Announcement

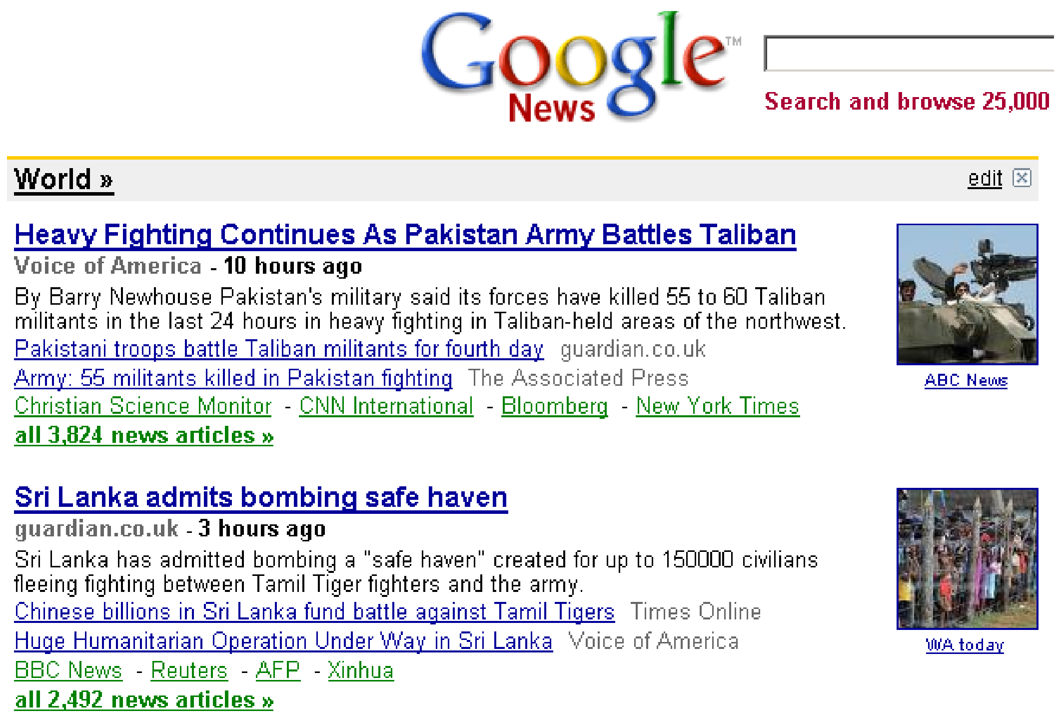
- Homework 1
  - Available in Blackboard -> Homework
  - Due: Feb. 26, 11:59pm

Text Clustering

INLP Ch 5

Text Clustering

- Application
  - News aggregation website



Text Clustering

- Application
  - Group customer reviews

Text Clustering: Definition

- Input:
  - A set of document $\{d_1, d_2, \dots, d_n\}$
- Output:
  - A cluster assignment
    - $C_1 = \{d_1, d_3, \dots\}$
    - $C_2 = \{d_2, d_6, \dots\}$
    - $C_3 = \{d_4, \dots\}$
    - …

A Common Method

- Represent text with feature vectors
- Apply any clustering algorithm
  - K-means
  - Hierarchical agglomerative clustering
  - Expectation-maximization with mixture of Gaussian
  - …

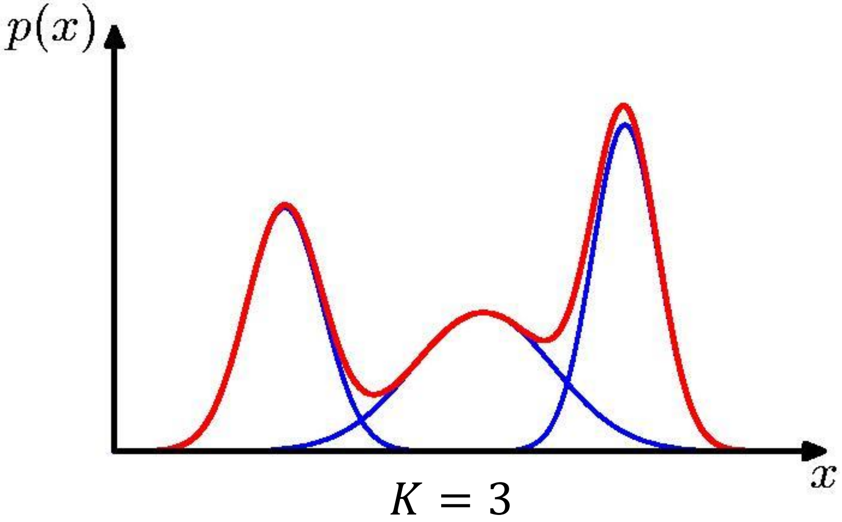
Requires a distance measure between vectors, e.g., L2

Mixture of Gaussian (MoG)

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \underbrace{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}_{\text{Component}}$$

Mixing coefficient

$$\forall k : \pi_k \geqslant 0 \qquad \sum_{k=1}^{K} \pi_k = 1$$

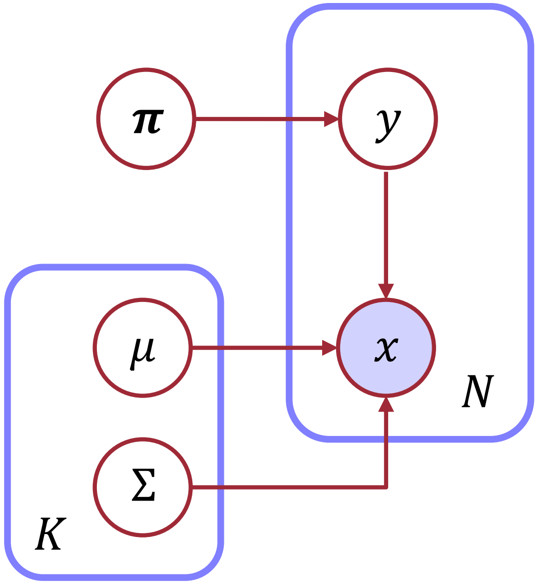


Mixture of Gaussian (MoG)

- $P(Y)$: Distribution over k components (clusters)
- $P(X|Y)$: Each component generates data from a **multivariate Gaussian** with mean μ_i and covariance matrix Σ_i

Each data point is sampled from a ***generative process*:**

1. Choose component $y = i$ with probability π_i
2. Generate data point from $\mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)$

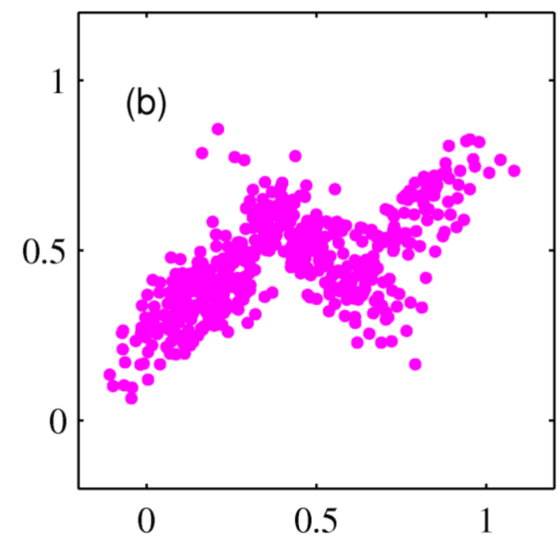


Unsupervised learning for MoG

- In clustering, we don't know the labels Y!
- Maximize marginal likelihood:

$$\prod_j P(\mathbf{x}_j) = \prod_j \sum_i P(y_j = i, \mathbf{x}_j) = \prod_j \sum_i \pi_i N(\mathbf{x}_j | \mu_i, \Sigma_i)$$

- How do we optimize it?
  - No closed form solution



Expectation-Maximization (EM)

- Pick K random cluster models (Gaussians)
- Alternate:
  - [E step] Assign data instances proportionately to different models
  - [M step] Revise each cluster model based on its (proportionately) assigned points
- Stop when no significant change (of marginal likelihood)

- EM = maximizing marginal likelihood by coordinate ascent

E-step

- [E step] Assign data instances proportionately to different models
  - Compute label distribution of each data point

$$P\left(y_j = i \,\middle|\, \boldsymbol{x}_j, \theta^{(t)}\right) \propto \pi_i{}^{(t)} N\left(\boldsymbol{x}_j \,\middle|\, \mu_i{}^{(t)}, \Sigma_i{}^{(t)}\right)$$

Just evaluate a Gaussian at $\boldsymbol{x}_j$

M-step

- [E step] Assign data instances proportionately to different models
  - Compute label distribution of each data point

$$\mathrm{P}\left(y_j = i \,\middle|\, \boldsymbol{x}_j, \theta^{(t)}\right) \propto {\pi_i}^{(t)} N\left(\boldsymbol{x}_j \,\middle|\, {\mu_i}^{(t)}, {\Sigma_i}^{(t)}\right)$$

- [M step] Revise each cluster model based on its (proportionately) assigned points
  - Compute weighted MLE of parameters given label distributions

$${\mu_i}^{(t+1)} = \frac{\sum_j \mathrm{P}\left(y_j = i \,\middle|\, \boldsymbol{x}_j, \theta^{(t)}\right) \boldsymbol{x}_j}{\sum_{j'} \mathrm{P}\left(y_{j'} = i \,\middle|\, \boldsymbol{x}_{j'}, \theta^{(t)}\right)}$$

