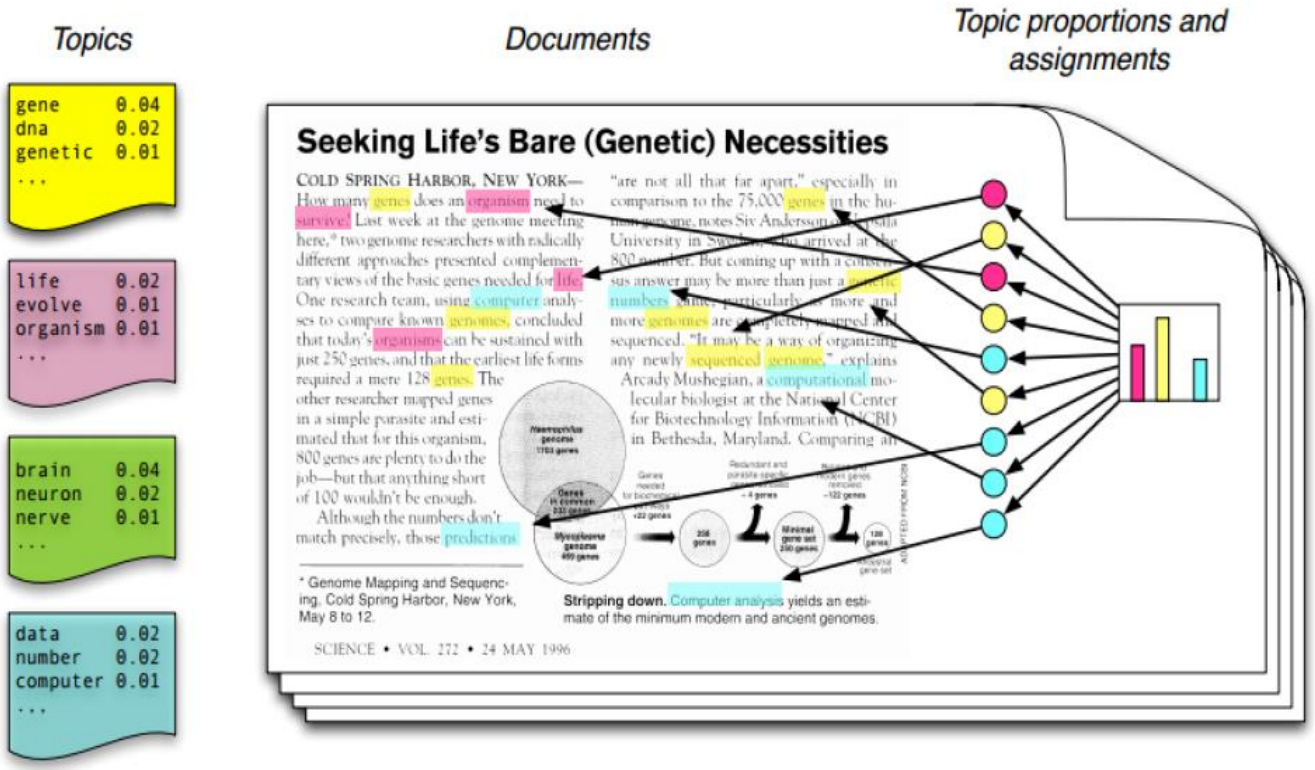
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996



Simple intuition: Documents exhibit multiple topics.

Generative Model Illustration



- Each **topic** is a distribution over words
- Each **document** is a mixture of corpus-wide topics
- Each **word** is drawn from one of these topics
- We only observe the words within the documents and the other structure are **hidden variables**.

Generative Model

To generate a document:

1. Randomly choose a distribution over topics

2. For each word in the document

   a. randomly choose a topic from the distribution over topics
   b. randomly choose a word from the corresponding topic (distribution over the vocabulary)

- Note that we need a distribution over a distribution (for step 1)

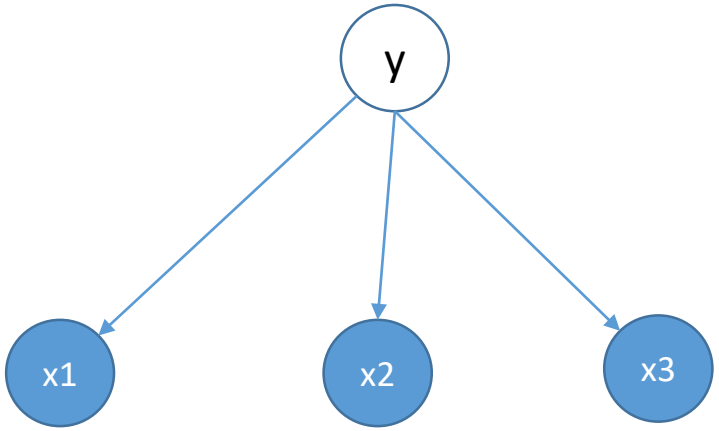
- Note that words are generated independently of other words (unigram bag-of-words model)

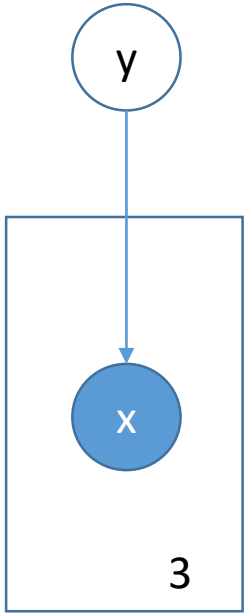
Notations

- Some notation:
  - $\beta_{1:K}$ are the topics where each β_k is a distribution over the vocabulary
  - θ_d are the topic proportions for document d
  - $\theta_{d,k}$ is the topic proportion for topic k in document d
  - z_d are the topic assignments for document d
  - $z_{d,n}$ is the topic assignment for word n in document d
  - w_d are the observed words for document d

Plate Diagrams (Graphical Model)

