Recent Advances in Bayesian Methods

Jun Zhu

dcszj@mail.tsinghua.edu.cn

Department of Computer Science and Technology

Tsinghua University

ACML 2013, Canberra, Nov 13, 2013

Overview

- **Part I (this morning, 1 hr):**
  - Basics of Bayesian methods
  - Nonparametric Bayesian methods
- **Lunch refill break**
- **Part II (this afternoon, 1.5 hr):**
  - Constrained Bayesian methods
  - Applications

Basic Rules of Probability

$p(X)$ probability of X

$p(X|\mathcal{M})$ conditional probability of X given $\mathcal{M}$

$p(X, \mathcal{M})$ joint probability of X and $\mathcal{M}$

- Joint probability – product rule

$$p(X, \mathcal{M}) = p(X|\mathcal{M})p(\mathcal{M})$$

- Marginal probability – sum/integral rule

$$p(X) = \int p(X|\mathcal{M})p(\mathcal{M})d\mathcal{M}$$

- Conditional probability

$$p(\mathcal{M}|X) = \frac{p(X, \mathcal{M})}{p(X)}$$

Bayes' Rule

- Combining the definition of conditional prob. with the product and sum rules, we have Bayes' rule or Bayes' theorem

$$p(\mathcal{M}|X) = \frac{p(X,\mathcal{M})}{p(X)} = \frac{p(\mathcal{M})p(X|\mathcal{M})}{\int p(\mathcal{M})p(X|\mathcal{M})d\mathcal{M}}$$

Thomas Bayes (1702 – 1761)

- "*An Essay towards Solving a Problem in the Doctrine of Chances*" published at Philosophical Transactions of the Royal Society of London in 1763
- This year marks the 250th Anniversary of Bayes' theorem
  - Events at: http://bayesian.org/

Bayes' Rule in Machine Learning

- Let $\mathcal{D}$ be a given data set; $\mathcal{M}$ be a model

$$p(\mathcal{M}|\mathcal{D}) = \frac{p(\mathcal{M})p(\mathcal{D}|\mathcal{M})}{p(\mathcal{D})}$$

$p(\mathcal{M})$	prior probability of $\mathcal{M}$
$p(\mathcal{D}\|\mathcal{M})$	likelihood of $\mathcal{M}$ on data
$p(\mathcal{M}\|\mathcal{D})$	posterior probability of $\mathcal{M}$ given $\mathcal{D}$
$p(\mathcal{D})$	marginal likelihood or evidence

- Model Comparison: $\mathbb{M} = \{\mathcal{M}\}$

$$p(\mathbb{M}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbb{M})p(\mathbb{M})}{p(\mathcal{D})} \qquad p(\mathcal{D}|\mathbb{M}) = \int p(\mathcal{D}|\mathcal{M},\mathbb{M})p(\mathcal{M}|\mathbb{M})d\mathcal{M}$$

- Prediction:

$$p(x|\mathcal{D},\mathbb{M}) = \int p(x|\mathcal{M},\mathcal{D},\mathbb{M})p(\mathcal{M}|\mathcal{D},\mathbb{M})d\mathcal{M}$$

under some common assumptions

$$p(x|\mathcal{M})$$

Common Questions

- Why be Bayesian?
- Where does the prior come from?
- How do we do these integrals?

Why Be Bayesian?

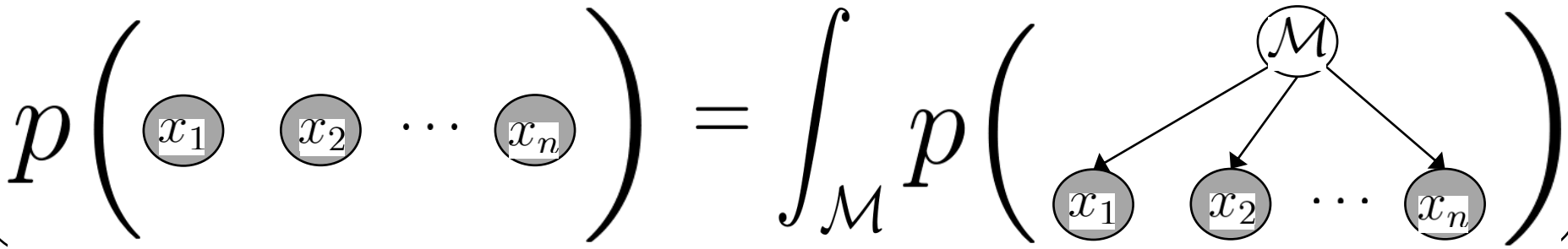
- One of many answers
- Infinite Exchangeability:

$$\forall n,\ \forall \sigma,\ p(x_1, \ldots, x_n) = p(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$$

- De Finetti's Theorem (1955): if $(x_1, x_2, \ldots)$ are *infinitely exchangeable*, then $\forall n$

$$p(x_1, \ldots, x_n) = \int \left(\prod_{i=1}^{n} p(x_i|\mathcal{M}) \right) dP(\mathcal{M})$$

for some random variable $\mathcal{M}$



How to Choose Priors?