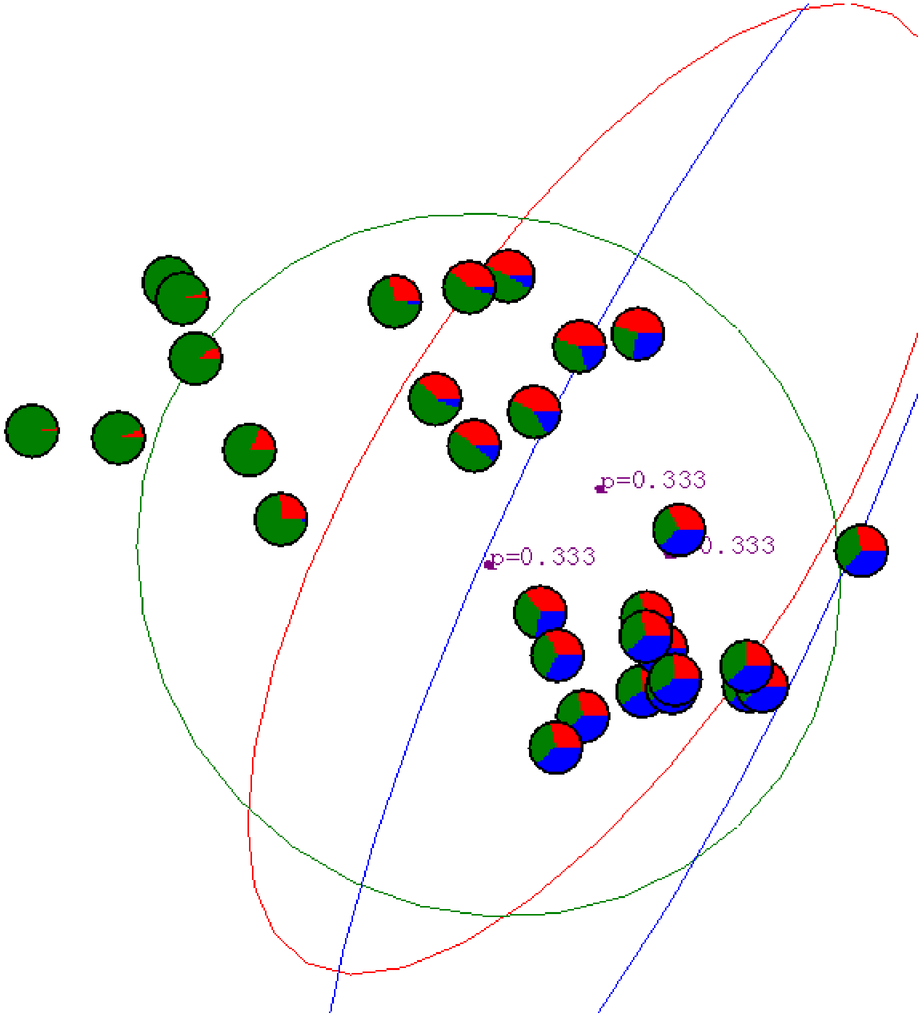
$${\pi_i}^{(t+1)} = \frac{\sum_j \mathrm{P}\left(y_j = i \,\middle|\, \boldsymbol{x}_j, \theta^{(t)}\right)}{m}$$

m = # training examples

$${\Sigma_i}^{(t+1)} = \frac{\sum_j \mathrm{P}\left(y_j = i \,\middle|\, \boldsymbol{x}_j, \theta^{(t)}\right) \left[\boldsymbol{x}_j - {\mu_i}^{(t+1)}\right]\left[\boldsymbol{x}_j - {\mu_i}^{(t+1)}\right]^T}{\sum_{j'} \mathrm{P}\left(y_{j'} = i \,\middle|\, \boldsymbol{x}_{j'}, \theta^{(t)}\right)}$$

