Data Science Bootcamp, 11th January 2017

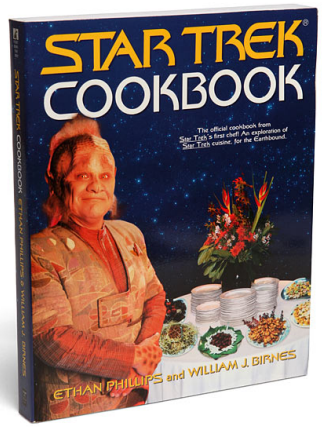
Probabilistic Modeling

Francisco J. R. Ruiz

Columbia University
Data Science Institute

Probabilistic Machine Learning

Traditional machine learning:



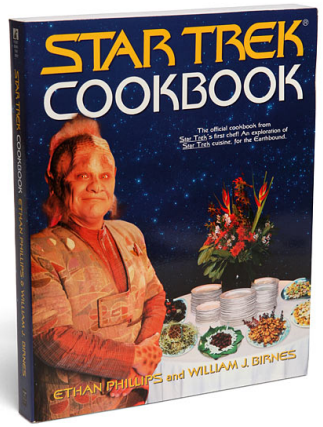
Probabilistic Machine Learning

Probabilistic machine learning:

(tailored to the problem at hand)

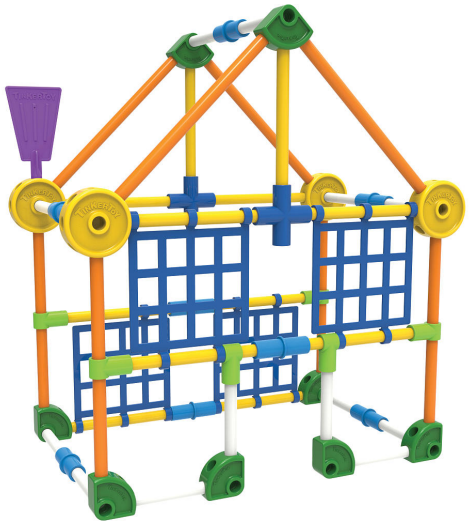
Probabilistic Machine Learning

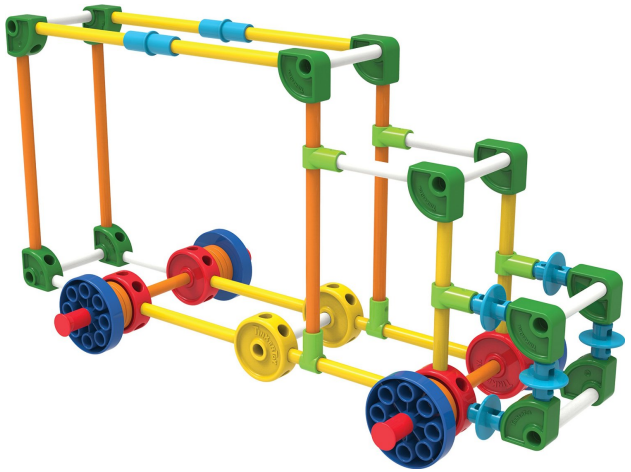
Fast and scalable; many packages of software available



Probabilistic Machine Learning

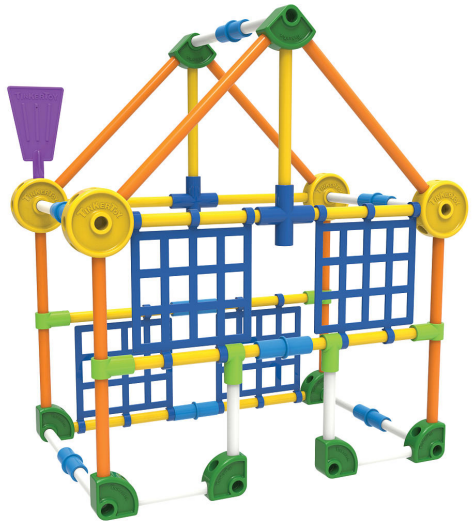
May not be fast or scalable; more challenging to implement

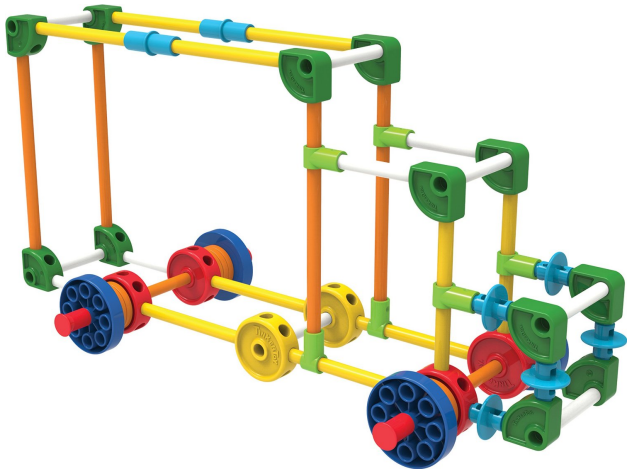




Probabilistic Machine Learning

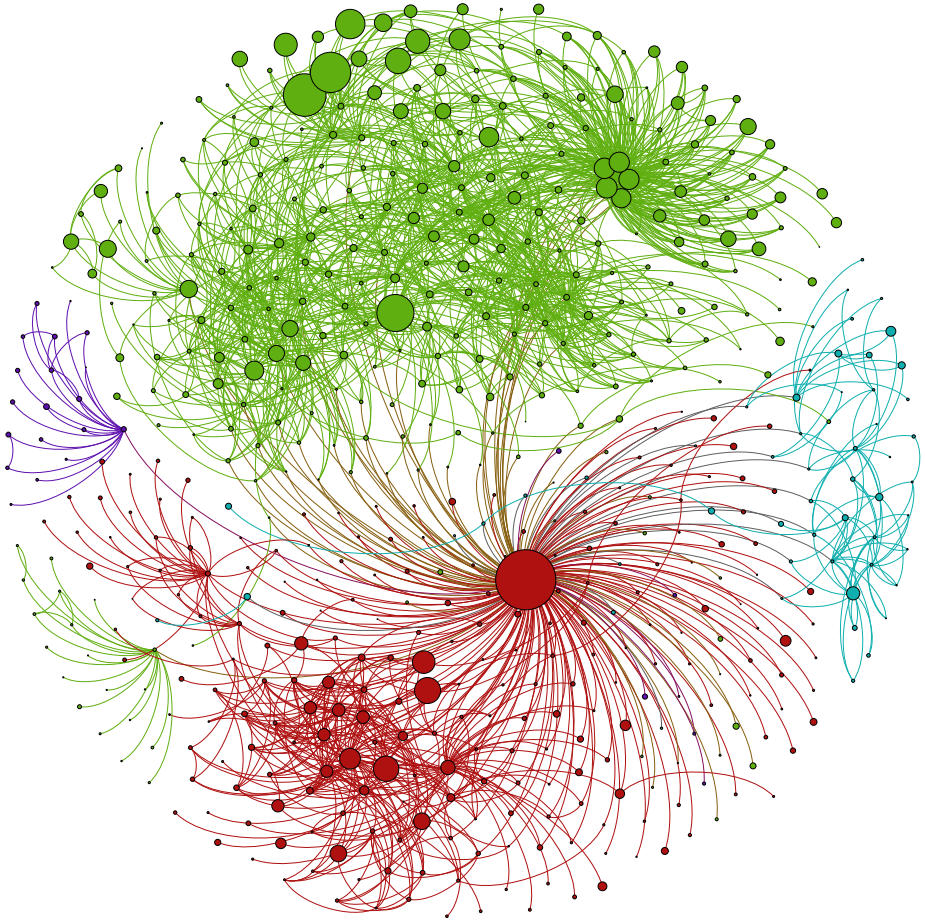
- More **flexible** in expressing dependencies
- Driven by **disciplinary** knowledge and questions
- Easily extended to **unsupervised** learning
- Focus on discovering **patterns** in the data
- Allow **exploratory** analysis
- **Large-scale** data
- Less prone to **overfitting**