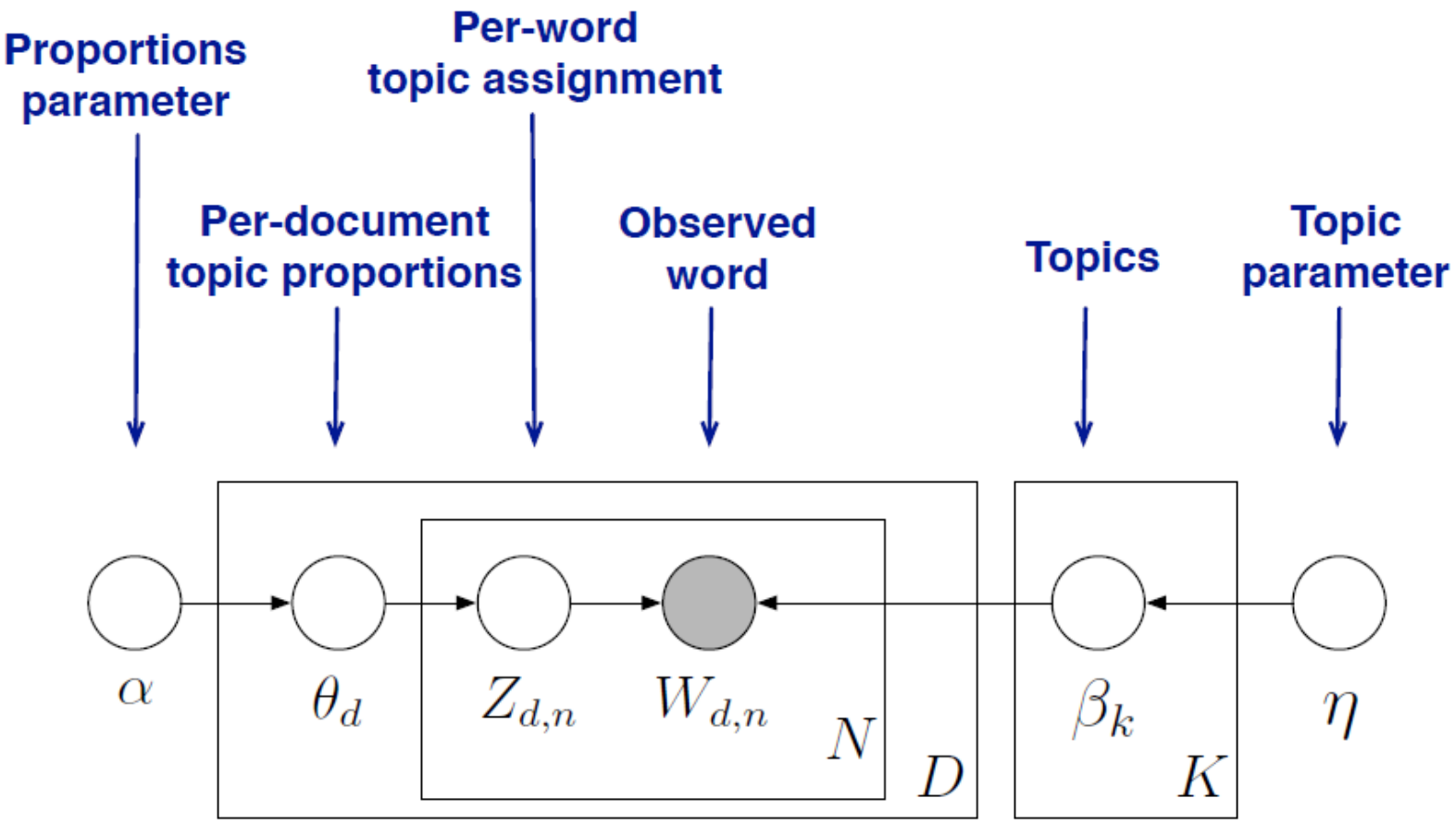
$p(x1,x2, x3, y) = p(x1|y)p(x2|y)p(x3|y)p(y)$

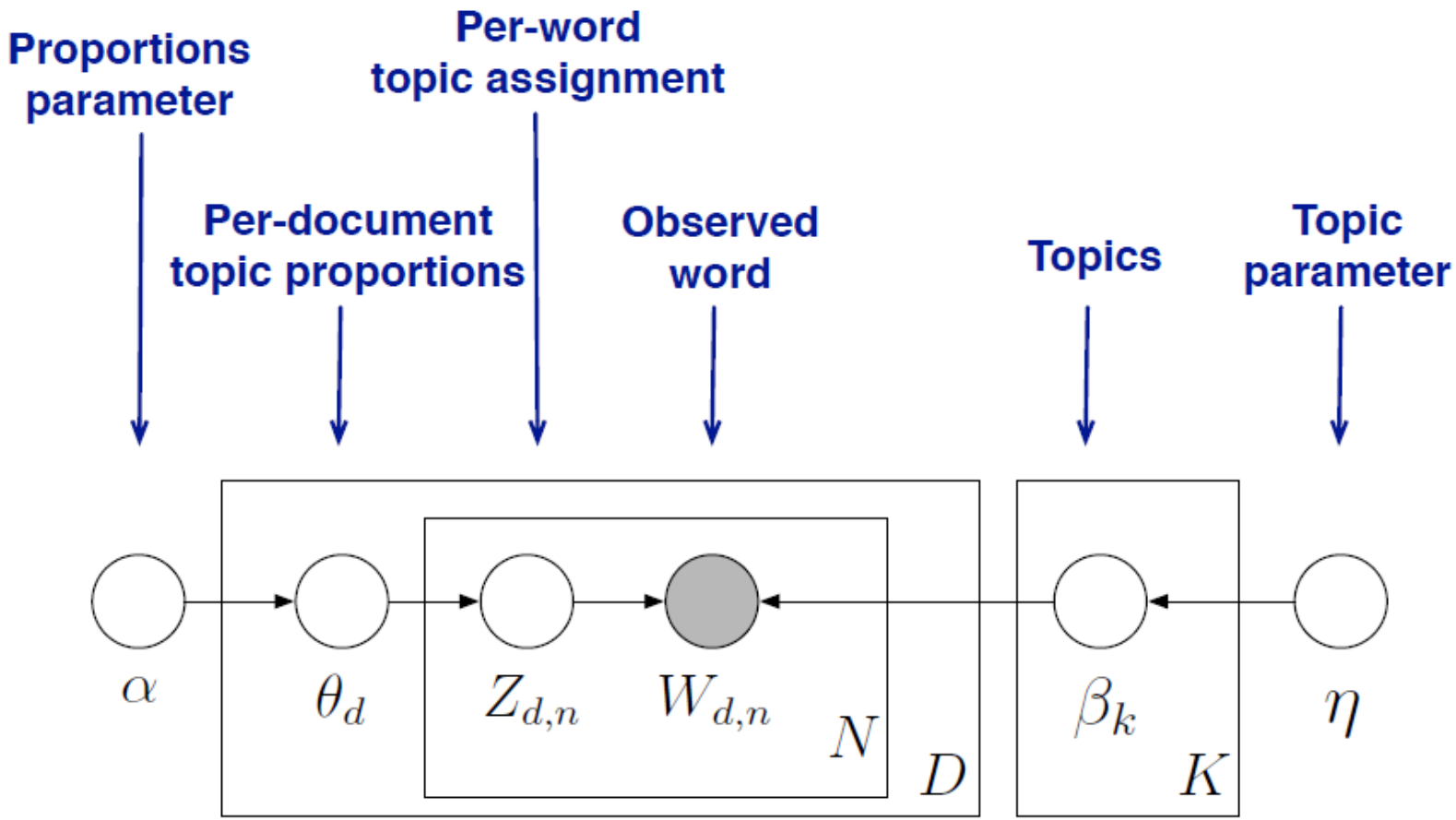




- Nodes are random variables/parameters
- Edges are direct influences
- Shaded nodes are observed
- Structure represents a factorization of the joint
- distribution