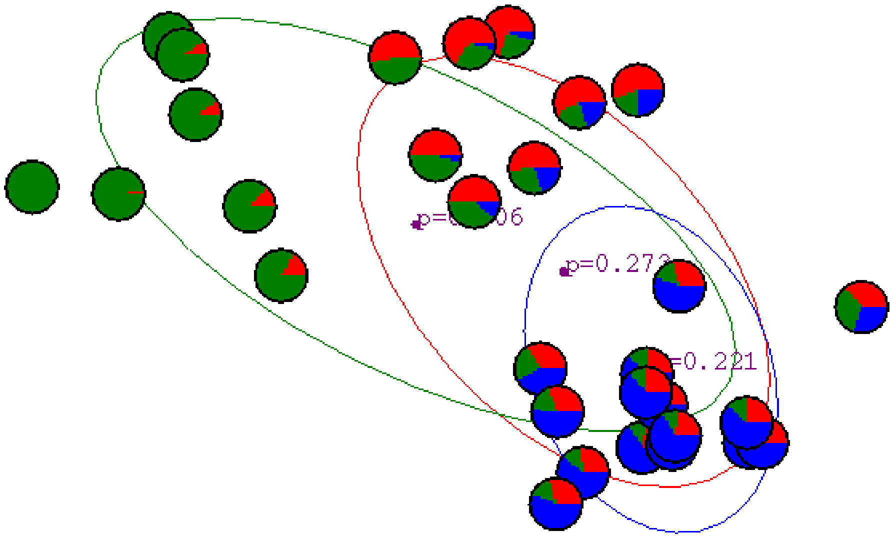
Gaussian Mixture Example

- Start



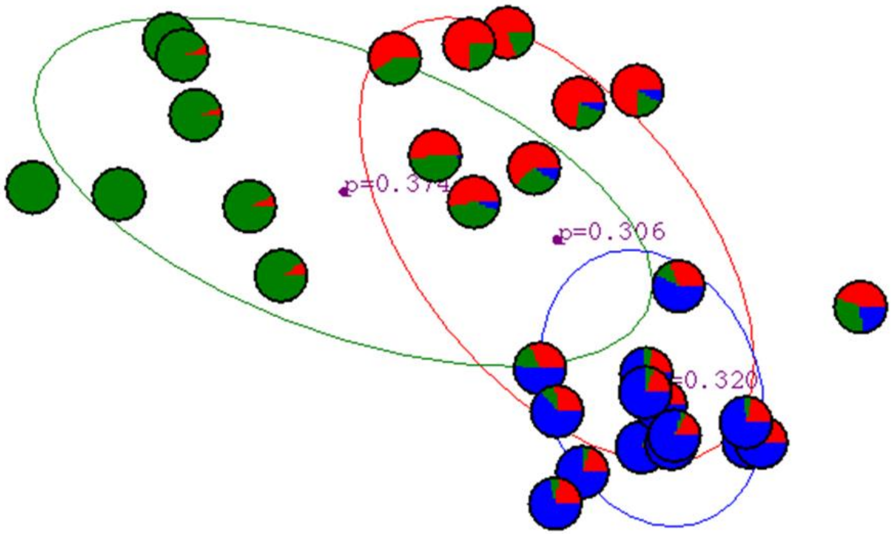
Gaussian Mixture Example

- 1st iteration



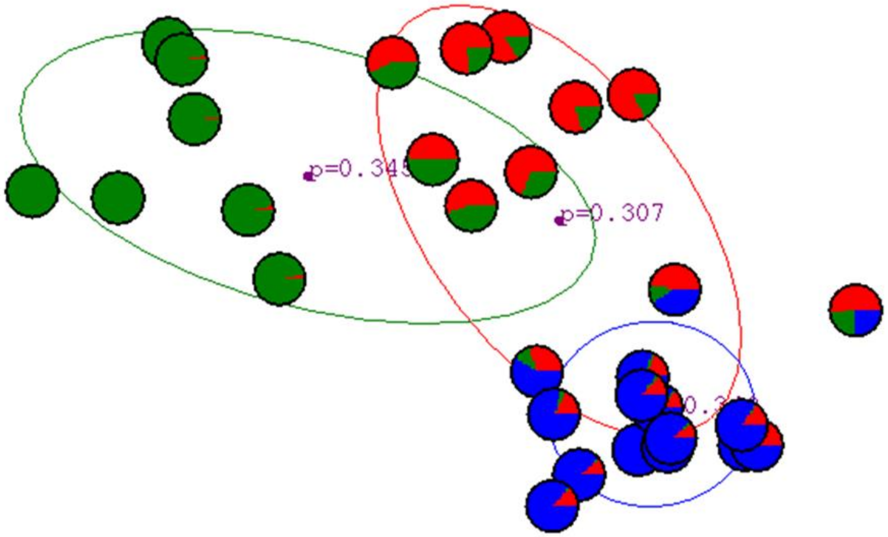
Gaussian Mixture Example

- 2^{nd} iteration



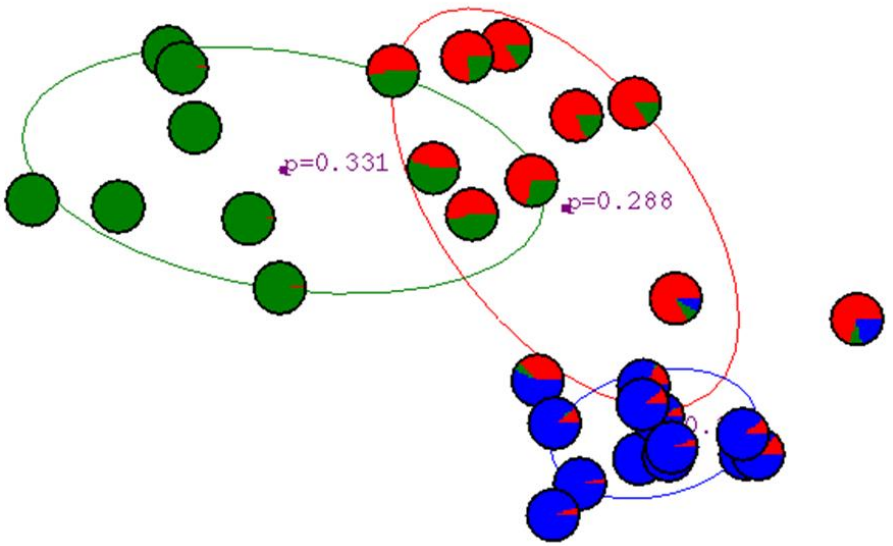
Gaussian Mixture Example

- 3rd iteration



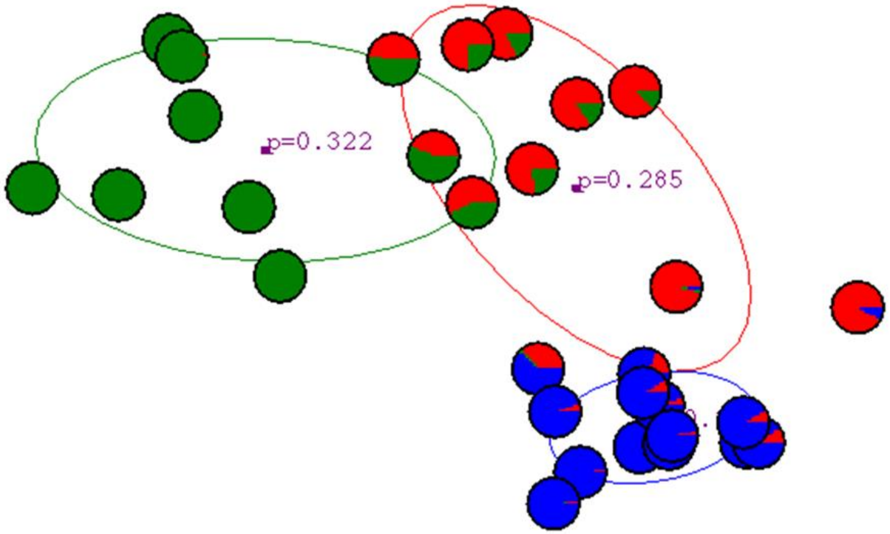
Gaussian Mixture Example

- 4^{th} iteration



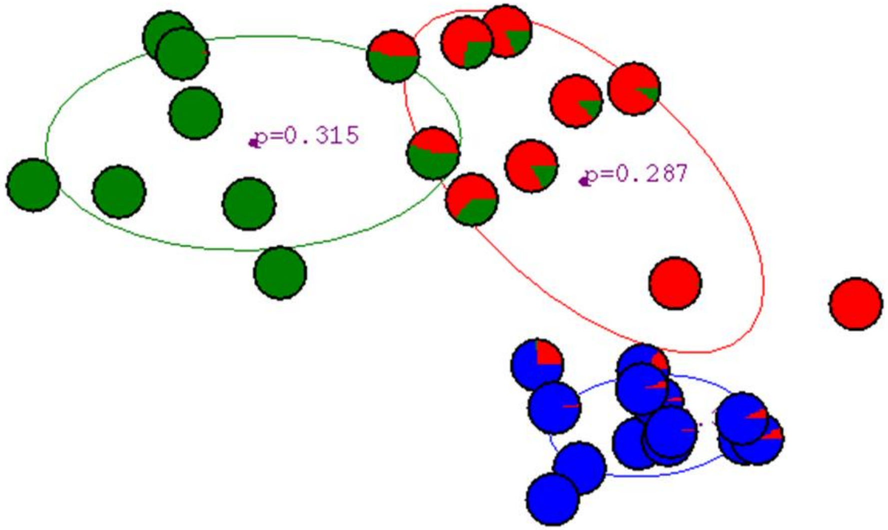
Gaussian Mixture Example

- 5^{th} iteration



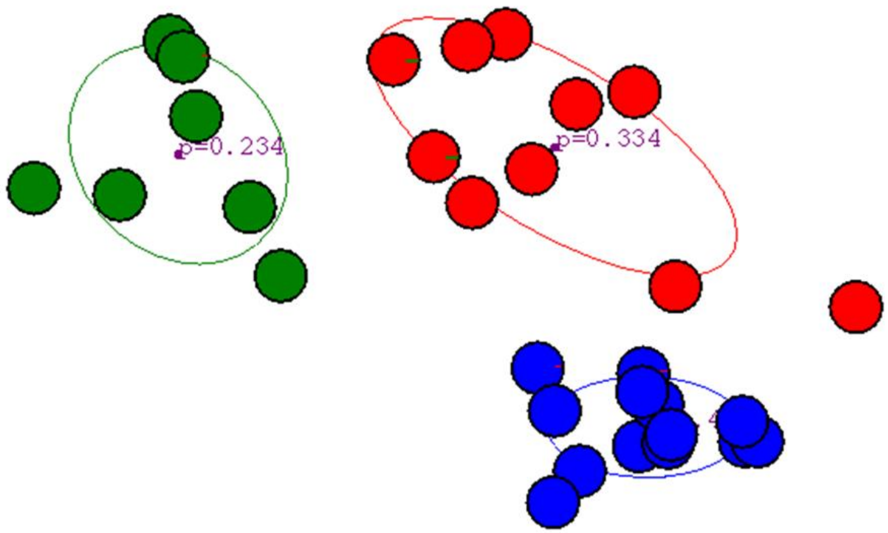
Gaussian Mixture Example

- 6^{th} iteration



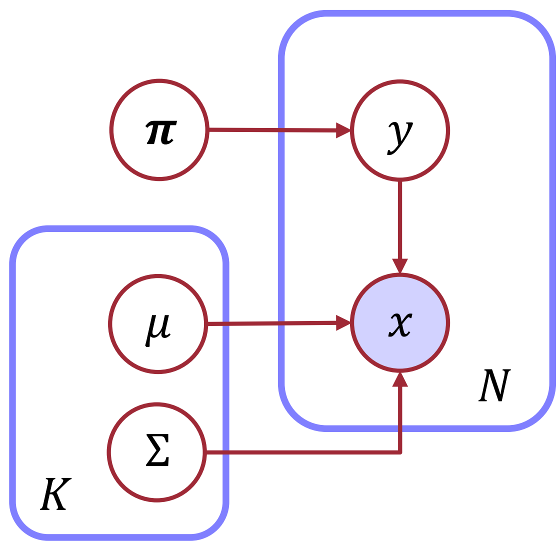
Gaussian Mixture Example

- 20^{th} iteration



Generative models

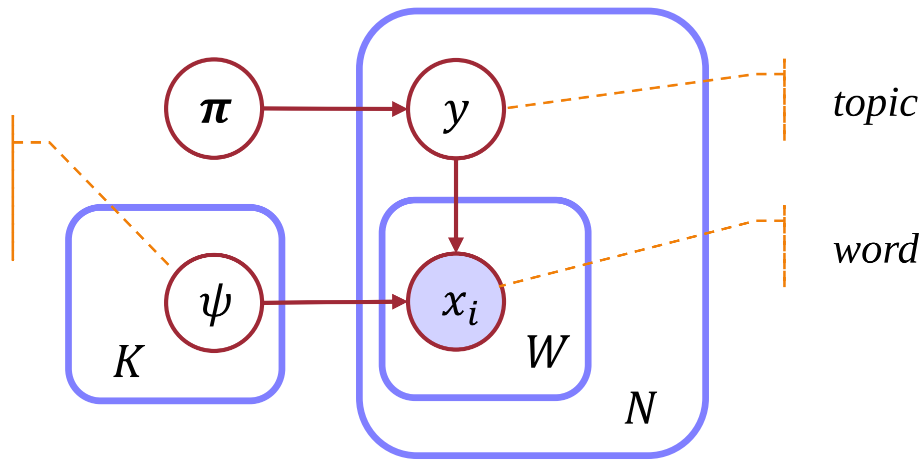
- MoG generates a document feature vector with one of K Gaussian distributions
- Can we directly generate a document (sequence of words)?
  - Yes! Generating words with one of K discrete distributions



Generative models

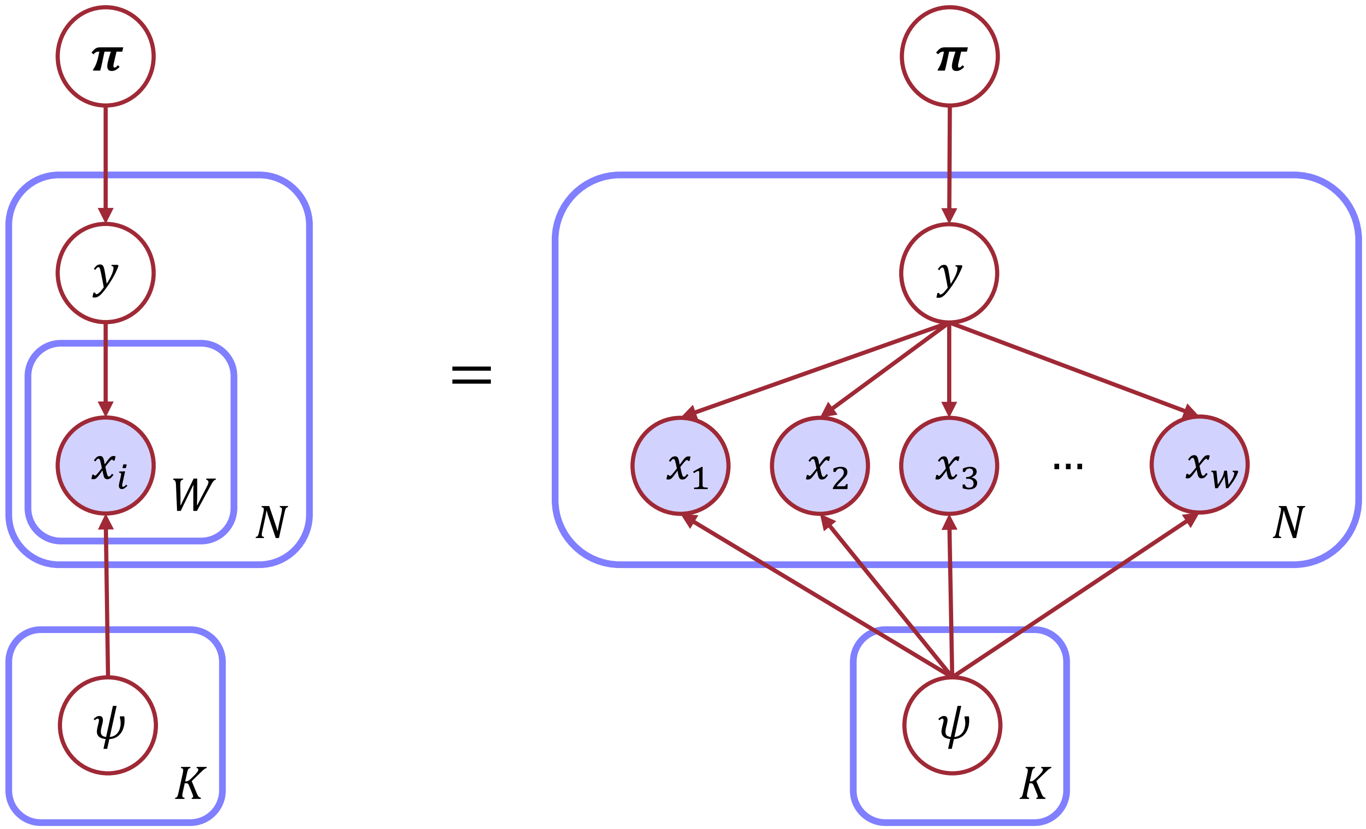
- MoG generates a document feature vector with one of K Gaussian distributions
- Can we directly generate a document (sequence of words)?
  - Yes! Generating words with one of K discrete distributions

distribution over words
for each topic



***e.g.*, <u>words - [*how, to, get, …*]</u>**
sports: [0.1, 0.3, 0.2, …]
arts: [0.2, 0.3, 0.1, …]
science: [0.3, 0.4, 0.1, …]
……

Generative models



This is exactly a naive Bayes model!