Applications

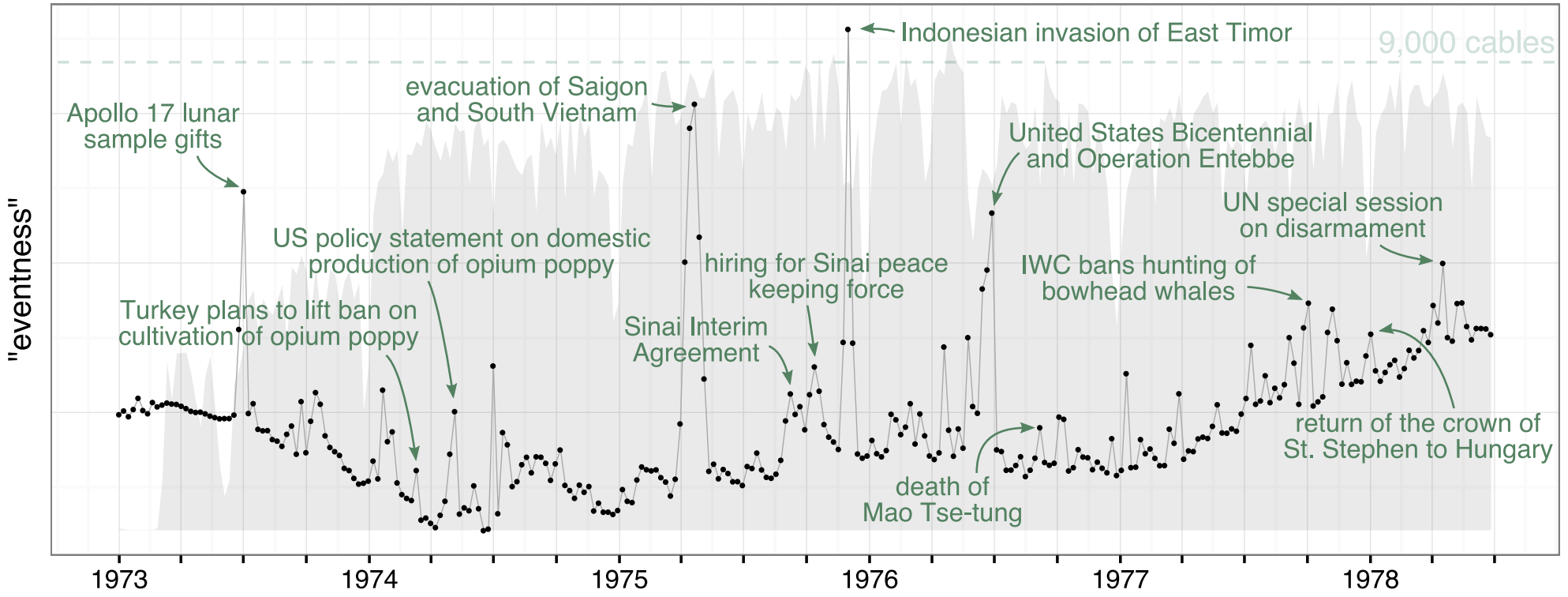
Communities discovered in a 3.7M node network of U.S. Patents



(Figure by P. Gopalan)

Applications

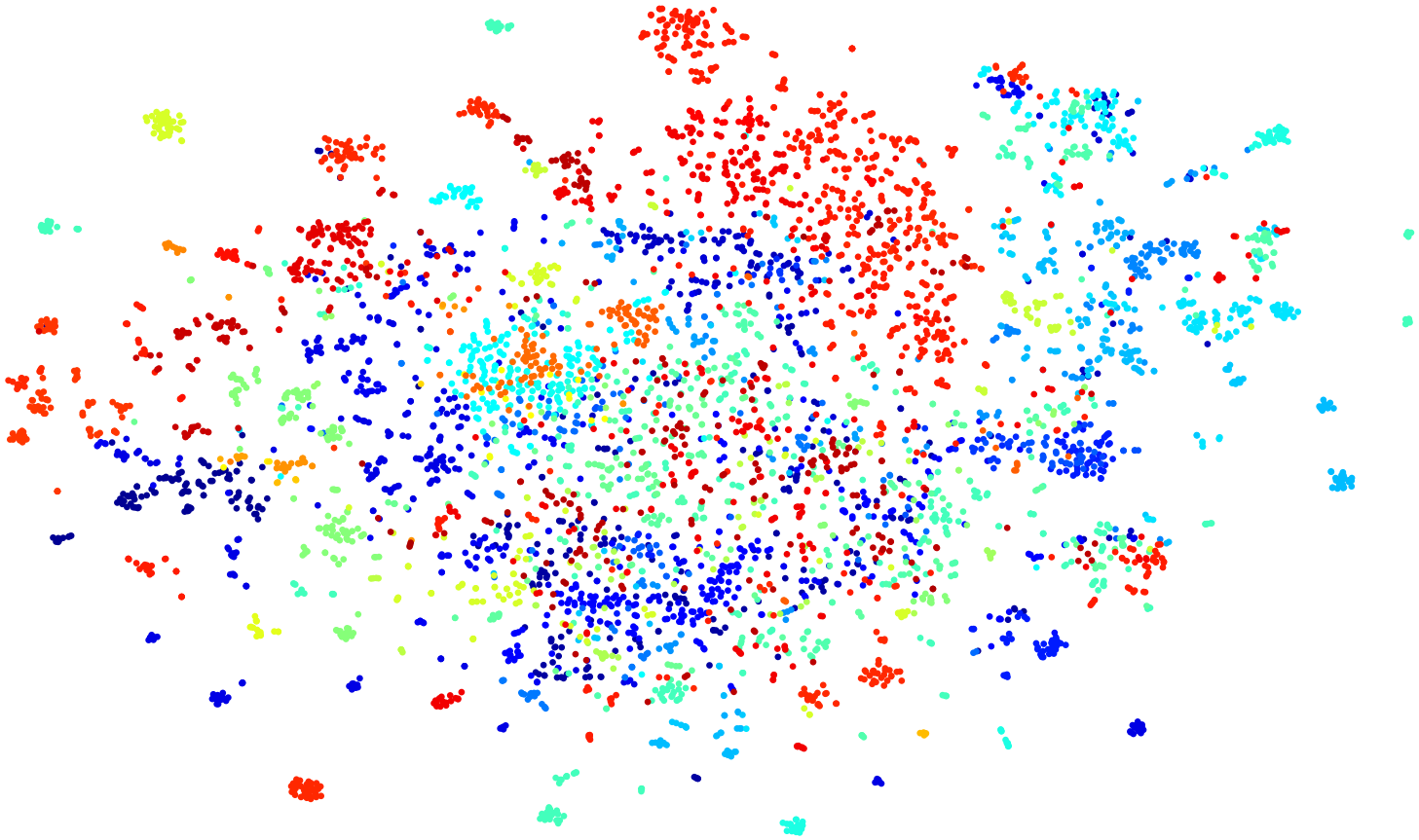
Events uncovered from 2M diplomatic cables



(Figure by A. J. B. Chaney)