LDA as a Graphical Model





$$\prod_{i=1}^{K} p(\beta_i \mid \eta) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n}) \right)$$

Distributions in the Model

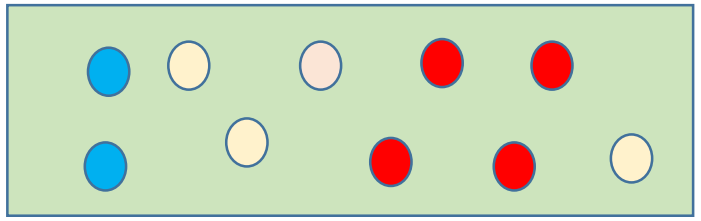
- Multinomial

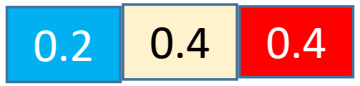
- Dirichlet – distribution over distributions

Multinomial Distribution

- A k-dimensional multinomial can be thought of as a k-faced dice
- A sample drawn from the distribution is an integer 1…k
- Parameters of the distribution is a vector [p_1, p_2, …, p_k], representing the probability of each face. The vector terms add to 1.

- Can also be thought of as a box containing balls of k different colours. The multiplicity of a colour is proportional to the parameters p_i. Sampling is equivalent to drawing a ball with replacement.

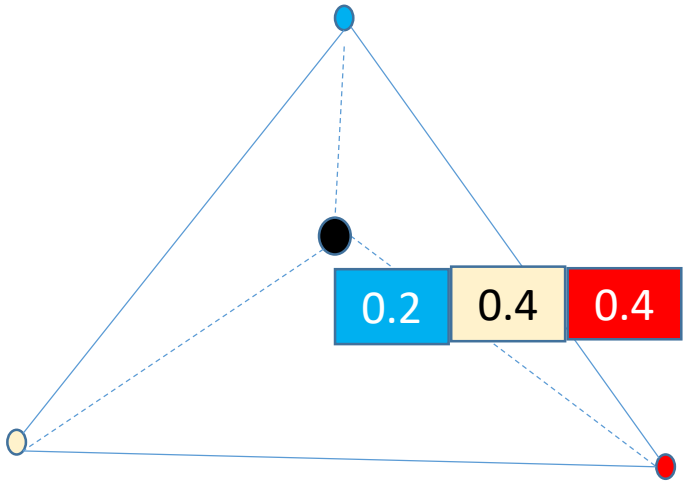




p vector

Simplex of the Multinomial Parameters

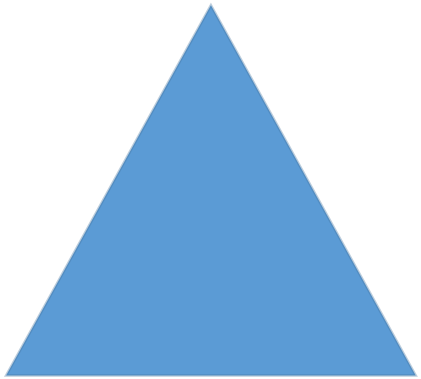
- The parameters of a k-dimensional multinomial lie in a k-1-simplex

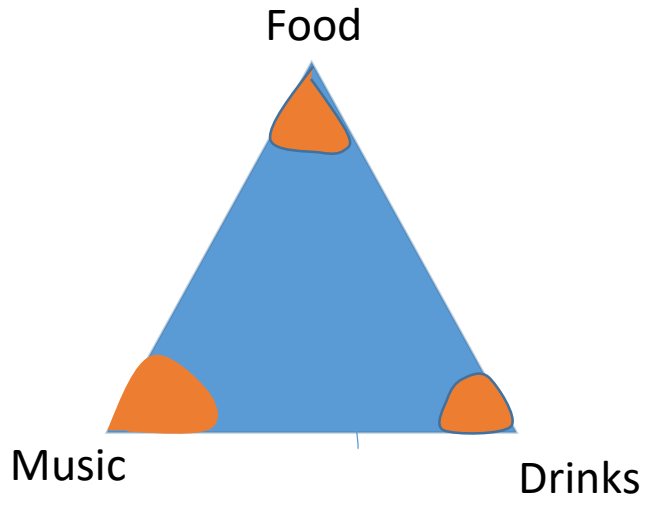


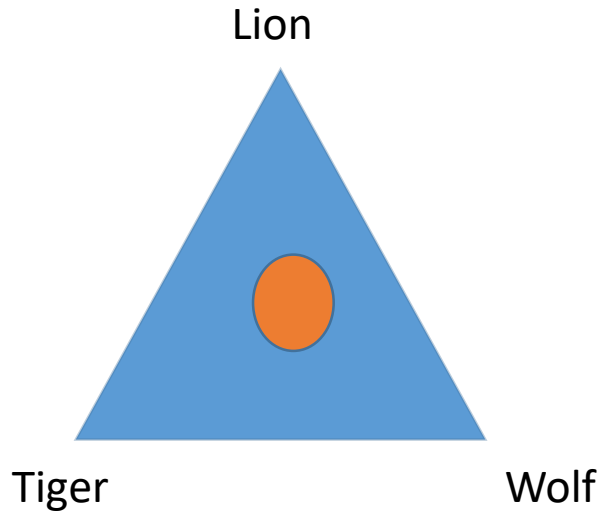
Points at the corners of the simplex has a single outcome
Points on the edges are mixture of fewer number of outcomes
Points near the centre are uniform over the outcomes (less sparse p vectors)