Unsupervised Naïve Bayes

- We can run EM for unsupervised learning of naive Bayes
  - i.e., text clustering based on words, not features
- [E step] Assign documents proportionately to different topics
  - Compute topic distribution of each document

$$\mathrm{P}\left(y_j = i \middle| x_{j,1:w}, \theta^{(t)}\right) \propto {\pi_i}^{(t)} \prod_{k=1}^{w} \mathrm{P}\left(x_{j,k} \middle| {\psi_i}^{(t)}\right)$$

Unsupervised Naïve Bayes

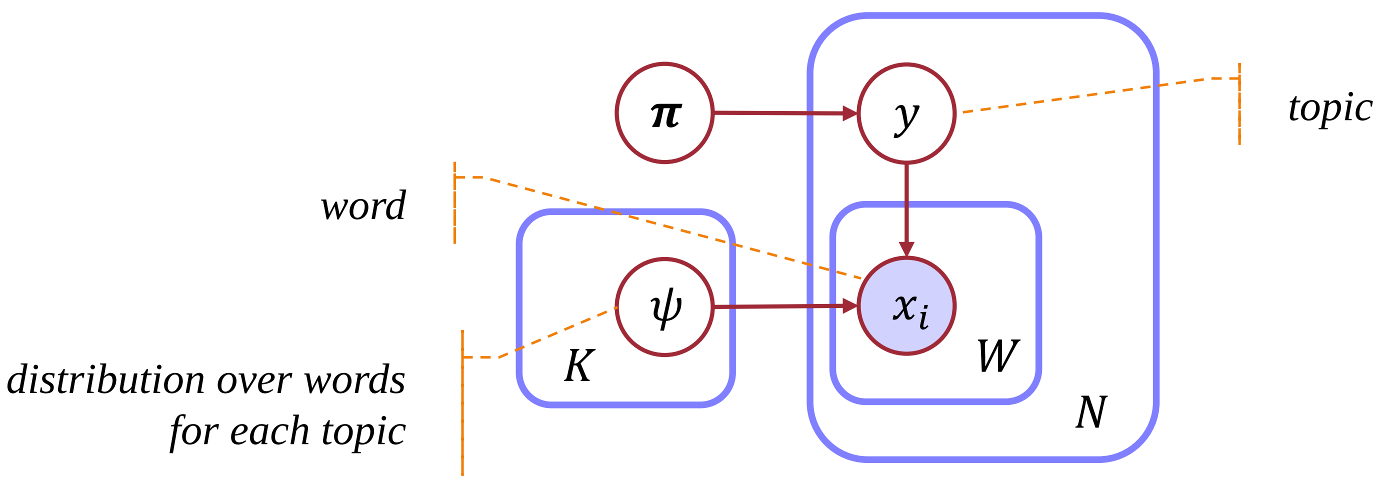
- We can run EM for unsupervised learning of naive Bayes
  - i.e., text clustering based on words, not features
- [M step] Revise each topic based on its (proportionately) assigned words
  - Compute weighted MLE of parameters given topic distributions
  - Denote $\psi_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,v}\}$

$$p_{i,l}^{(t+1)} = \frac{\sum_j \mathrm{P}\left(y_j = i \middle| x_{j,1:w}, \theta^{(t)}\right) \sum_k \mathbf{1}(x_{j,k} = l)}{\sum_j \mathrm{P}\left(y_j = i \middle| x_{j,1:w}, \theta^{(t)}\right) \cdot w_j} \qquad {\pi_i}^{(t+1)} = \frac{\sum_j \mathrm{P}\left(y_j = i \middle| x_{j,1:w}, \theta^{(t)}\right)}{m}$$

where v is the vocabulary size, m is the # of training documents.

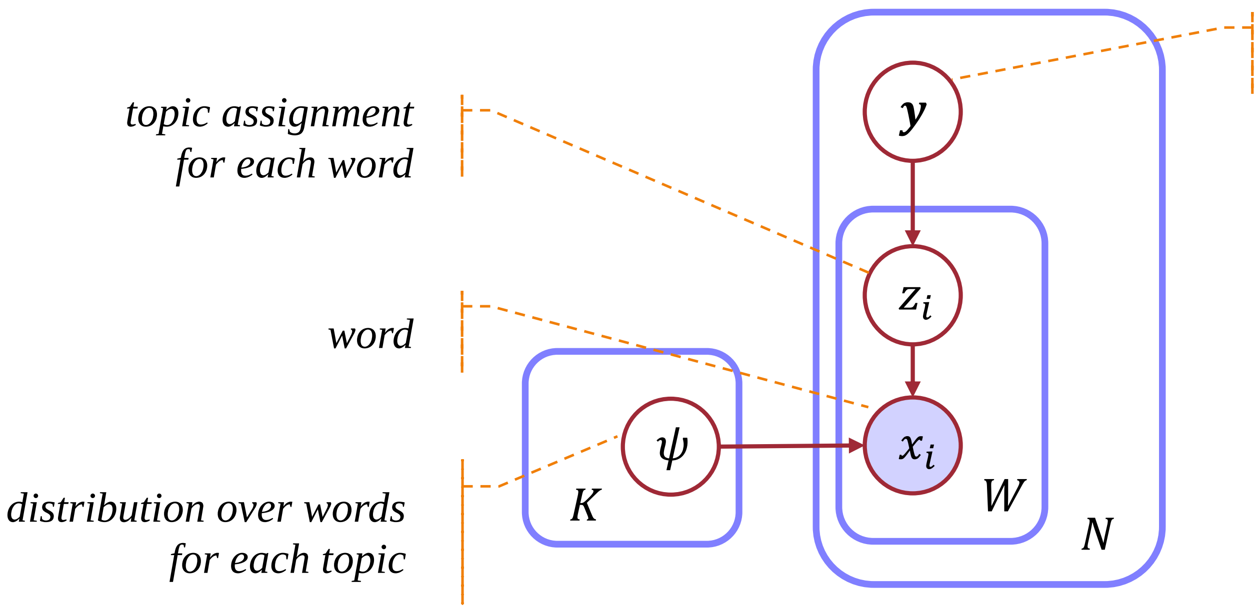
Topic modeling

- Text clusters may correspond to different topics
- So far, we assume a single cluster label for each document
- But, a document may cover multiple topics
  - Can we learn that?



Topic modeling

- Text clusters may correspond to different topics
- So far, we assume a single cluster label for each document
- But, a document may cover multiple topics
  - Can we learn that?

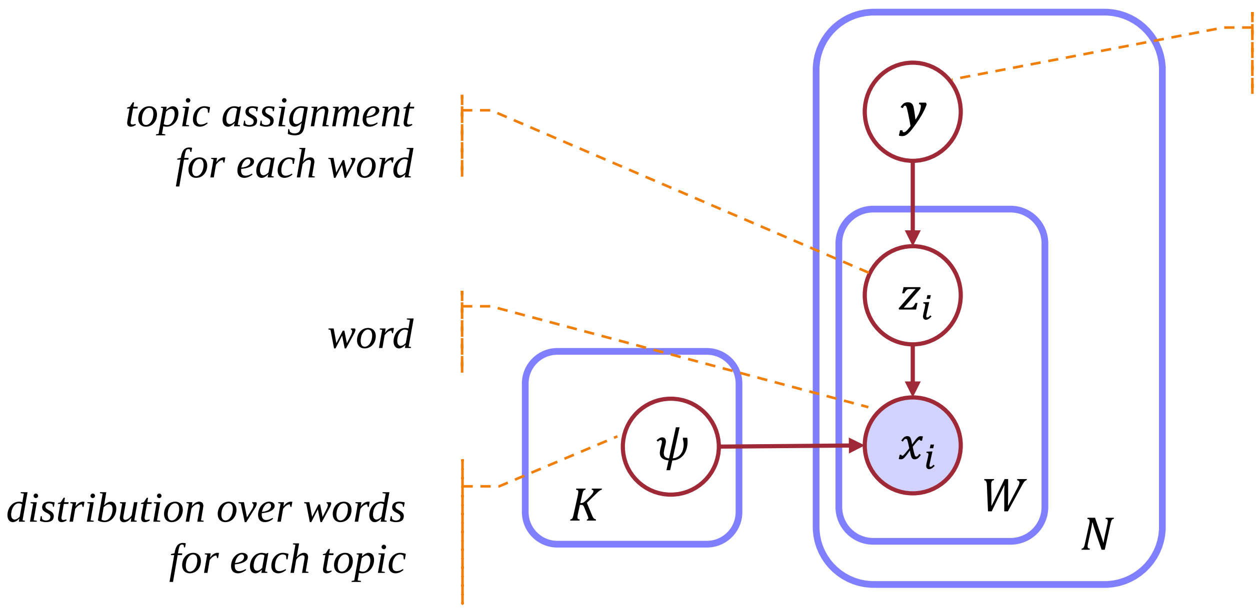


***e.g.*, [0.1, 0.3, 0.2, …]**

sports	**0.1**
arts	**0.3**
science	**0.2**
…	**…**

Topic modeling

- This is called probabilistic latent semantic analysis (pLSA)
  - Again, we can run EM to learn it
- We can further add Dirichlet priors over topic & word distributions