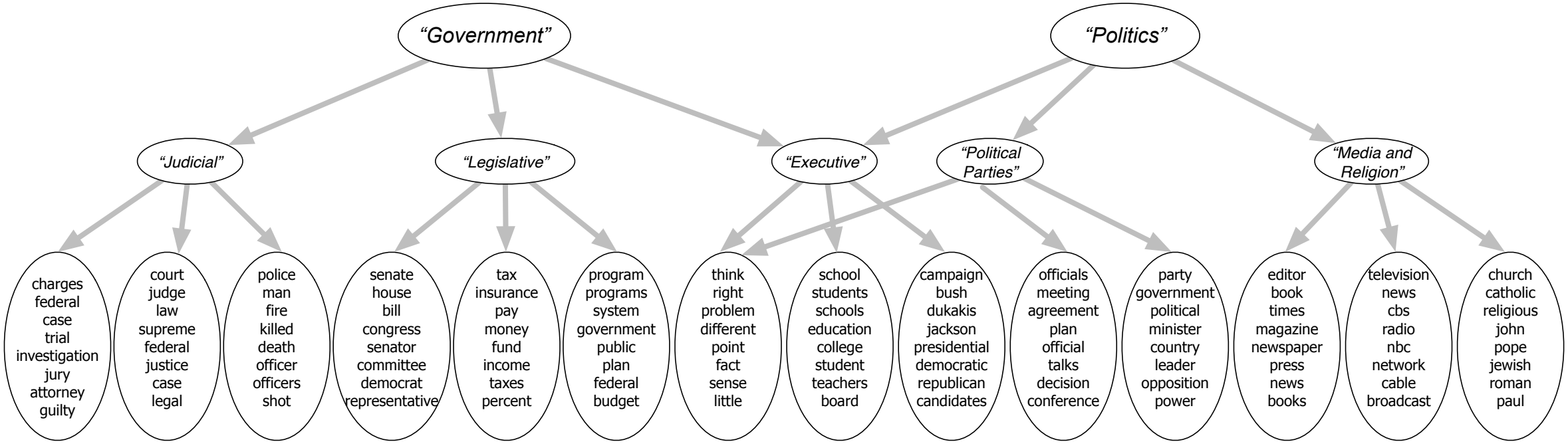
Applications

Item characteristics learned from 5.6M purchases



Applications

Hierarchical topics found in 166K articles

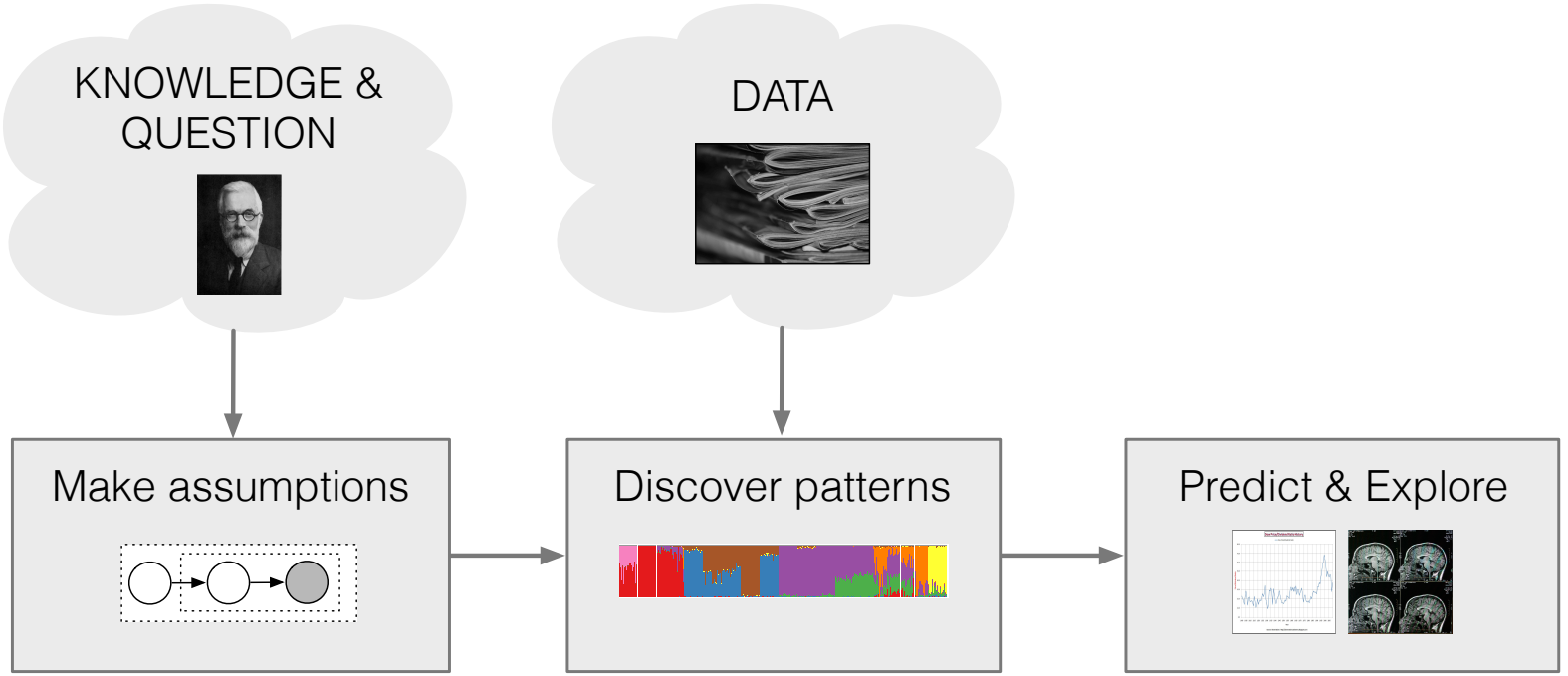


(Figure by R. Ranganath)