Dirichlet Distribution

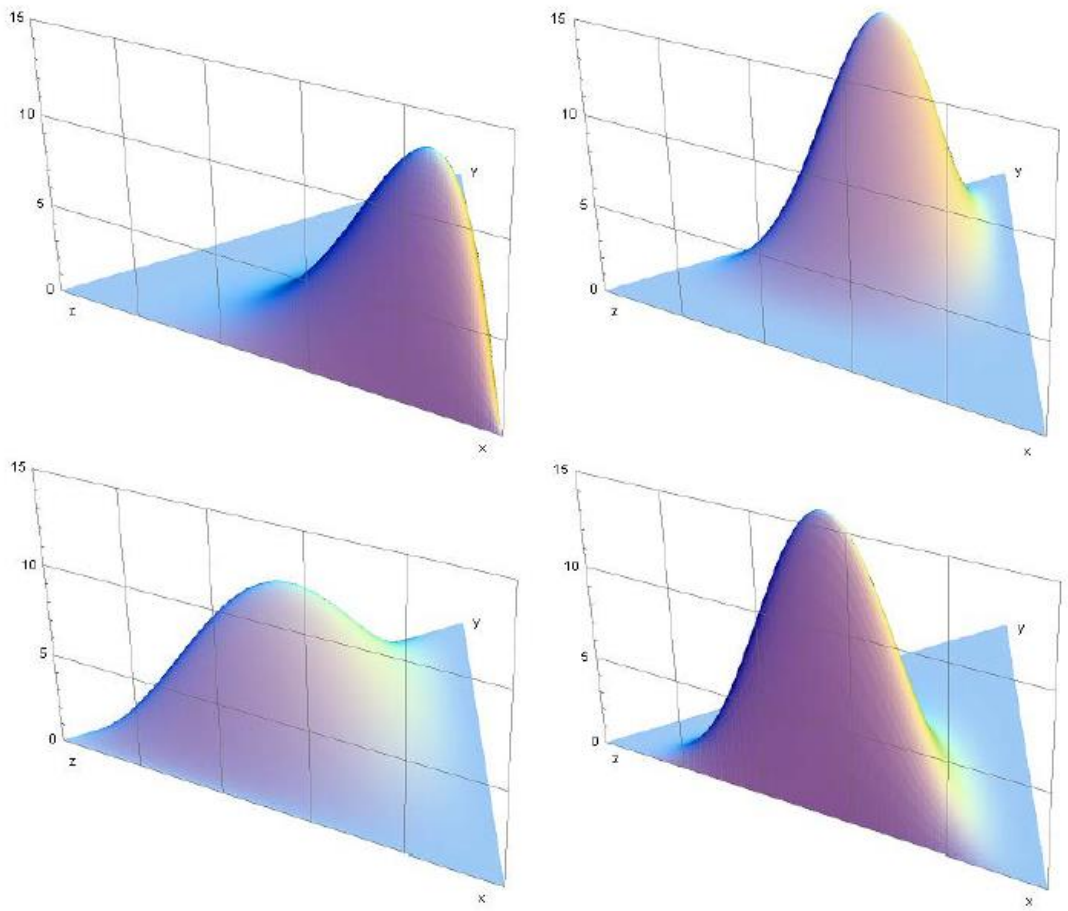
- Imagine a party of N persons being held in a triangular room.
- What is the probability of persons being found in a particular position inside the room/simplex?







Dirichlet Distributions



Distribution Functions

1. Multinomial:

$$P(X_1 = x_1, \dots, X_K = x_K) = \frac{n!}{x_1! \dots x_K!} p_1^{x_1} \dots p_K^{x_K} \qquad X_i \in \{0, \dots, n\} \qquad \sum_{i=1}^{K} X_i = n$$

2. Dirichlet: Good for modeling a distribution over distributions

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \dots \theta_K^{\alpha_K - 1} \qquad \alpha = k - dimensional\ vector \qquad \alpha_i > 0$$

$$variable\ \theta\ can\ take\ values\ in\ the\ (k-1)\ simplex: \quad \theta_i > 0 \quad and \quad \sum_{i=1}^{K} \theta_i = 1$$

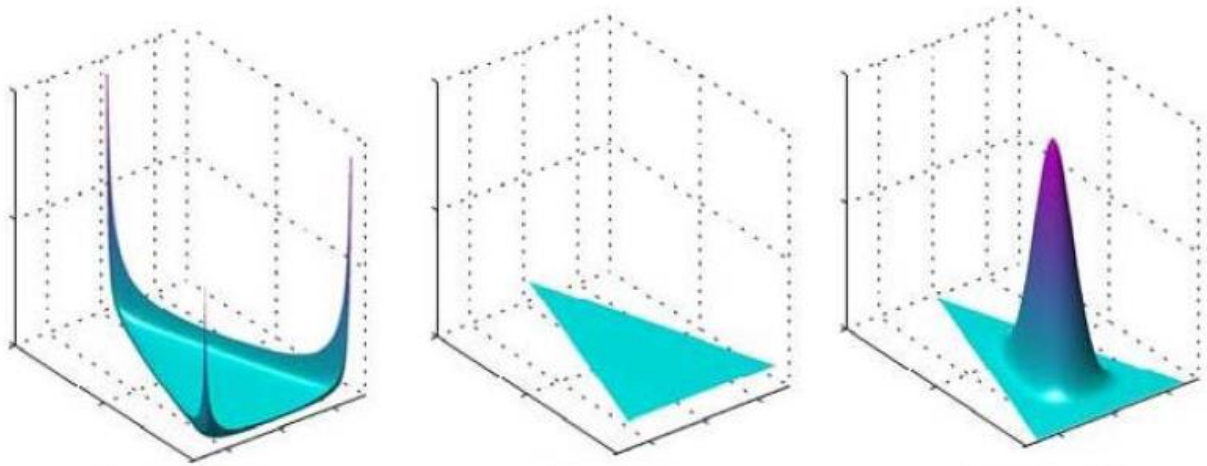
Multinomial and Dirchlet are conjugates. Belong to exponential family.

Dirichlet Distribution

Role of parameter $\vec{\alpha} = (\alpha_1, \dots, \alpha_K)$:

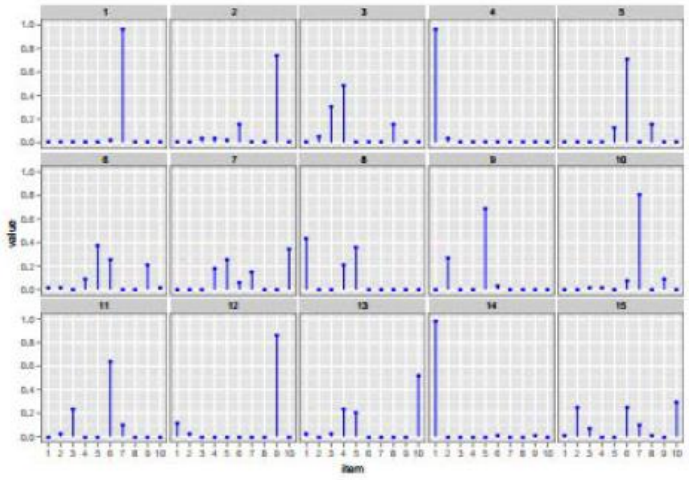
$$E\{\theta_i|\alpha\} = \frac{\alpha_i}{\sum \alpha_i}$$

Note that here we are working with symmetric (exchangeable) Dirichlet distributions meaning $\alpha_1 = \cdots = \alpha_K$