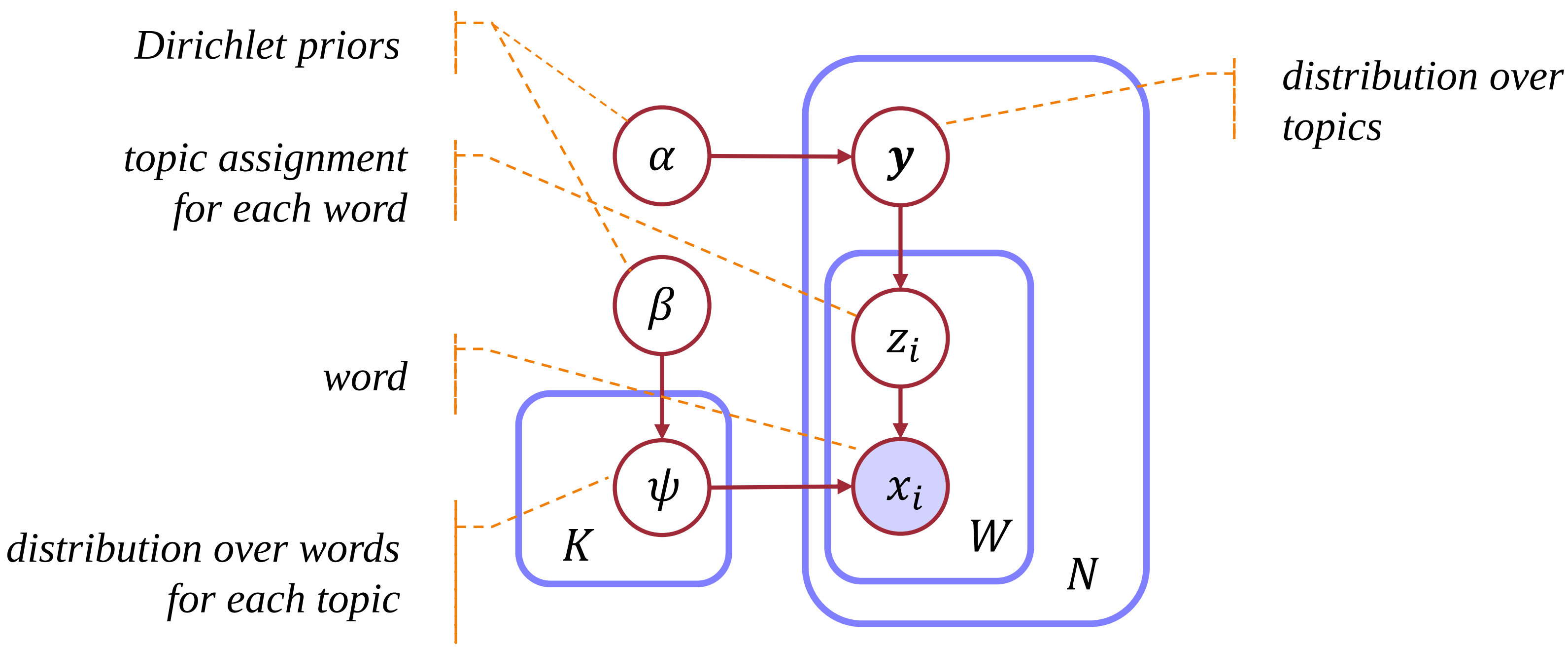


***e.g.*, [0.1, 0.3, 0.2, …]**

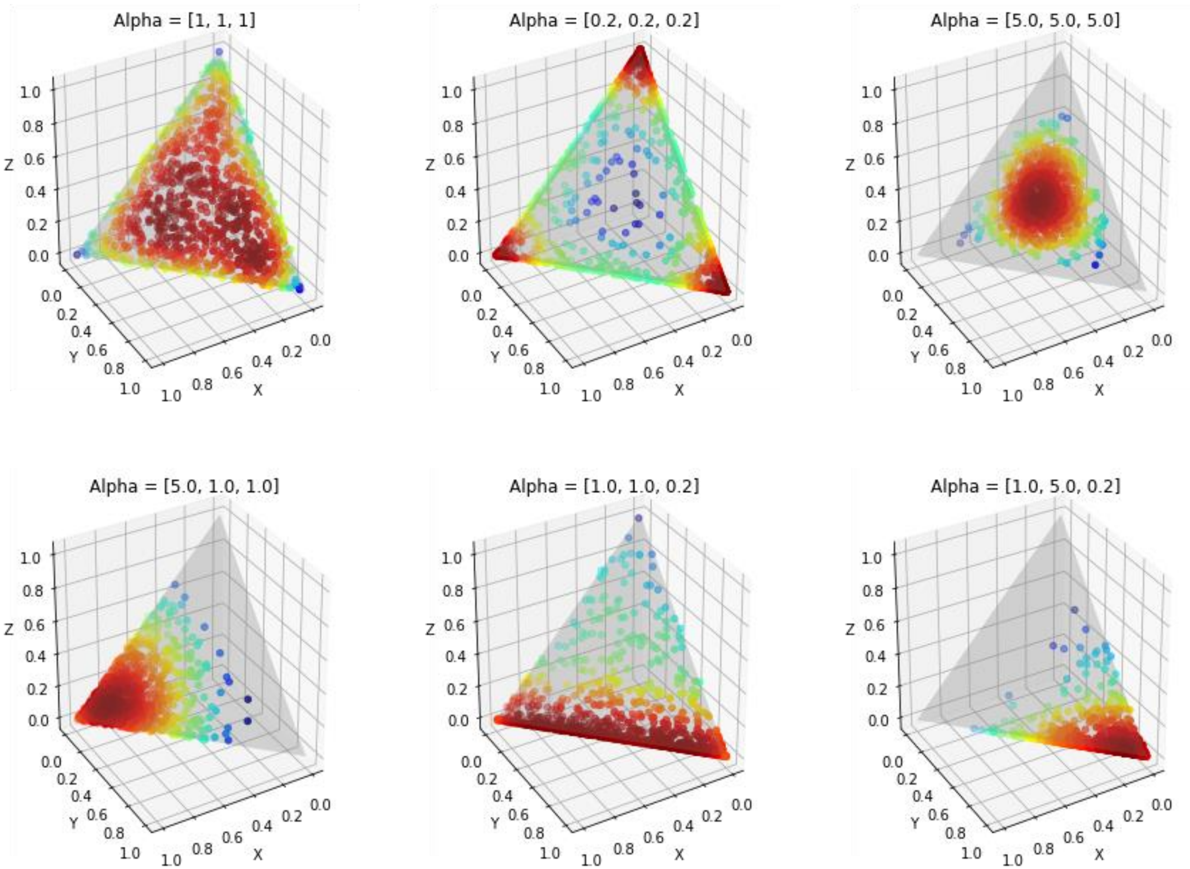
sports	**0.1**
arts	**0.3**
science	**0.2**
…	**…**

Topic modeling

- Dirichlet priors: encourage topic & word distributions to be sparse
  - A document shall cover only a few topics
  - In a topic, only a subset of words has high frequency