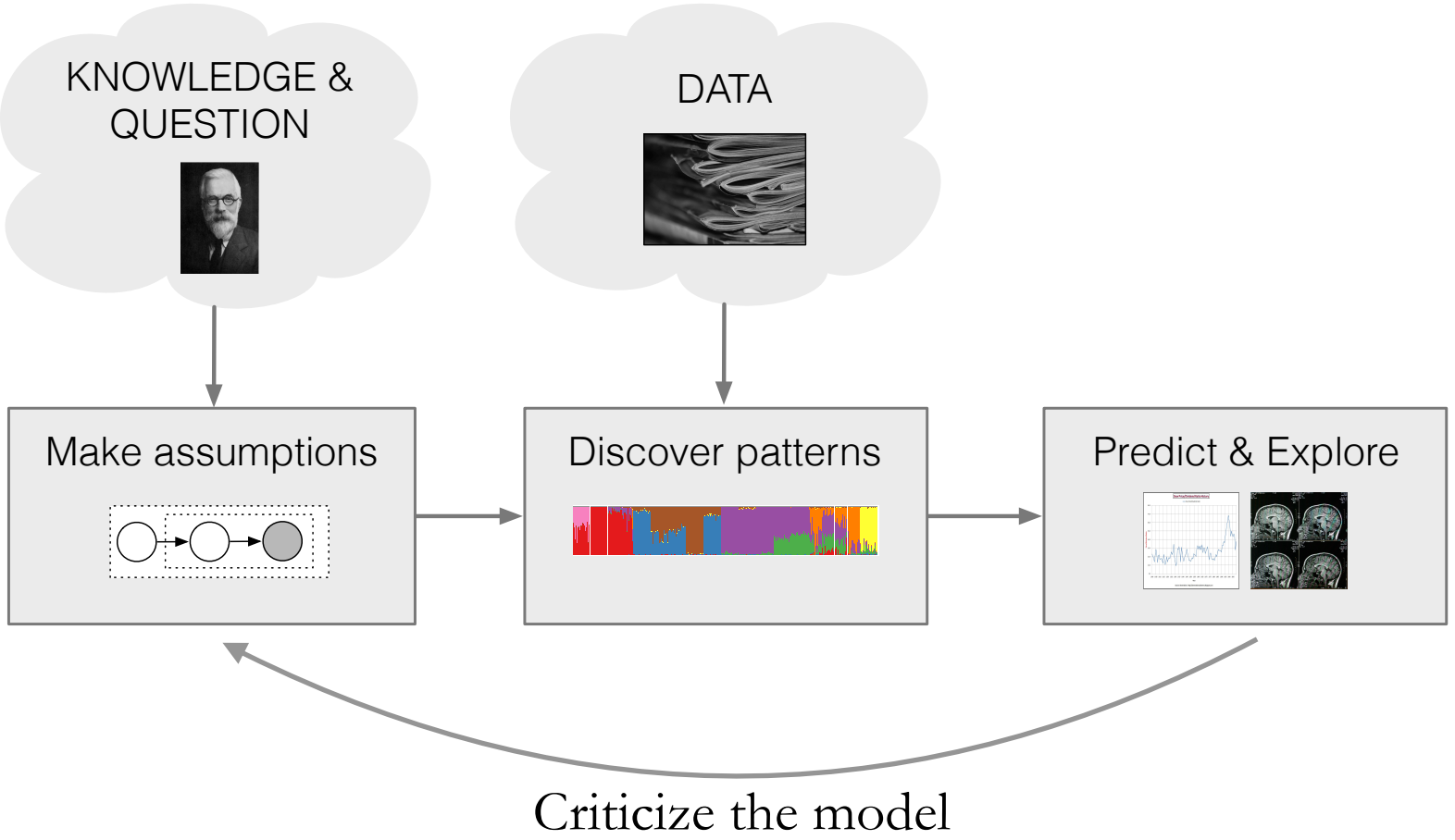
Applications

- Finance (forecasting)
- Spam identification (classification)
- Disease diagnosis (classification)
- Communication systems (error decoding)
- Recommendation systems (collaborative filtering)
- Object tracking (regression)
- …

The Probabilistic Pipeline

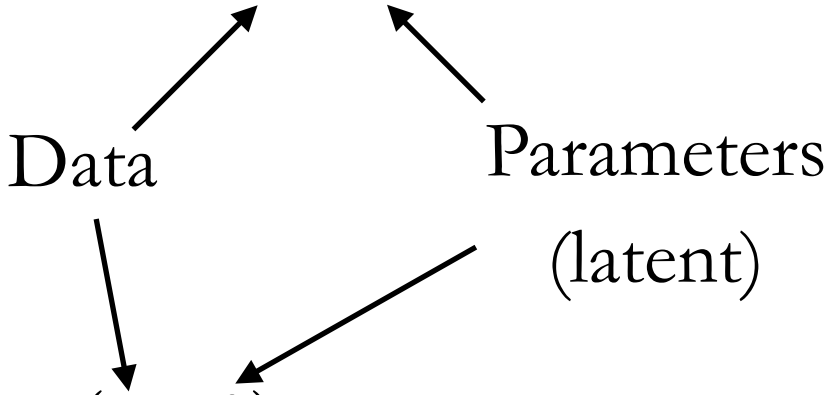


The Probabilistic Pipeline



Fundamentals

- Discriminative models: $p(y \mid \theta)$ ← *Conditional probability*



- Generative models: $p(y, \theta)$ ← *Joint probability*

Probabilistic Models

- Joint probability:

$$p(y, \theta) = \underbrace{p(y \mid \theta)}_{\text{Likelihood}} \times \underbrace{p(\theta)}_{\text{Prior}}$$

(by the rules of probability)

Prior Probability Distribution

- Intuition behind the **prior**:
  - Probability of the parameters before observing any data
  - Encode prior beliefs as a probability distribution

$$p(y, \theta) = \underbrace{p(y \mid \theta)}_{\text{Likelihood}} \times \underbrace{p(\theta)}_{\text{Prior}}$$

Prior Probability Distribution

- How to determine the **prior**:
  - From past information
  - Assessment from experts in the field
  - Uninformative prior
  - Mathematical convenience
- The prior can reflect a complex hidden structure

Likelihood

- **Likelihood**:
  - Probability of data conditioned on latent parameters
  - We say "likelihood of the *parameters*" although it is the probability distribution of the *data*

$$p(y, \theta) = \underbrace{p(y \mid \theta)}_{\text{Likelihood}} \times \underbrace{p(\theta)}_{\text{Prior}}$$

The Generative Model

- The underlying (imaginary) generative process:

  1. The parameters were generated from $p(\theta)$
  2. The data was generated from $p(y \mid \theta)$

Posterior Distribution

- **Assumption:** The dataset was generated following this process

- **Goal:** Infer the parameters that are more likely to have generated the data

Posterior Distribution

- Posterior distribution:

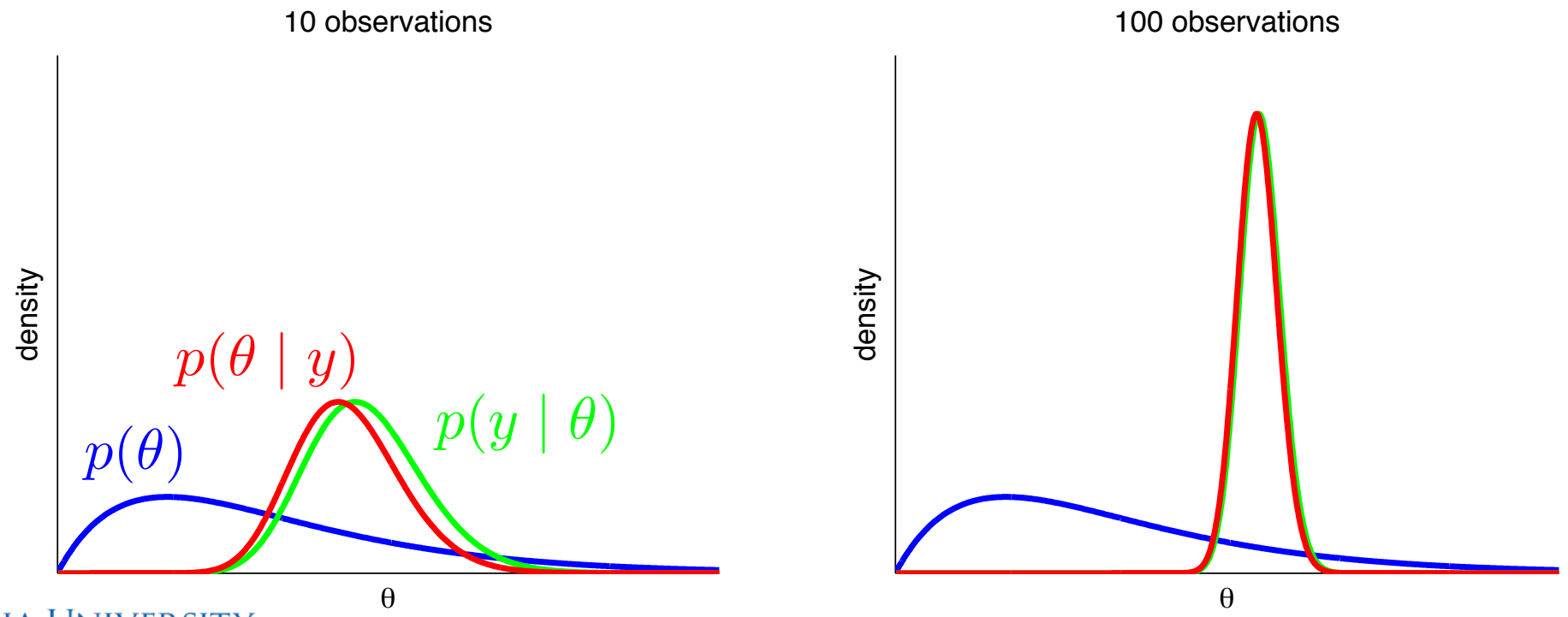
$$p(\theta \mid y) = \frac{p(y \mid \theta) \times p(\theta)}{p(y)}$$

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$

(by Bayes rule)