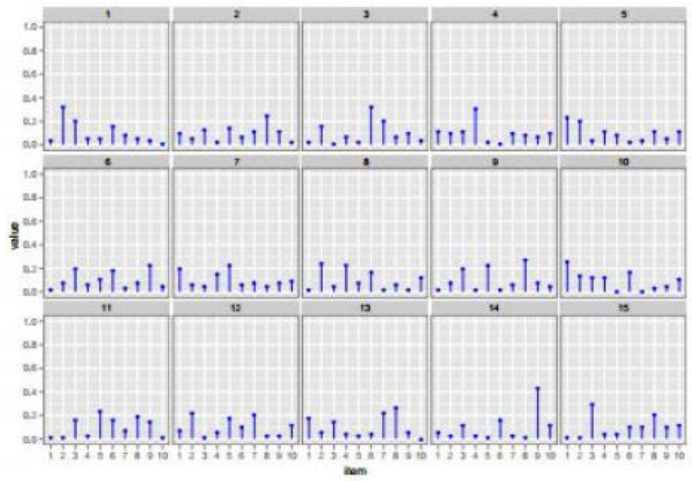


Multinomials Generated from Dirichlet Distributions

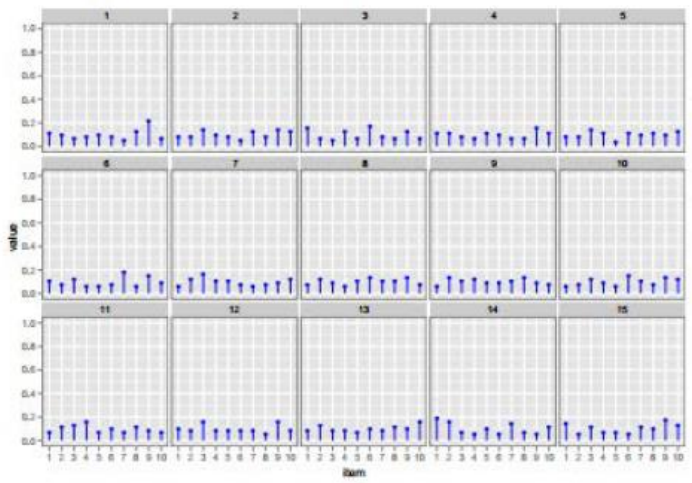
$\alpha_i = 0.1$



$\alpha_i = 1$



$\alpha_i = 10$



Formal Definition: LDA

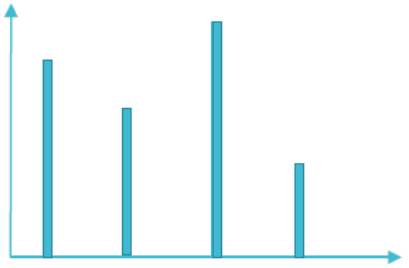
Formal definition of the model:

$$p(\beta, \theta, z, w) = \left(\prod_{i=1}^{K} p(\beta_i|\eta)\right)\left(\prod_{d=1}^{D} p(\theta_d|\alpha) \prod_{n=1}^{N} p(z_{d,n}|\theta_d) p(w_{d,n}|\beta_{1:K}, z_{d,n})\right)$$

$$(\beta_d|\eta) \sim Dir(\beta) \qquad (\theta_d|\alpha) \sim Dir(\alpha) \qquad Z_{d,n} \sim Multi(\theta_d) \qquad W_{d,n} \sim Multi(\beta_{z_{d,n}})$$

$$p(z_{d,n}|\theta_d) = \theta_{d,z_{d,n}} \qquad p(w_{d,n}|z_{d,n}, \beta_{1:K}) = \beta_{z_{d,n},w_{d,n}}$$

θ_d:



β:

	Word probabilities for each topic		
Topics			

Illustration of Generative Process

- Topic Proportion Multinomial
- Word Multinomial
- Topic-Document Dirichlet
- Topic-Word Dirichlet

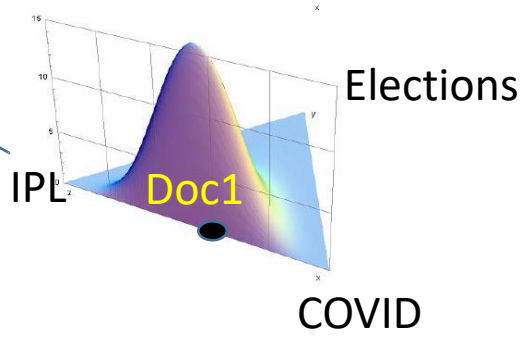
Doc1

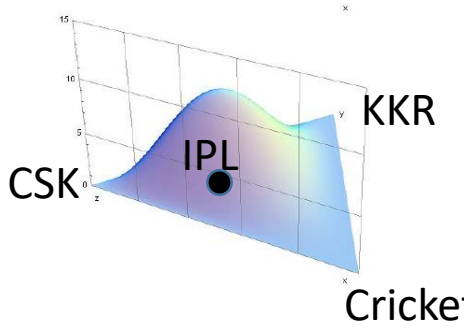
IPL, COVID, COVID, COVID

IPL

KKR, CSK, CSK, CSK, RR, MI

Doc1
KKR
SARS
Lockdown
Cricket



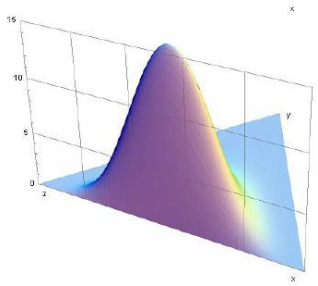


Hyperparameters: Dirichlet parameters, Doc length, No. of Topics

LDA Generative Process Summary

Multinomials

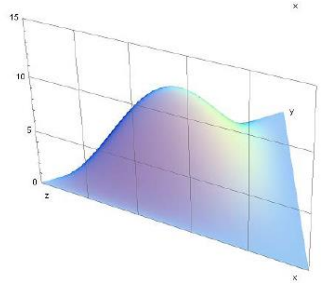
	T1	T2	T3	T4
Doc1	0.0	0.0	0.7	0.3
Doc2	0.9	0.0	0.1	0.0
Doc3	0.8	0.2	0.0	0.0
Doc4	0.0	0.0	0.8	0.2



Dir(α)

Multinomials

	W1	W2	W3	W4
T1	0.0	0.1	0.8	0.1
T2	0.0	0.0	0.7	0.3
T3	0.0	0.2	0.8	0.0
T4	0.0	0.9	0.1	0.0



Dir(β)

W1, W1, W2, W3, W1

Doc1

Why does LDA “work”?

- LDA trades off two goals.
  1. For each document, allocate its words to as few topics as possible.
  2. For each topic, assign high probability to as few terms as possible.
- These goals are at odds.
  - Putting a document in a single topic makes #2 hard:
    All of its words must have probability under that topic.
  - Putting very few words in each topic makes #1 hard:
    To cover a document’s words, it must assign many topics to it.
- Trading off these goals finds groups of tightly co-occurring words.