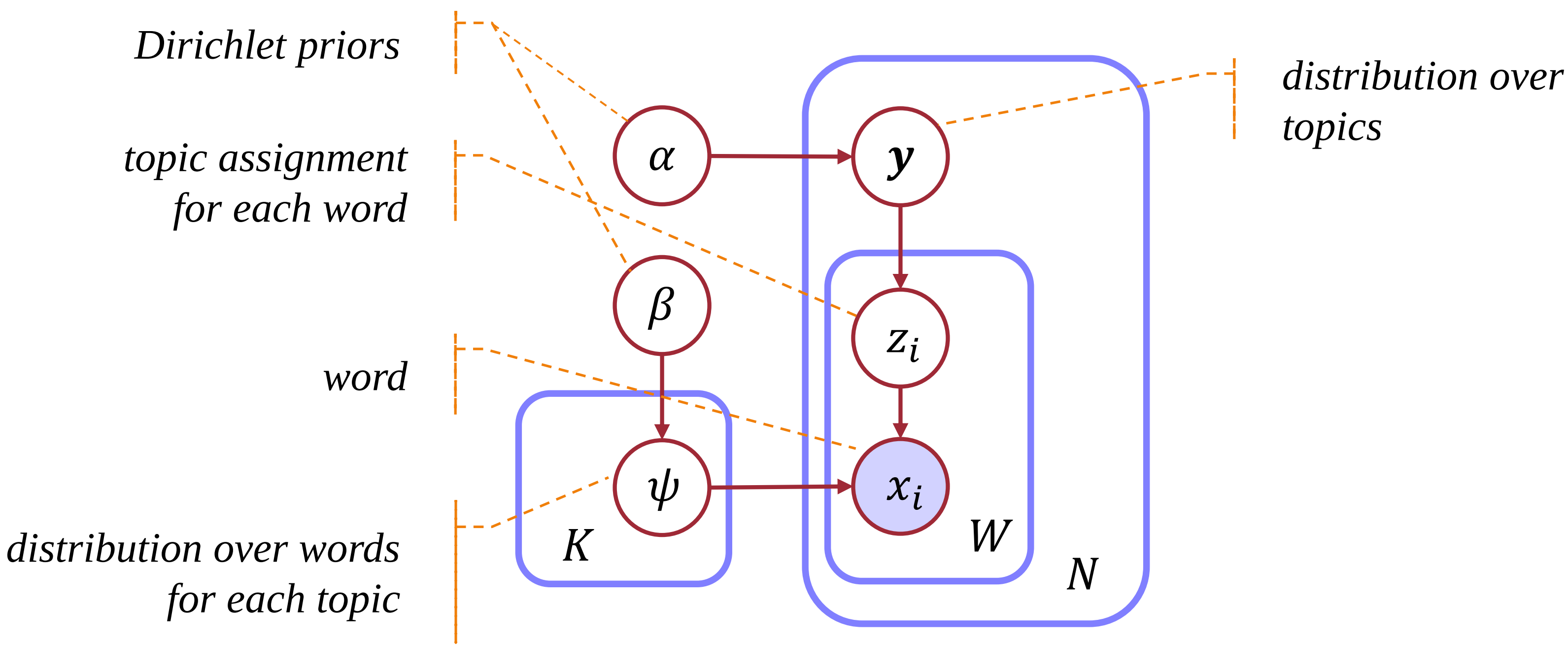


Dirichlet Distribution

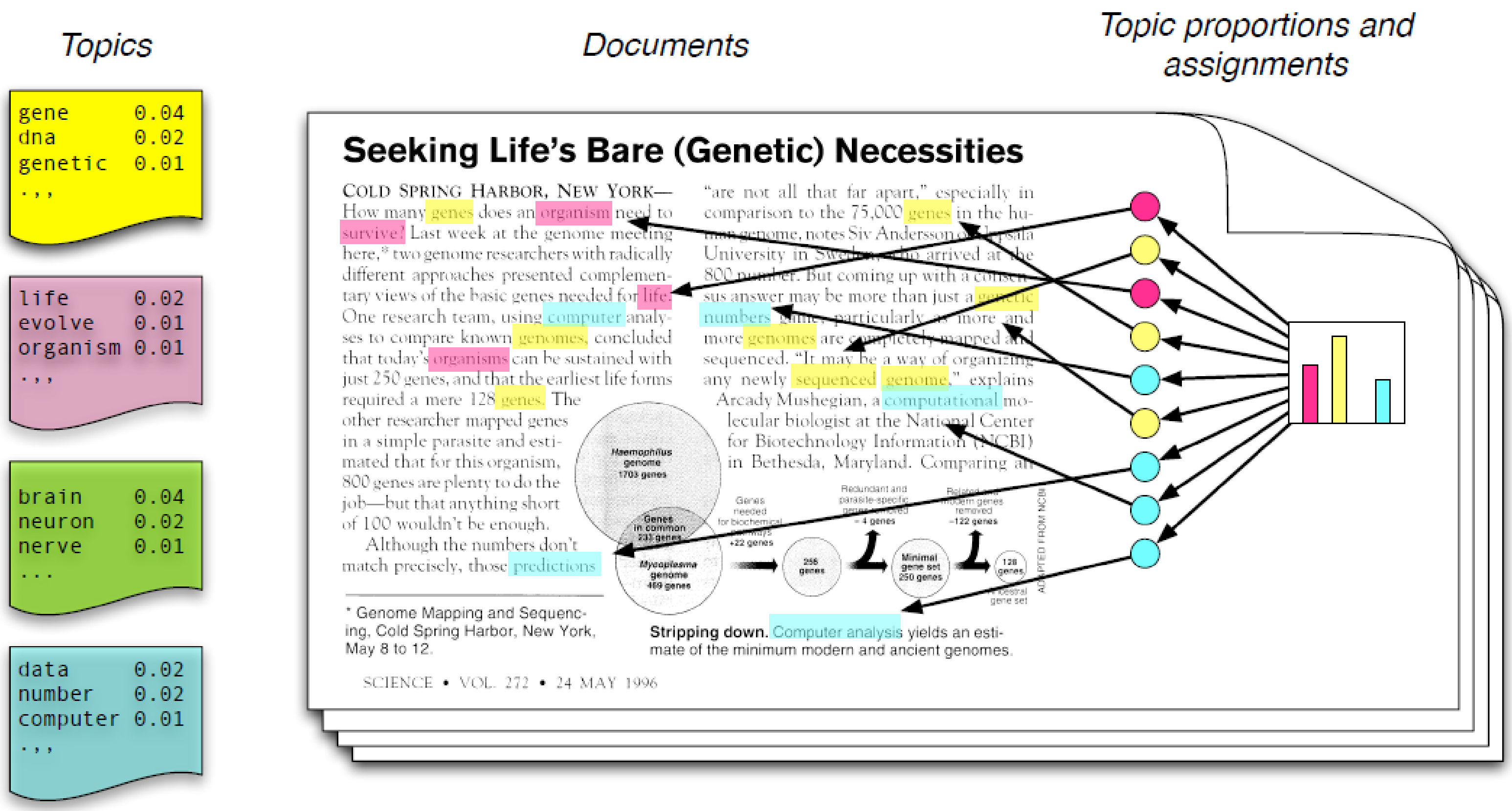


Topic modeling

- This is called Latent Dirichlet Allocation (LDA)
  - Learning: variational inference or MCMC

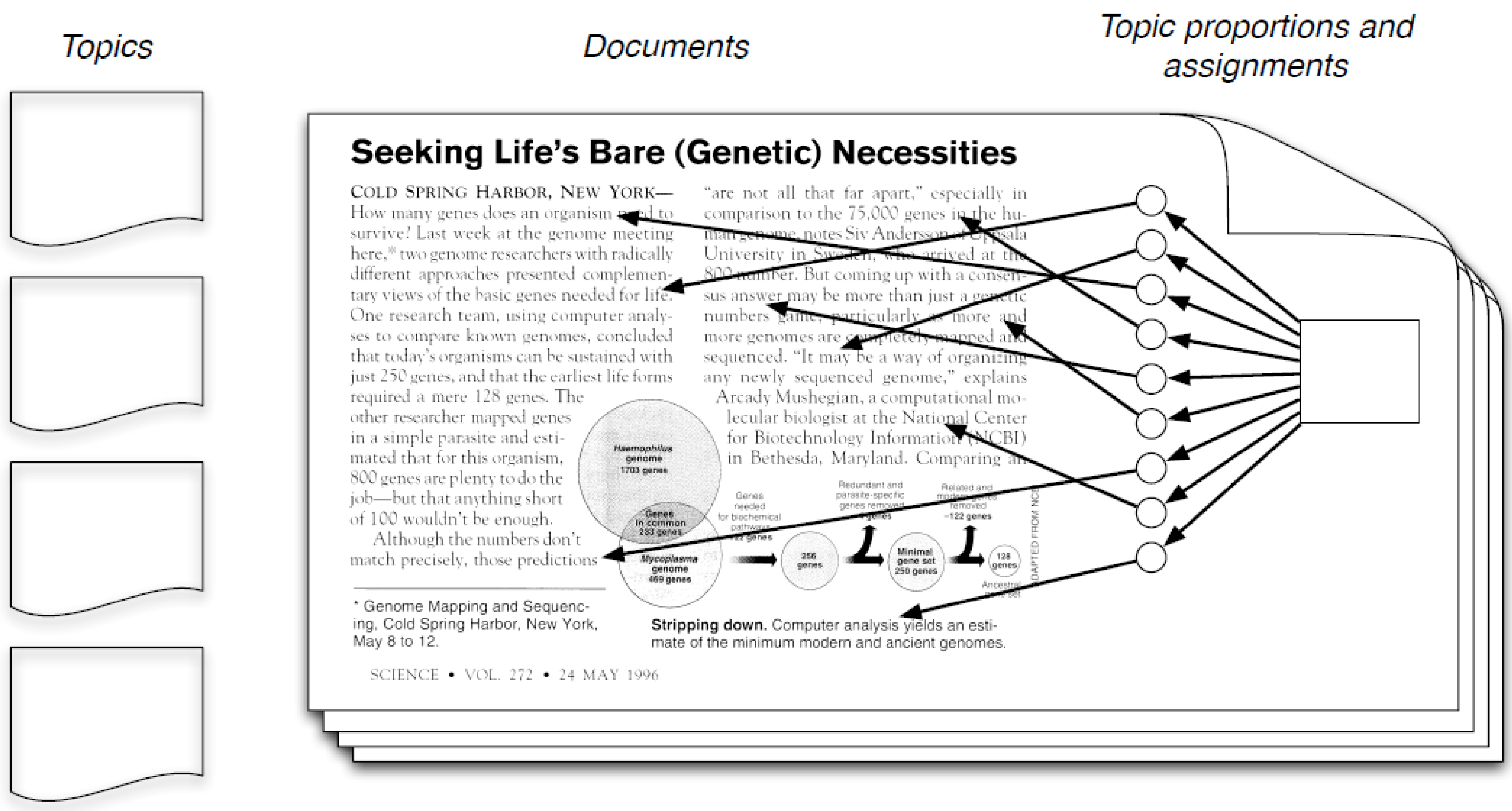


Illustration



- Each **topic** is a distribution of words; each **document** is a mixture of corpus-wide topics; and each **word** is drawn from one of those topics.

Illustration



- In reality, we only observe documents. The other structures are hidden variables that must be inferred.

Topics inferred by LDA

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

Topic assignments in document

- Based on the topics shown in last slide

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

EM in General

- Can be used to learn any model with hidden variables (missing data)
- Alternate:
  - Compute distributions over hidden variables based on current parameter values
  - Compute new parameter values to maximize expected log likelihood based on distributions over hidden variables
- Stop when no changes

- Can reach a local optimum but not necessarily a global optimum

Math Behind EM

- EM is coordinate ascent on F(θ, Q)

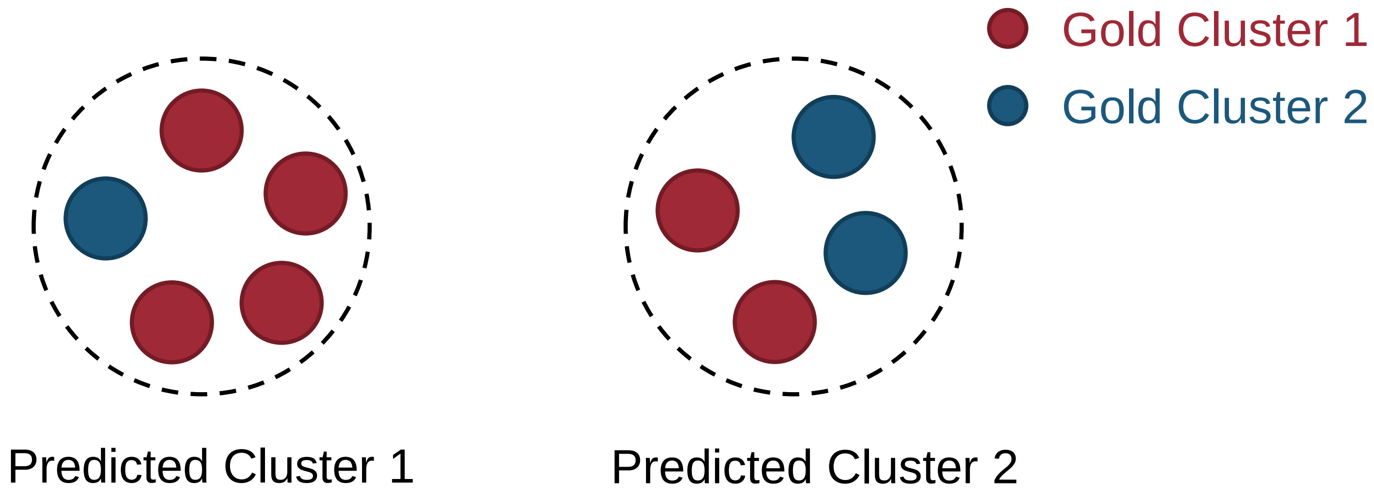
$$\ell(\theta : \mathcal{D}) \underset{\substack{\uparrow \\ \text{Jensen's inequality}}}{\geq} F(\theta, Q) = \sum_{j=1}^{m} \sum_{\mathbf{z}} Q(\mathbf{z} \mid \mathbf{x}_j) \log \frac{P(\mathbf{z}, \mathbf{x}_j \mid \theta)}{Q(\mathbf{z} \mid \mathbf{x}_j)}$$