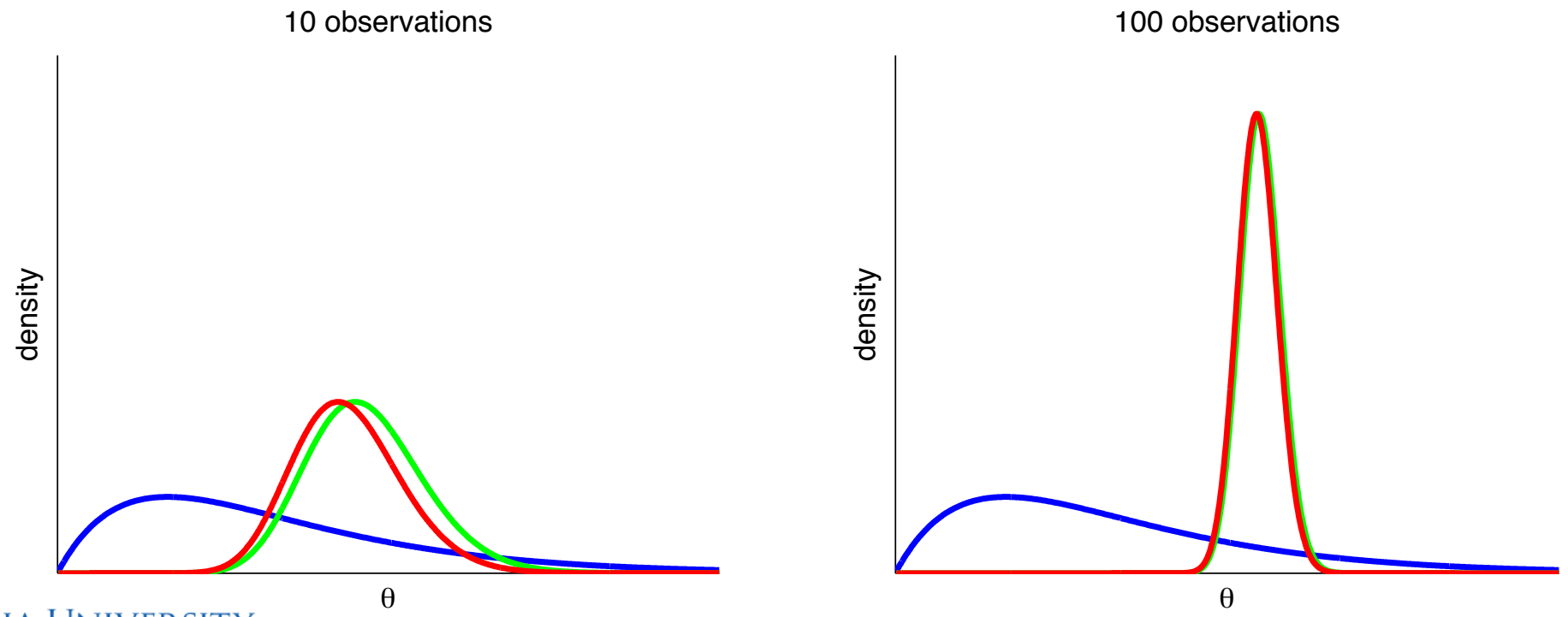
Posterior Distribution

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$



Posterior Distribution

- The **posterior** tells us how uncertain we are after we have observed the data



Posterior Distribution

- The **posterior** allows us to explore the latent patterns

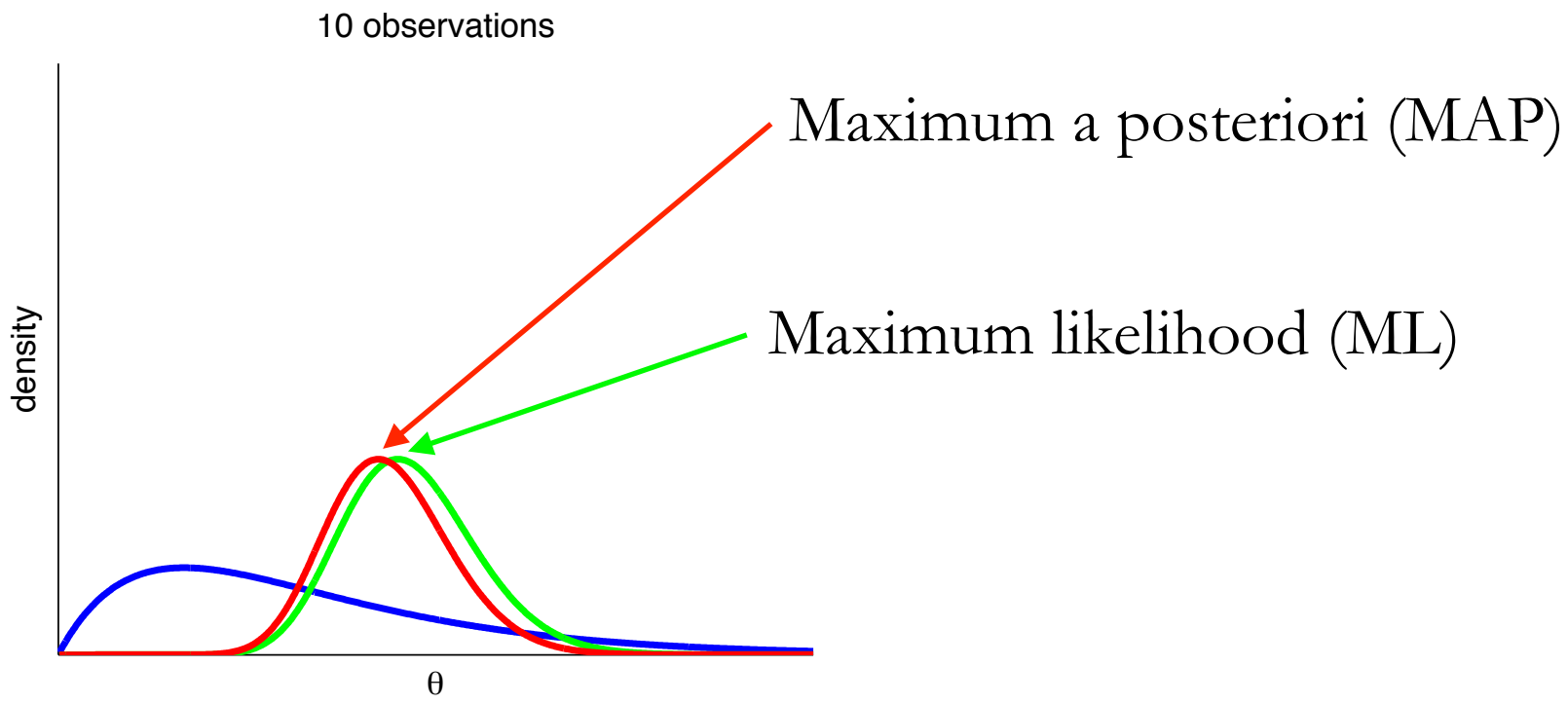
$$p(\theta \mid y)$$

- The **posterior** allows us to form predictions

$$p(y^{\text{new}} \mid y) = \int p(y^{\text{new}} \mid \theta) p(\theta \mid y) d\theta$$

ML / MAP

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$



Inference

- The posterior is the central object in Bayesian inference

- Two families of methods:
  - Sample from the posterior (MCMC)
  - Approximate the posterior (Variational)