- **Objective priors** -- noninformative priors that attempt to capture ignorance and have good frequentist properties
- **Subjective priors** -- priors should capture our beliefs as well as possible
- **Hierarchical priors** -- multiple layers of priors

$$p(\mathcal{M}) = \int p(\mathcal{M}|\alpha)p(\alpha)d\alpha = \int\int p(\mathcal{M}|\alpha)p(\alpha|\beta)p(\beta)d\alpha d\beta = \cdots$$

  - the higher, the weaker
- **Empirical priors** -- Learn some of the parameters of the prior from the data; known as "Empirical Bayes"

$$p(\mathcal{M}|\hat{\alpha}) \qquad \hat{\alpha} = \operatorname*{argmax}_{\alpha} p(\mathcal{D}|\alpha)$$

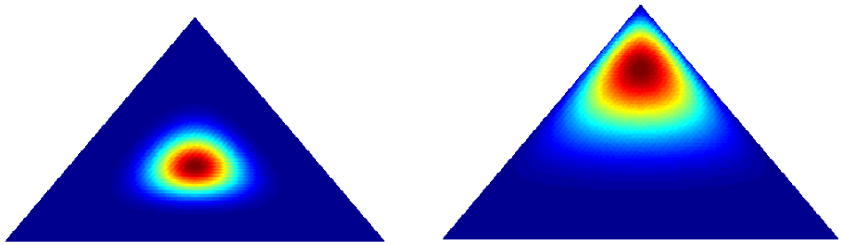
  - **Pros**: robust – overcomes some limitations of mis-specification
  - **Cons**: double counting of evidence / overfitting

How to Choose Priors?

- Conjugate and Non-conjugate tradeoff

- Conjugate priors are relatively easier to compute, but they might be limited
  - Ex: Gaussian-Gaussian, Dirichlet-Multinomial, Beta-Bernoulli, etc.

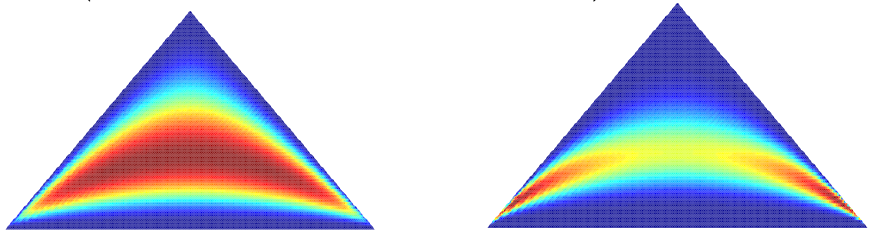
$$p(\theta|\alpha) \propto \theta_1^{\alpha_1-1}\theta_2^{\alpha_2-1}\theta_3^{\alpha_3-1}$$



- Non-conjugate priors are more flexible, but harder to compute
  - Ex: LogisticNormal-Multinomial (Aitchison & Shen, 1980)

$$\eta \sim \mathcal{N}(0, \Sigma)$$

$$\theta_k \propto \exp(\eta_k)$$



How do We Compute the Integrals?

- Recall that:

$$p(\mathcal{D}|\mathbb{M}) = \int p(\mathcal{D}|\mathcal{M}, \mathbb{M}) p(\mathcal{M}|\mathbb{M}) d\mathcal{M}$$

- This can be a very high dimensional integral

- If we consider latent variables, it leads to additional dimensions to be integrated out

$$p(\mathcal{D}|\mathbb{M}) = \int \int p(\mathcal{D}, H|\mathcal{M}, \mathbb{M}) p(\mathcal{M}|\mathbb{M}) dH d\mathcal{M}$$

  - This could be very complicated!

Approximate Bayesian Inference

- In many cases, we resort to approximation methods

- Common examples
  - Variational approximations
  - Markov chain Monte Carlo methods (MCMC)
  - Expectation Propagation (EP)
  - Laplace approximation
  - …

- Developing **accurate** and **scalable** inference algorithms is an active area!

Basics of Variational Approximation

- We can lower bound the marginal likelihood

$$\log p(\mathcal{D}|\mathbb{M}) = \log \int\int p(\mathcal{D}, H|\mathcal{M}, \mathbb{M}) p(\mathcal{M}|\mathbb{M}) dH d\mathcal{M}$$

$$= \log \int\int q(H, \mathcal{M}) \frac{p(\mathcal{D}, H|\mathcal{M}, \mathbb{M}) p(\mathcal{M}|\mathbb{M})}{q(H, \mathcal{M})} dH d\mathcal{M}$$

$$\geq \int\int q(H, \mathcal{M}) \log \frac{p(\mathcal{D}, H|\mathcal{M}, \mathbb{M}) p(\mathcal{M}|\mathbb{M})}{q(H, \mathcal{M})} dH d\mathcal{M}$$