  - E-step fixes θ and optimizes Q
  - M-step fixes Q and optimizes θ
- Convergence of EM
  - Neither E-step nor M-step decreases F(θ, Q)

Evaluation

Evaluation of Clustering

- Many different metrics
  - Purity / Inverse Purity
  - Rand index
  - MUC
  - B-CUBED
  - …



Evaluation

- Purity
  - Extent to which predicted clusters contain a single gold clusters.

$$Purity = \frac{1}{N} \sum_{m \in M} \max_{d \in D} |m \cap d|$$

N: # of data points

M: Set of predicted clusters

D: Set of gold clusters

Evaluation

- Purity

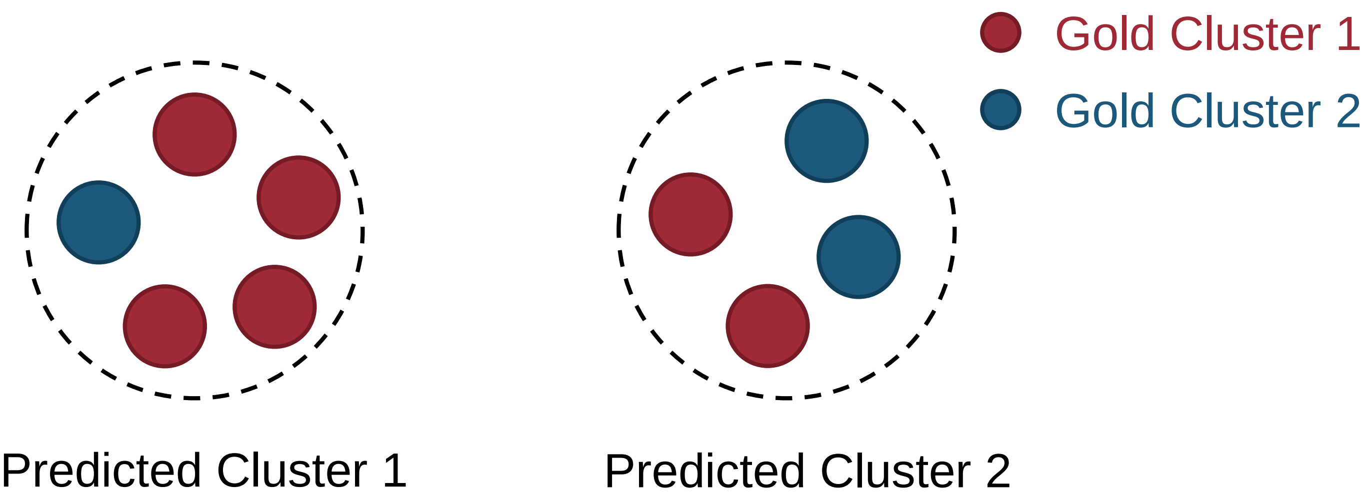
$$\frac{1}{N}\sum_{m\in M}\max_{d\in D}|m\cap d|$$

$$=\frac{1}{9}(\max\{4,1\}+\max\{2,2\})$$

$$=\frac{1}{9}(4+2)\approx 0.667$$

The Confusion Matrix

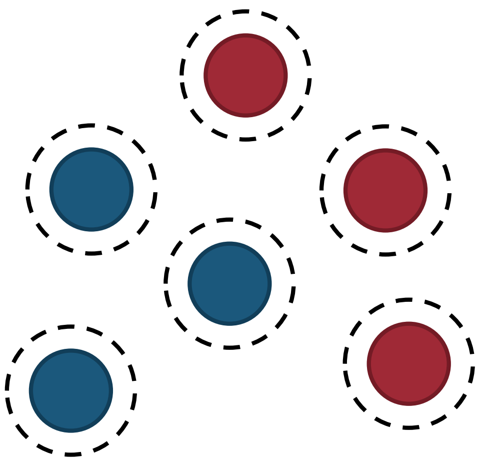
	Pred. 1	**Pred. 2**
Gold. 1	4	2
Gold. 2	1	2



Evaluation

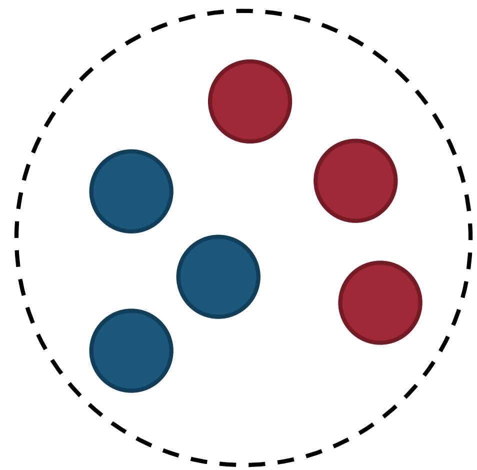
- Purity: $\frac{1}{N}\sum_{m\in M}\max_{d\in D}|m\cap d|$
- Inverse Purity
  - Same formula, with M & D exchanged

Purity=1



Inverse-Purity=1/3

Purity=1/2

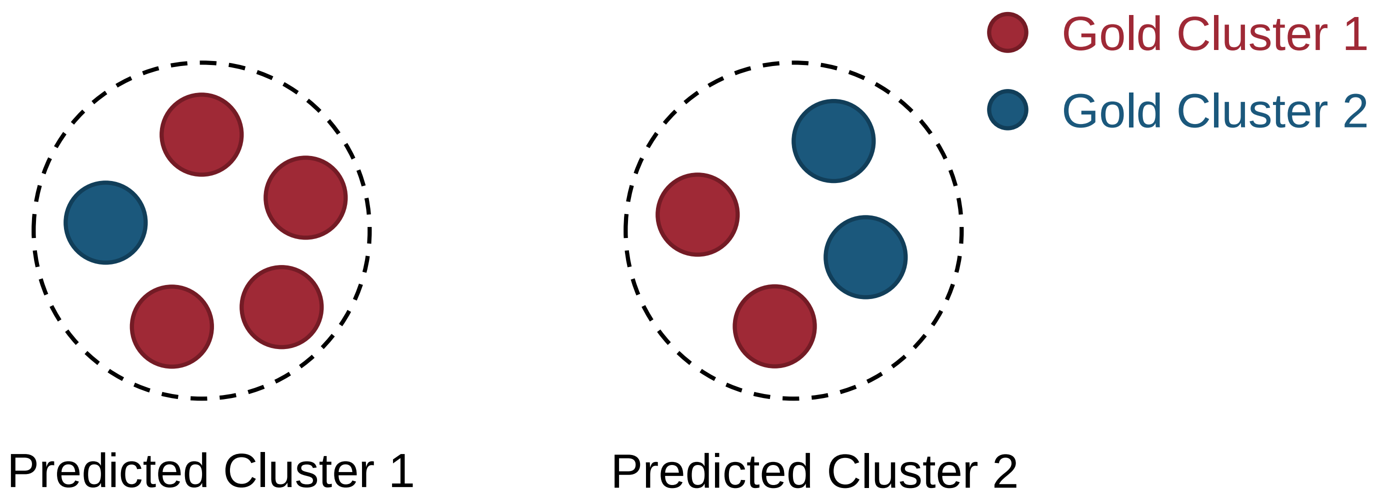


Inverse-Purity=1

Evaluation

- Rand Index
  - Computer accuracy of pairwise assignments: whether every pair of elements belong to the same cluster

$$RI = \frac{TP + TN}{TP + FP + FN + TN} = \frac{TP + TN}{\binom{N}{2}}$$

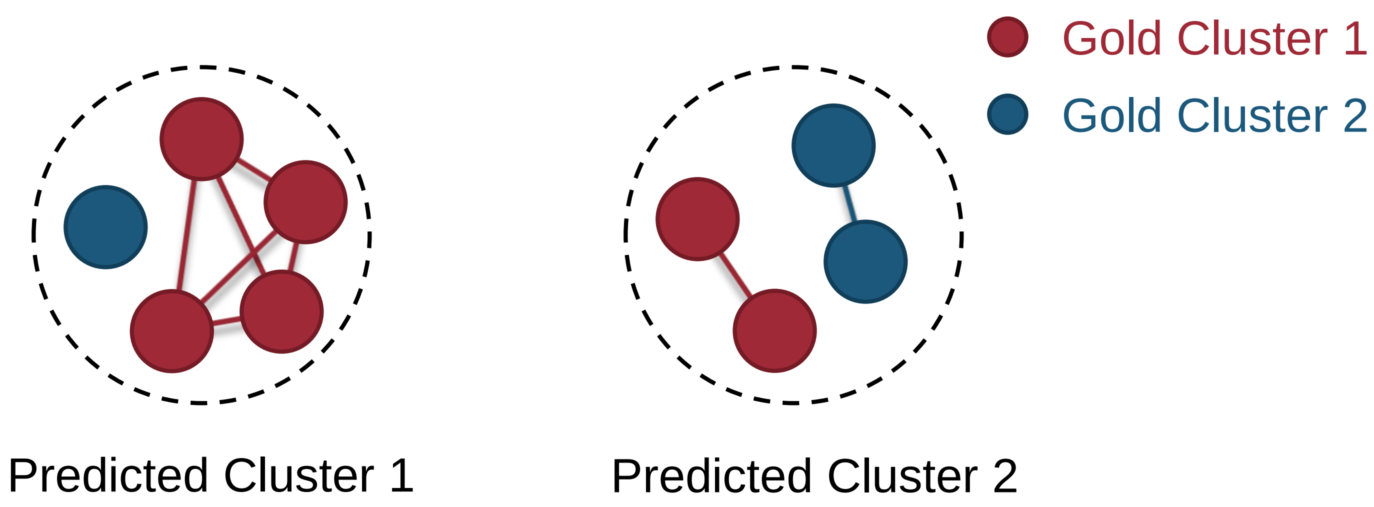


Evaluation

- Rand Index

$$RI = \frac{\mathrm{TP} + \mathrm{TN}}{TP + FP + FN + TN} = \frac{TP + TN}{\binom{N}{2}}$$