Inference

- Inference is the main computational bottleneck

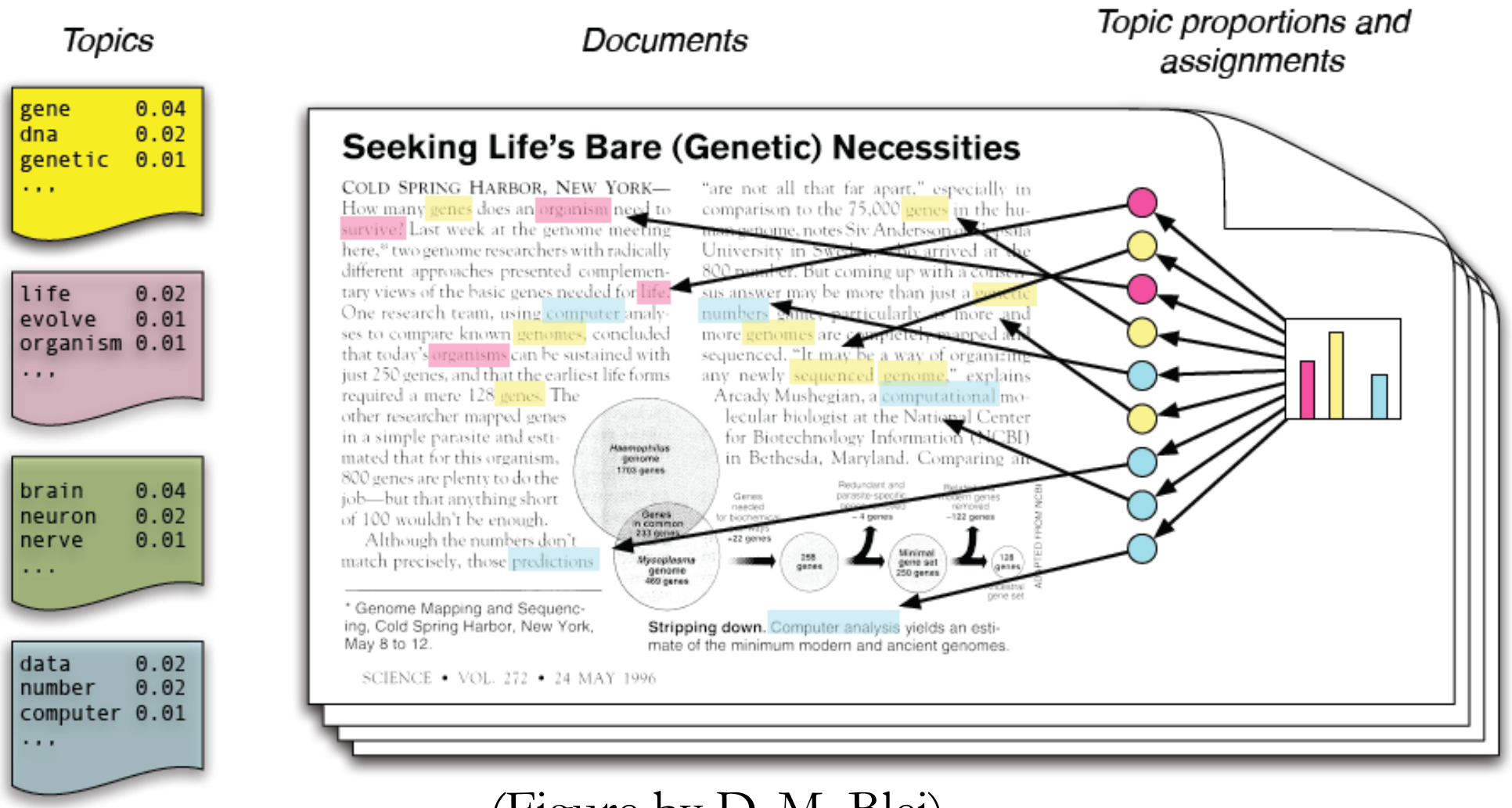
- Researchers are working to speed it up

Example: LDA

Example: LDA

- LDA is a probabilistic model of text
  - Each **topic** is a distribution over the vocabulary words
  - Each **document** is a mixture of topics
  - Each **word** is drawn from one of those topics

Example: LDA



(Figure by D. M. Blei)