Training LDA

Doc1
gene
dna
rna
mutation
species

Doc2
gene
mutation
crossover
fitness
species

Doc3
dna
selection
data
statistics
genetics

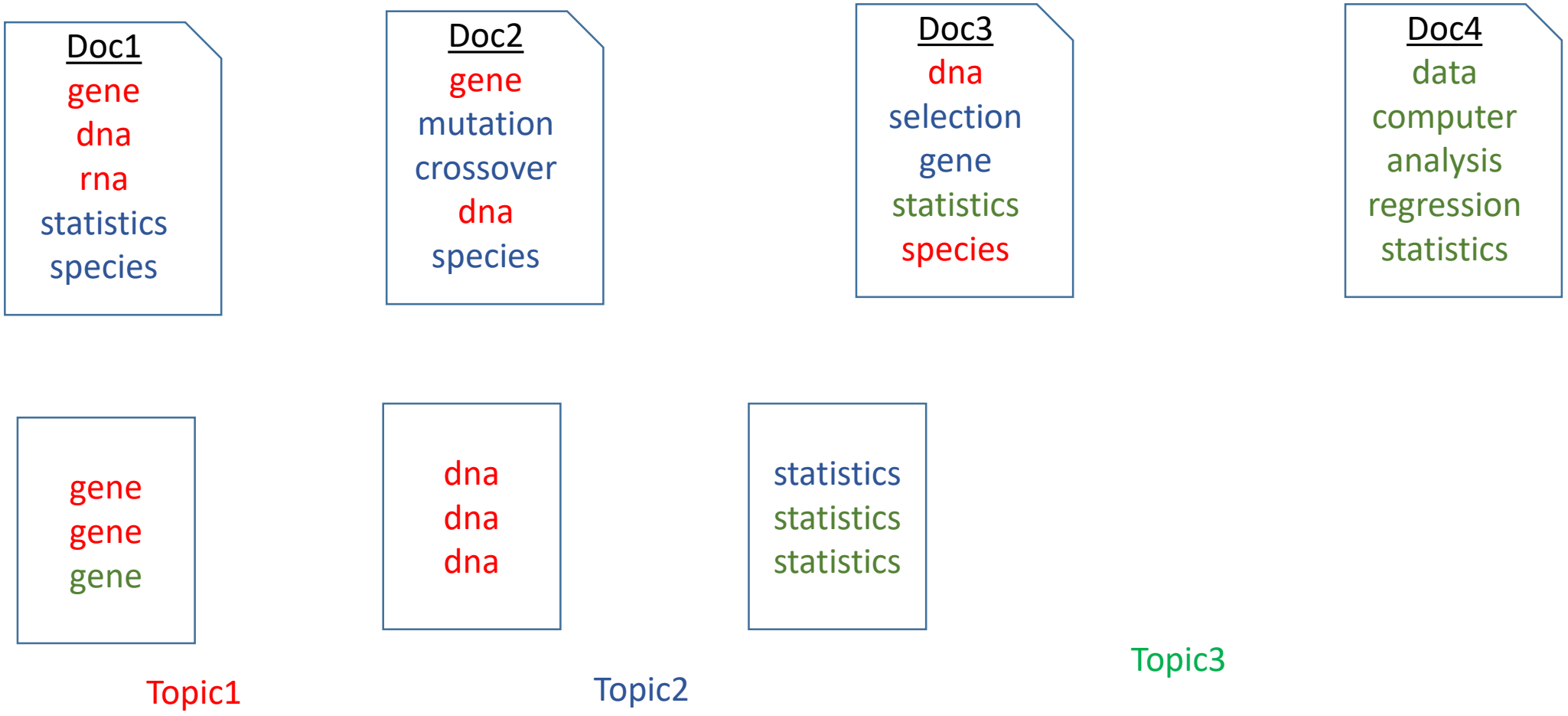
Doc4
data
computer
analysis
regression
statistics