  - N: # of data points $\binom{9}{2} = 36$
  - TP: # of true positives 8

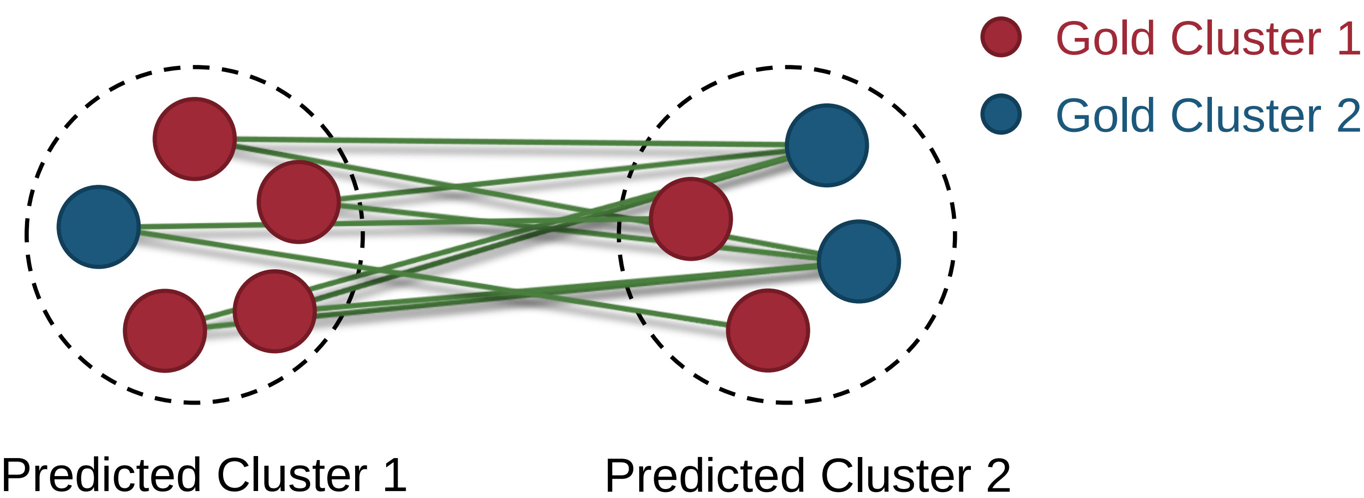


Evaluation

- Rand Index

$$RI = \frac{\mathrm{TP} + \mathrm{TN}}{TP + FP + FN + TN} = \frac{TP + TN}{\binom{N}{2}}$$

  - N: # of data points $\binom{9}{2} = 36$
  - TP: # of true positives 8
  - TN: # of true negatives 10



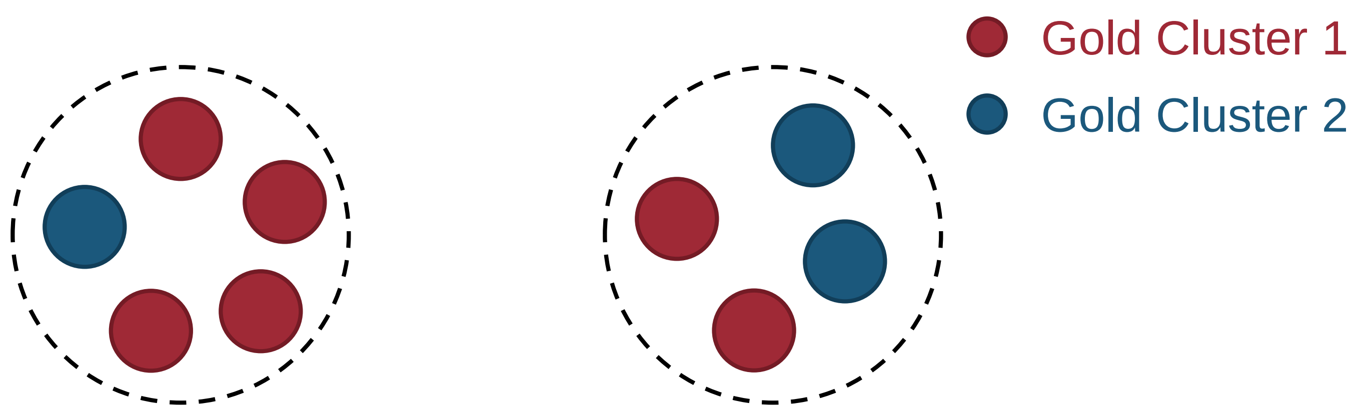
Evaluation

- Rand Index

$$RI = \frac{\mathrm{TP} + \mathrm{TN}}{TP + FP + FN + TN} = \frac{TP + TN}{\binom{N}{2}}$$

  - N: # of data points $\binom{9}{2} = 36$
  - TP: # of true positives 8
  - TN: # of true negatives 10

$$RI = \frac{10 + 8}{36} = 0.5$$

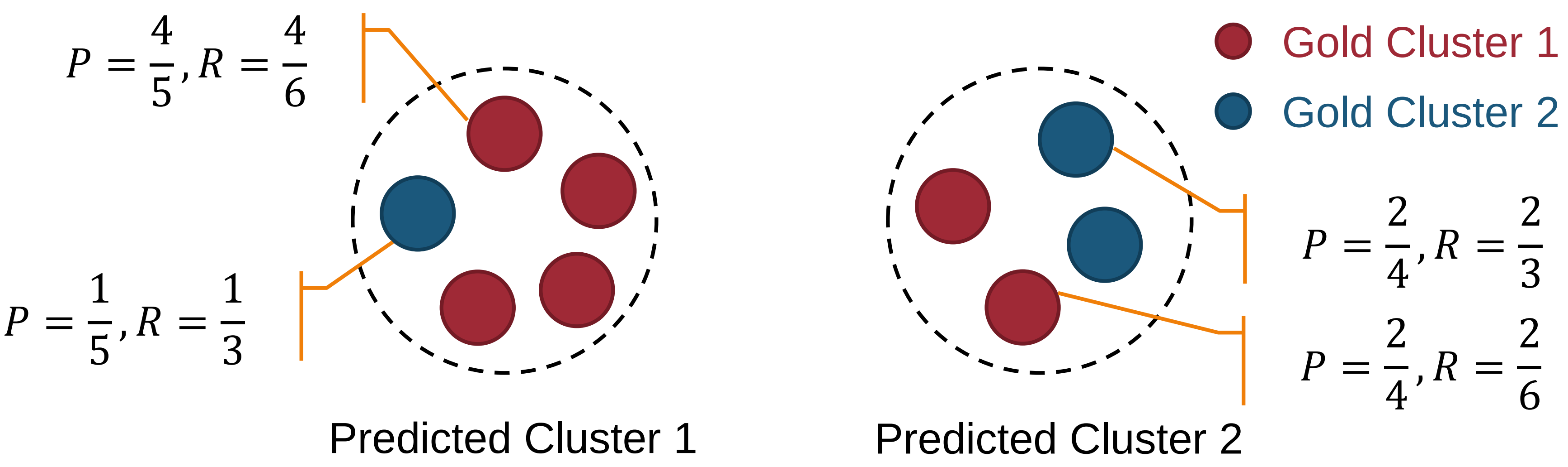


Predicted Cluster 1

Predicted Cluster 2

Evaluation

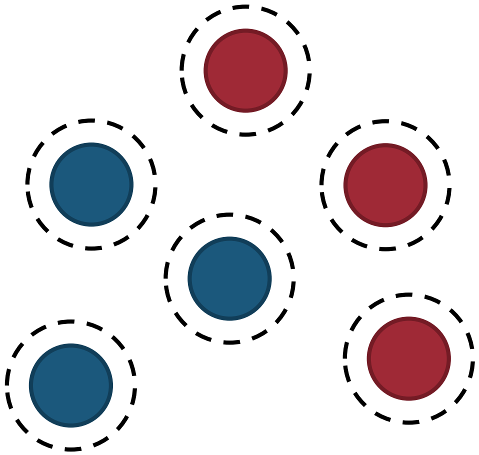
- B-cubed
  - For each element, compute a precision and a recall
  - Then average the individual Ps and Rs
    - P = [4(4/5) + 1(1/5) + 2(2/4) + 2(2/4)] / 9 = 0.6



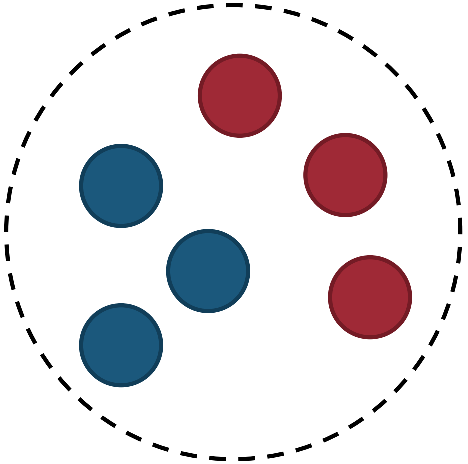
Evaluation

- B-cubed

100% Precision, 33% Recall



50% Precision, 100% Recall



Summary

Text Clustering

- Mixture of Gaussian
- Unsupervised Naive Bayes
- Topic models
  - pLSA, LDA
- Learning
  - Expectation-maximization
- Evaluation