Example: LDA

- Generative process in math:
  - Generate each topic,

$$p(\beta_k) = \text{Dirichlet}(a)$$

  - Generate the topic proportions for each document,

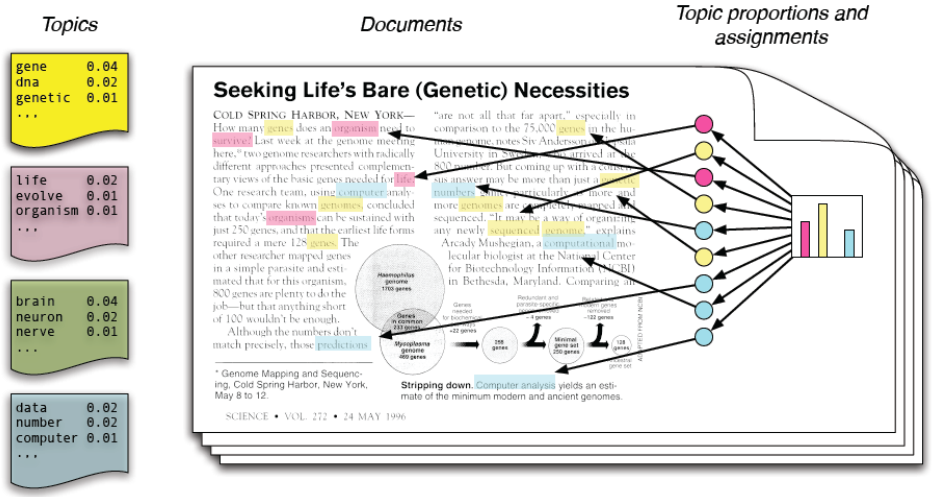
$$p(\theta_d) = \text{Dirichlet}(b)$$

  - Generate a topic indicator for each word,

$$p(z_{nd} \mid \theta_d) = \text{Categorical}(\theta_d)$$

  - Generate the actual words,

$$p(y_{nd} \mid z_{nd}, \{\beta_k\}) = \text{Categorical}(\beta_{z_{nd}})$$



Example: LDA

- Generative process in math:
  - Generate each topic,

$$p(\beta_k) = \text{Dirichlet}(a)$$

  - Generate the topic proportions for each document,

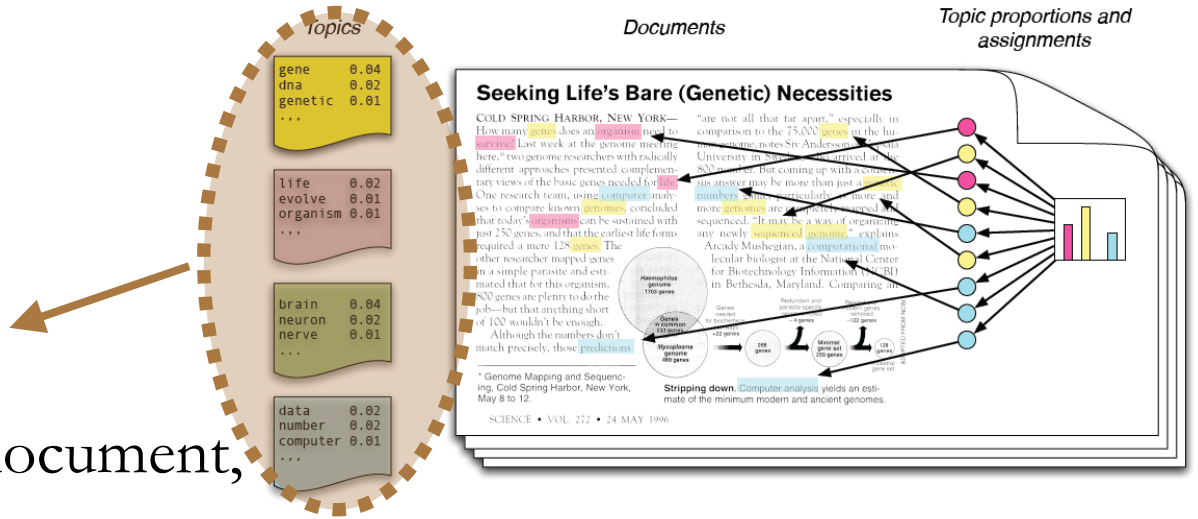
$$p(\theta_d) = \text{Dirichlet}(b)$$

  - Generate a topic indicator for each word,

$$p(z_{nd} \mid \theta_d) = \text{Categorical}(\theta_d)$$

  - Generate the actual words,

$$p(y_{nd} \mid z_{nd}, \{\beta_k\}) = \text{Categorical}(\beta_{z_{nd}})$$



Example: LDA

- Generative process in math:
  - Generate each topic,

$$p(\beta_k) = \text{Dirichlet}(a)$$

  - Generate the topic proportions for each document,

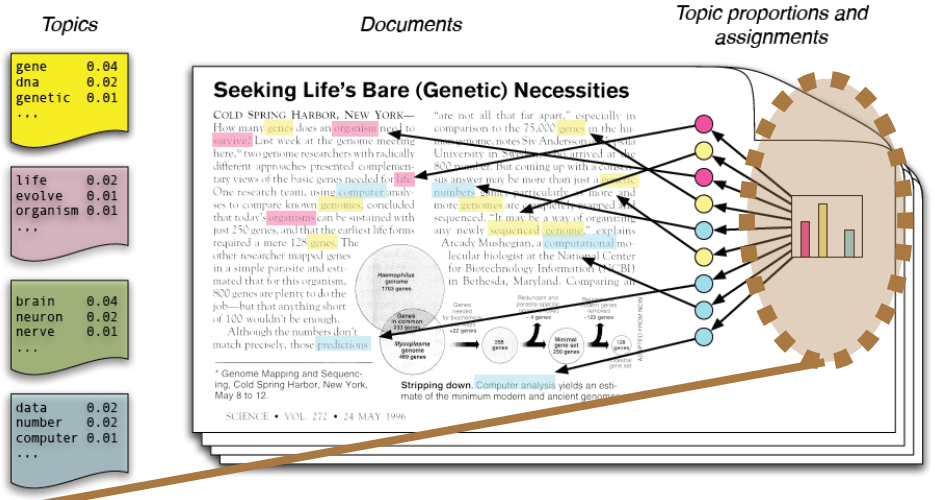
$$p(\theta_d) = \text{Dirichlet}(b)$$

  - Generate a topic indicator for each word,

$$p(z_{nd} \mid \theta_d) = \text{Categorical}(\theta_d)$$

  - Generate the actual words,

$$p(y_{nd} \mid z_{nd}, \{\beta_k\}) = \text{Categorical}(\beta_{z_{nd}})$$



Example: LDA

- Generative process in math:
  - Generate each topic,

$$p(\beta_k) = \text{Dirichlet}(a)$$

  - Generate the topic proportions for each document,

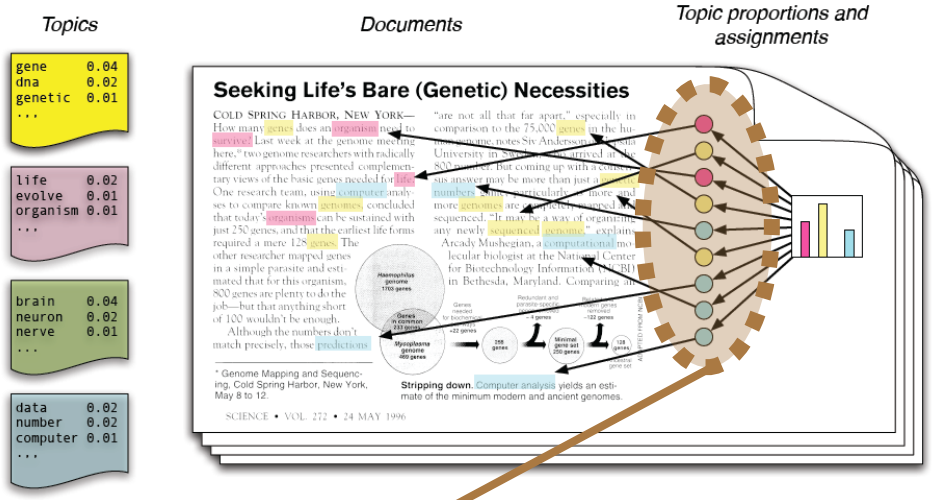
$$p(\theta_d) = \text{Dirichlet}(b)$$

  - Generate a topic indicator for each word,

$$p(z_{nd} \mid \theta_d) = \text{Categorical}(\theta_d)$$

  - Generate the actual words,