Topic1

Topic2

Topic3

Training LDA

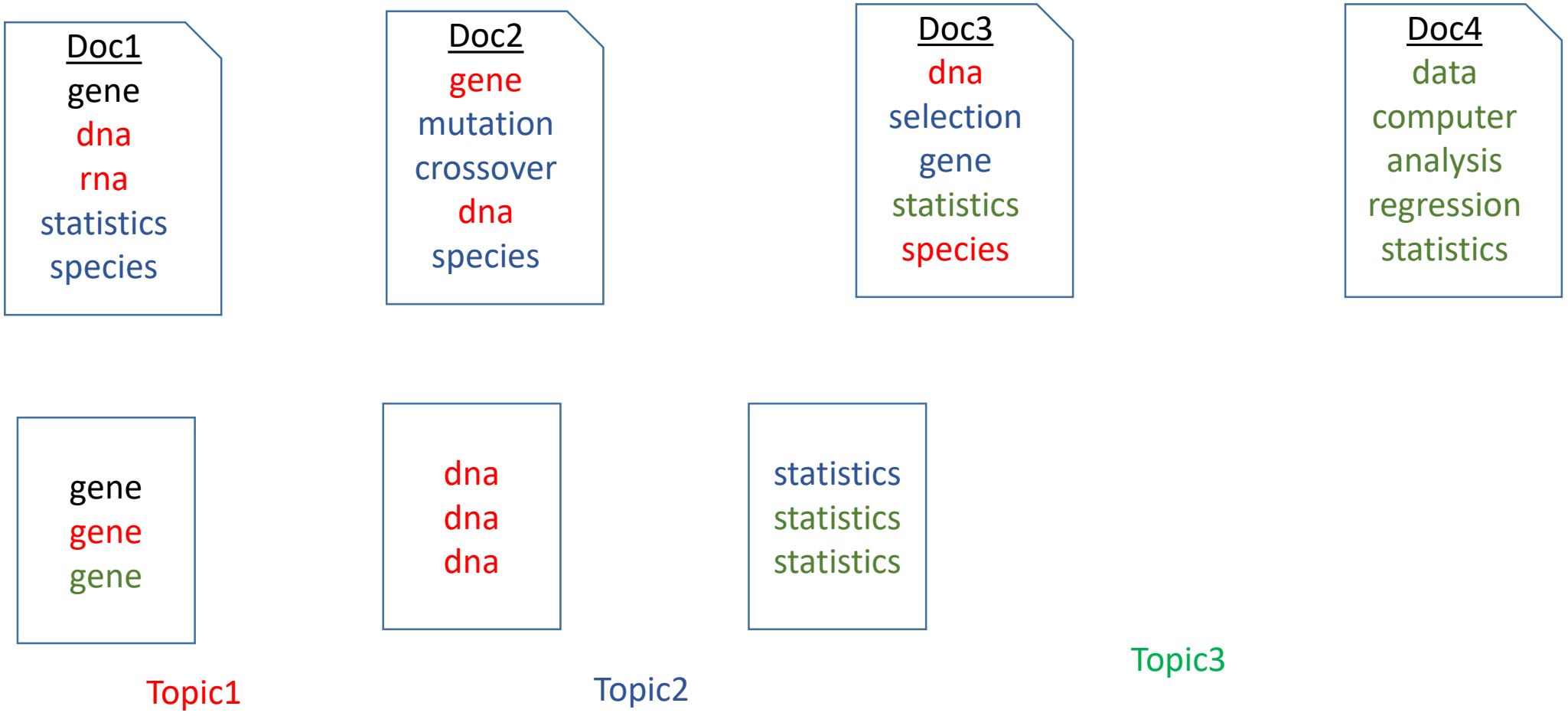


Training LDA

- Start with an initial random assignment of topics to words
- Keeping topic of all other words constant randomly change topic for one word –
- Towards the goals –
  - Individual docs should be as monochromatic as possible
  - Each word should be as monochromatic as possible

- Gibbs sampling

Training LDA



Training LDA

Doc1
gene
dna
rna
statistics
species

	Topic1	Topic2	Topic3
Doc1	2	2	0

gene
gene
gene

Word (gene)	1	0	1	
Total score	2x1	2x0	0x1	gene – Topic1

Training LDA: Smoothing

Doc1
gene
dna
rna
statistics
species

	Topic1	Topic2	Topic3
Doc1	2+α	2+α	0+α

η, α

Parameters of Dirichlet distributions

Hyperparameters

gene
gene
gene

Word (gene)	1+η	0+ η	1+ η
Total score	(2+ α)x(1+ η)	….	….

Training LDA: Smoothing

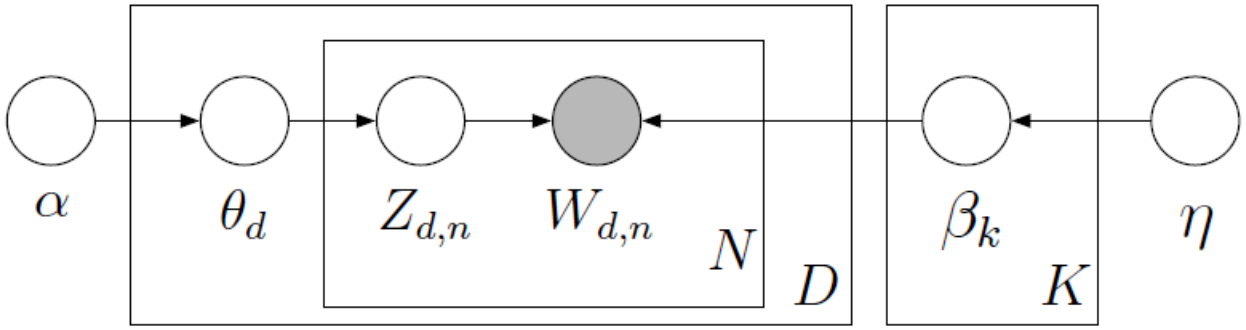
Doc1
gene
dna
rna
statistics
species

gene
gene
gene

	Topic1	Topic2	Topic3
Doc1	2+α	2+α	0+α
Word (gene)	1+η	0+ η	1+ η
Total score	(2+ α)x(1+ η)	(2+α)+(0+η)	(0+α)+(1+η)

Assign a topic to gene by generating a random integer 1-3 with probability proportional to the total scores

Training LDA: Posterior Inference



- The joint distribution of the latent variables and documents is

$$\prod_{i=1}^{K} p(\beta_i \mid \eta) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n}) \right).$$

- The posterior of the latent variables given the documents is

$$p(\beta, \theta, \mathbf{z} \mid \mathbf{w}).$$

Approximate Inference: Gibbs Sampling

- Generates a sequence of samples from the joint probability distribution of two or more random variables.

- Aim: compute posterior distribution over latent variable z
- Pre-request: we must know the conditional probability of z

  $P(z_i = j \mid z_{-i} , w_i , d_i , .)$

Gibbs Sampling based Inference

- z_i is the topic assigned to the ith token in the whole collection;
- d_i is the document containing the ith token;
- w_i is the word type of the ith token;
- $\mathbf{z}_{-i}$ is the set of topic assignments of all other tokens;
- $\cdot$ is any remaining information such as the α and η hyperparameters:

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

where $\mathbf{C}^{WT}$ and $\mathbf{C}^{DT}$ are matrices of counts (word-topic and document-topic)

Contd..

$$\beta_{ij} = \frac{C_{ij}^{WT} + \eta}{\sum_{k=1}^{W} C_{kj}^{WT} + W\eta} \quad \theta_{dj} = \frac{C_{dj}^{DT} + \alpha}{\sum_{k=1}^{T} C_{dk}^{DT} + T\alpha}$$

- Using the count matrices as before, where β_{ij} is the probability of word type i for topic j, and θ_{dj} is the proportion of topic j in document d

Gibbs Sampling for LDA

Probability that topic ***j*** is chosen for word w_i, conditioned on all other assigned topics of words in this doc and all other observed vars.

Count number of times a word token w_i was assigned to a topic ***j*** across all docs

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

unnormalized!

Count number of times a topic ***j*** was already assigned to some word token in doc d_i

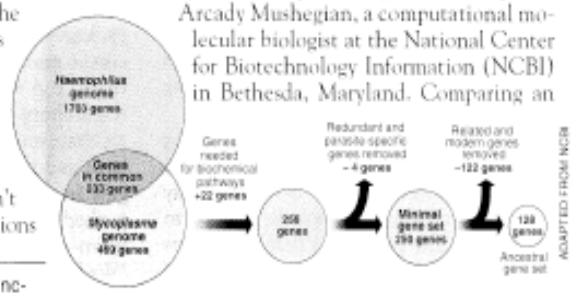
=> divide the probability of assigning topic j to word wi by the sum over all topics T

Example inference

Seeking Life's Bare (Genetic) Necessities

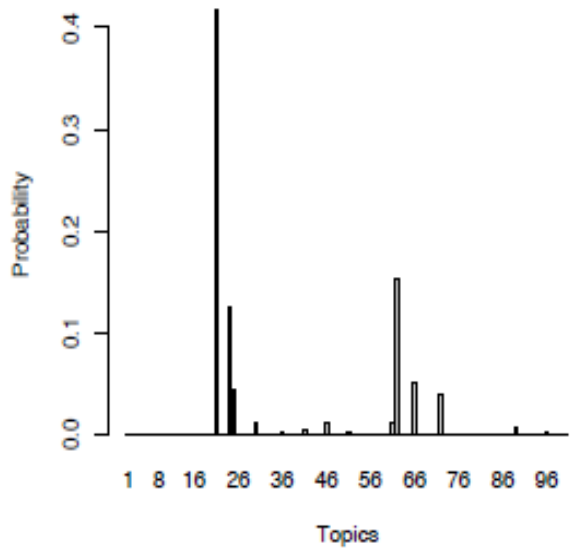
COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known 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