$$p(y_{nd} \mid z_{nd}, \{\beta_k\}) = \text{Categorical}(\beta_{z_{nd}})$$



Example: LDA

- Generative process in math:
  - Generate each topic,

$$p(\beta_k) = \text{Dirichlet}(a)$$

  - Generate the topic proportions for each document,

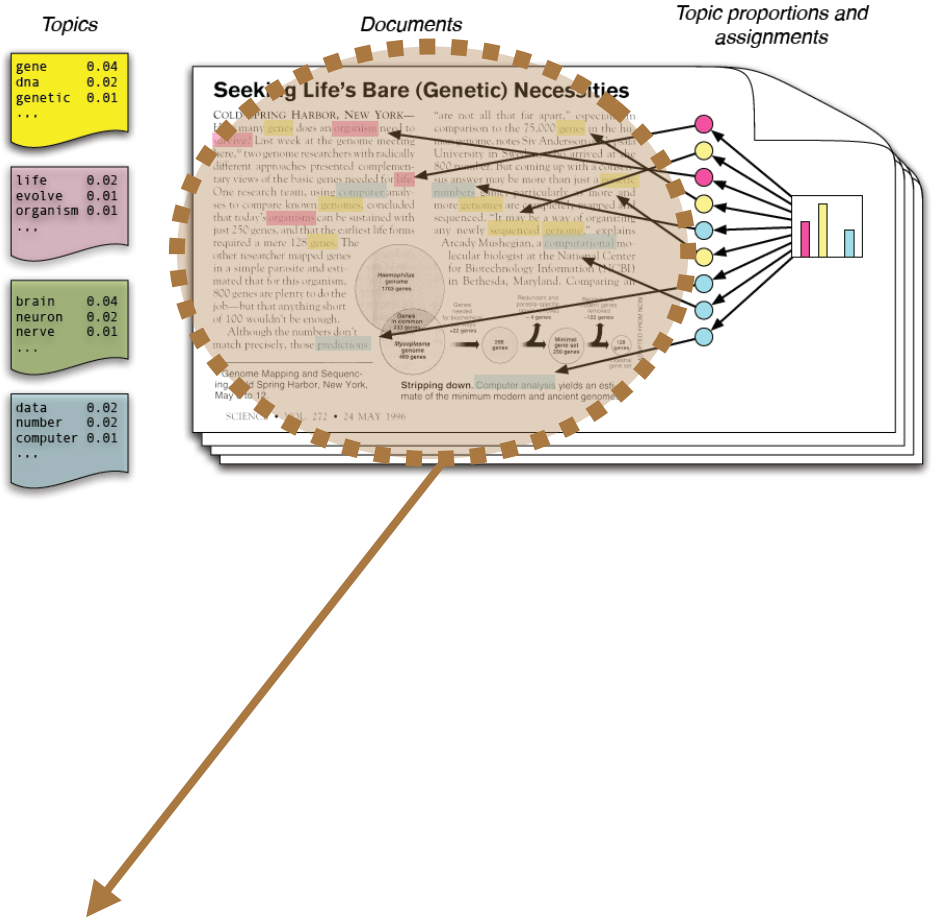
$$p(\theta_d) = \text{Dirichlet}(b)$$

  - Generate a topic indicator for each word,

$$p(z_{nd} \mid \theta_d) = \text{Categorical}(\theta_d)$$

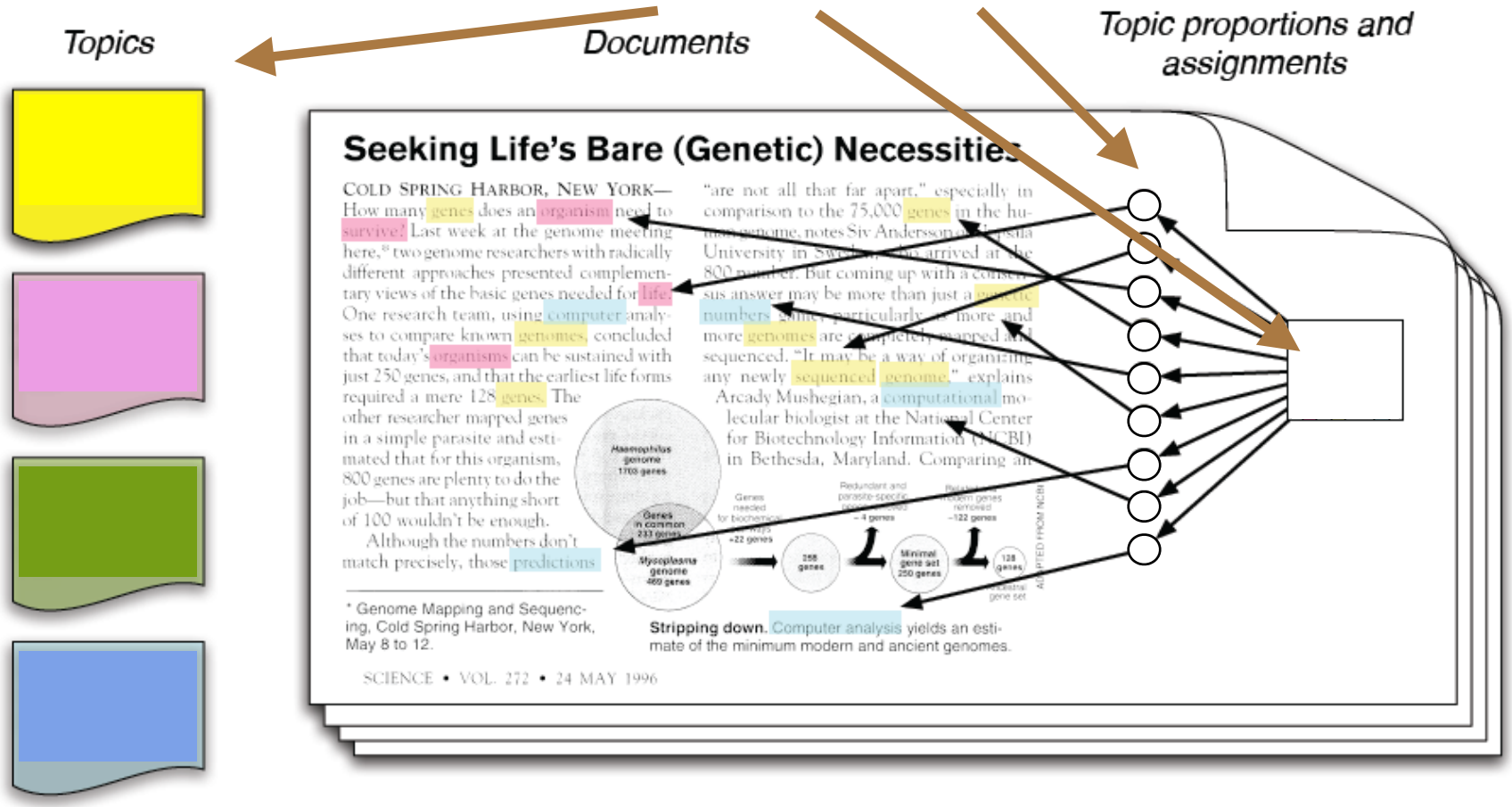
  - Generate the actual words,

$$p(y_{nd} \mid z_{nd}, \{\beta_k\}) = \text{Categorical}(\beta_{z_{nd}})$$



Example: LDA

- Posterior distribution: $p(\{\beta_k\}, \{\theta_d\}, \{z_{nd}\} \mid \{y_{nd}\})$



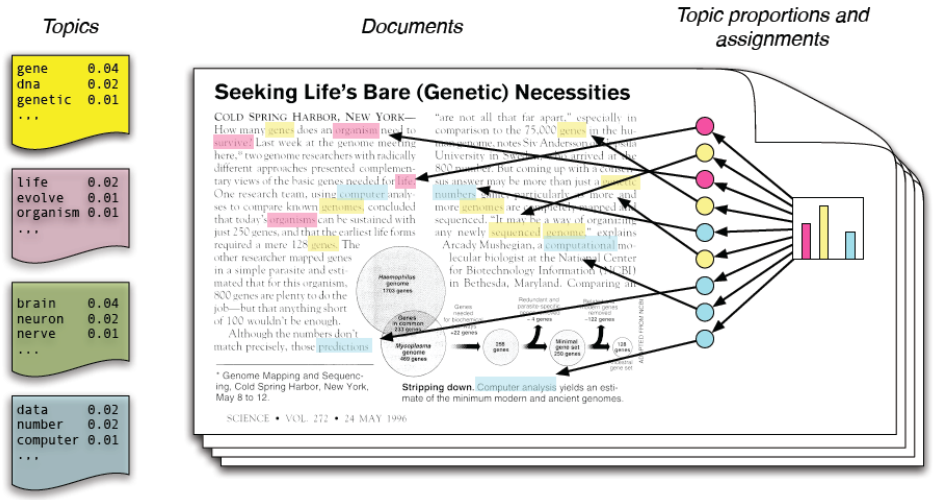
Example: LDA

- Demo by A. J. B. Chaney (*Wikipedia*):

  http://www.princeton.edu/~achaney/tmve/wiki100k/browse/topic-presence.html

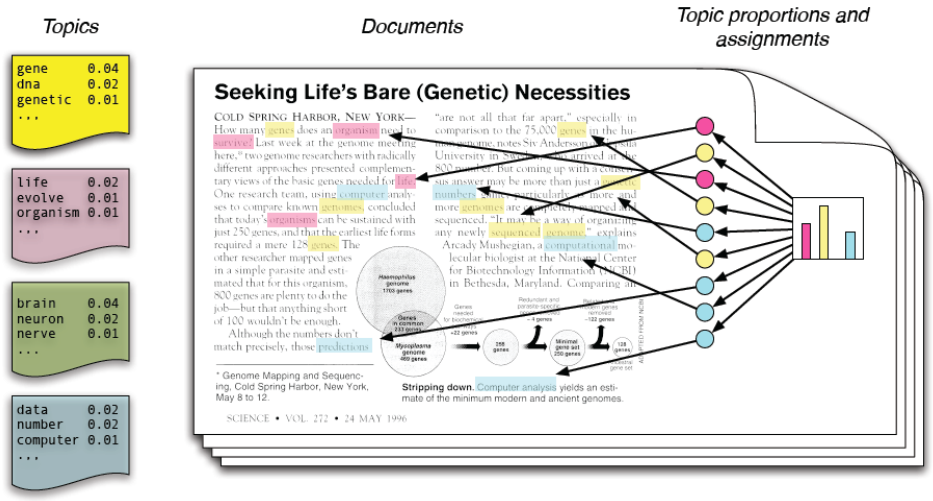
Today's Session 1

- Fit LDA to a collection of *New York Times* articles
  - Python package: gensim
  - (Inference: Variational EM)
  - Predict the topic proportions of unseen documents
  - Explore the results



Today's Session 1

- Structure:
  1. Tutorial (toy data; focus on preprocessing steps)
  2. Assigned task (NYT data; full process)



COLUMBIA UNIVERSITY
Data Science Institute

Today's Session 2

- Find ML/MAP for (probabilistic) logistic regression
  - Binary classification
  - Breast cancer data
  - Gradient ascent to maximize the (log-)likelihood
  - Gradient ascent to maximize the (log-)posterior

Today's Session 2

- Objective function and expression of the gradients are given in the IPython notebook
- Tip: Same algorithm we implemented on Tuesday
- Compare ML and MAP results

COLUMBIA UNIVERSITY
Data Science Institute