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

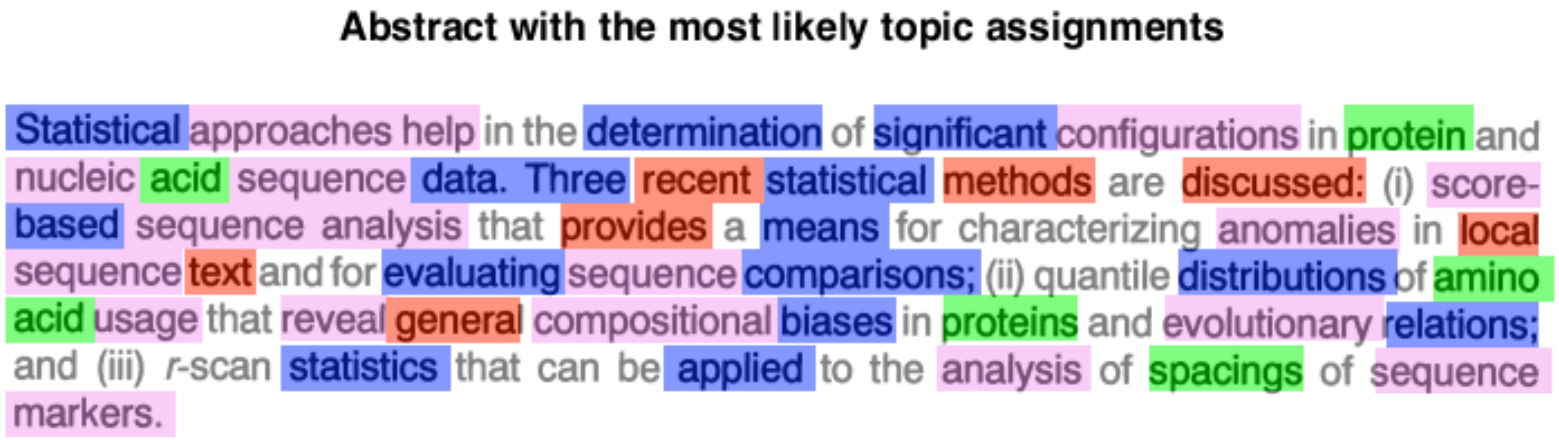
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.



Topics vs. words

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

Visualizing a document



- Use the posterior topic probabilities of each document and the posterior topic assignments to each word

Document Similarity

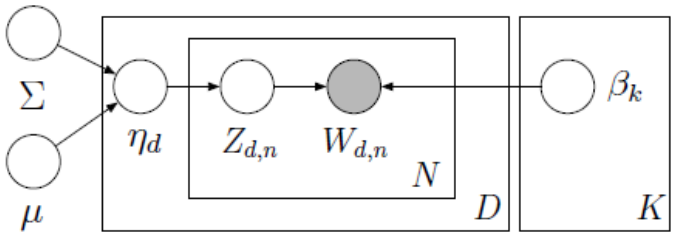
- Two documents are similar if they assign similar probabilities to topics

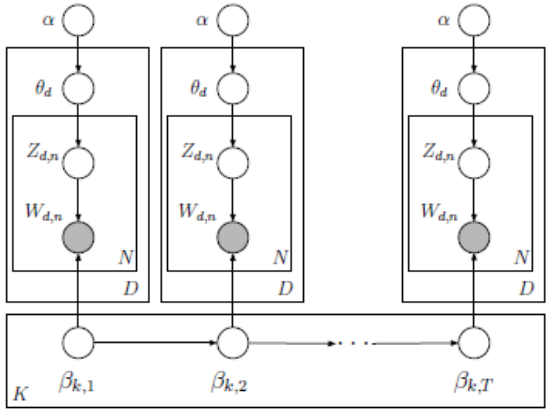
$$\text{document-similarity}_{d,f} = \sum_{k=1}^{K} \left(\sqrt{\widehat{\theta}_{d,k}} - \sqrt{\widehat{\theta}_{f,k}} \right)^2 .$$

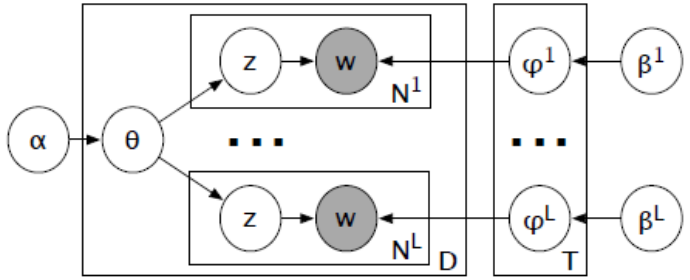
Extensions of LDA

- Correlated topic models
  - Logistic normal prior over topic assignments
- Dynamic topic models
  - Learns topic changes over time
- Polylingual topic models
  - Learns topics aligned across multiple languages

…







Evaluating Topic Models

- Consider a decomposition of a corpus into topics, i.e., $\{w_{d,n}, z_{d,n}\}$. Note that $z_{d,n}$ is a latent variable.
- For all the observations assigned to a topic, consider the variable $\{w_{d,n}, d\}$. This is the observed word and the document it appeared in.
- One measure of how well a topic model fits the LDA assumptions is to look at the **per-topic mutual information** between w and d.
- If the words from the topic are independently generated then we expect lower mutual information.
- What is “low”? To answer that, we can shuffle the words and recompute. This gives values of the MI when the words are independent.

Summary

- **The Task of Topic Modeling**
  - Topic modeling enables the **analysis of large** (possibly unannotated) **corpora**
  - Applicable to more than just bags of words
  - Extrinsic evaluations are often appropriate for these unsupervised methods
- **Constructing Models**
  - LDA is comprised of **simple building blocks** (Dirichlet, Multinomial)
  - LDA itself can act as a building block **for other models**
- **Approximate Inference**
  - Many different approaches to inference (and learning) can be applied to the same model

Research Issues

- **Model interpretation and model checking**
  Which model should I choose for which task?
- **Incorporating corpus, discourse, or linguistic structure**
  How can our knowledge of language help us build and use exploratory models of text?
- **Interfaces and "downstream" applications of topic modeling**
  What can I do with an annotated corpus? How can I incorporate latent variables into a user interface?