  - Note: the lower bound is tight if no assumptions made

- **Mean-field assumptions**: a factorized approximation

$$q(H, \mathcal{M}) = q(H) q(\mathcal{M})$$

  - optimizes the lower bound with the assumption leads to local optimums

Basics of MCMC

- To draw samples from a desired distribution $p(\mathcal{M}|\mathcal{D})$

- We define a Markov chain

$$\mathcal{M}_0 \to \mathcal{M}_1 \to \mathcal{M}_2 \to \mathcal{M}_3 \to \cdots$$

  - where $p_t(\mathcal{M}) = \int p_{t-1}(\mathcal{M}')T(\mathcal{M}' \to \mathcal{M})d\mathcal{M}'$

  - $T(\mathcal{M}' \to \mathcal{M})$ is the Markov chain transition probability

- $p(\mathcal{M}|\mathcal{D})$ is an **invariant (or stationary) distribution** of the Markov chain T iff:

$$p(\mathcal{M}|\mathcal{D}) = \int p(\mathcal{M}'|\mathcal{D})T(\mathcal{M}' \to \mathcal{M})d\mathcal{M}'$$

Basics of MCMC

- A useful condition that implies invariance of $p(\mathcal{M}|\mathcal{D})$ is **detailed balance**:

$$p(\mathcal{M}'|\mathcal{D})T(\mathcal{M}' \to \mathcal{M}) = p(\mathcal{M}|\mathcal{D})T(\mathcal{M} \to \mathcal{M}')$$

- MCMC methods define ergodic Markov chains that converge to a unique stationary distribution (or equilibrium distribution) regardless of the initial states:

$$\lim_{t\to\infty} p_t(\mathcal{M}) = p(\mathcal{M}|\mathcal{D})$$

Parametric Bayesian Inference

$\mathcal{M}$ is represented as a finite set of parameters θ

- A parametric likelihood: $\mathbf{x} \sim p(\cdot|\theta)$
- Prior on θ : $p(\theta)$
- Posterior distribution

$$p(\theta|\mathbf{x}) = \frac{p(\mathbf{x}|\theta)p(\theta)}{p(\mathbf{x})} \propto p(\mathbf{x}|\theta)p(\theta)$$

Examples:

- Gaussian distribution prior + Gaussian likelihood → Gaussian posterior distribution
- Dirichilet distribution prior + Multinomial likelihood → Dirichlet posterior distribution
- Sparsity-inducing priors + some likelihood models → Sparse Bayesian inference

Nonparametric Bayesian Inference

$\mathcal{M}$ is a richer model, e.g., with an infinite set of parameters

- A nonparametric likelihood: $\mathbf{x} \sim p(\cdot|\mathcal{M})$
- Prior on $\mathcal{M}$: $p(\mathcal{M})$
- Posterior distribution

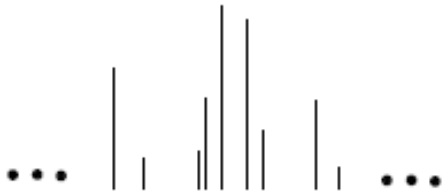
$$p(\mathcal{M}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M})p(\mathcal{M})}{p(\mathbf{x})} \propto p(\mathbf{x}|\mathcal{M})p(\mathcal{M})$$

Examples:

→ see next slide

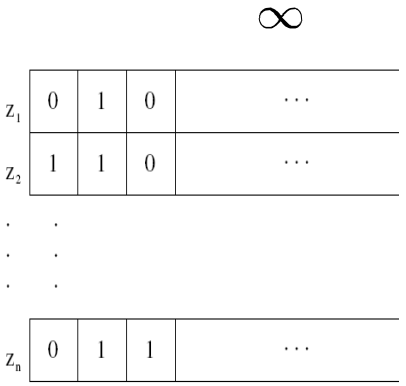
Nonparametric Bayesian Inference

probability measure



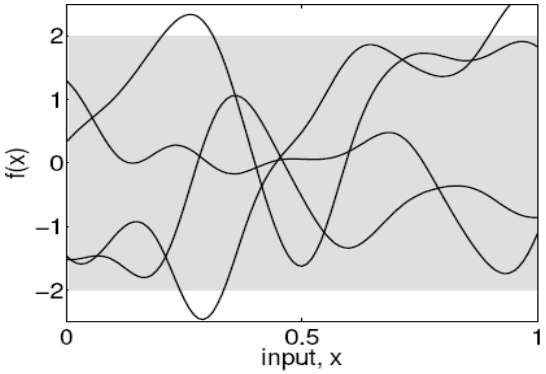
Dirichlet Process Prior [Antoniak, 1974]
+ Multinomial/Gaussian/Softmax likelihood

binary matrix



Indian Buffet Process Prior [Griffiths & Gharamani, 2005]
+ Gaussian/Sigmoid/Softmax likelihood

function

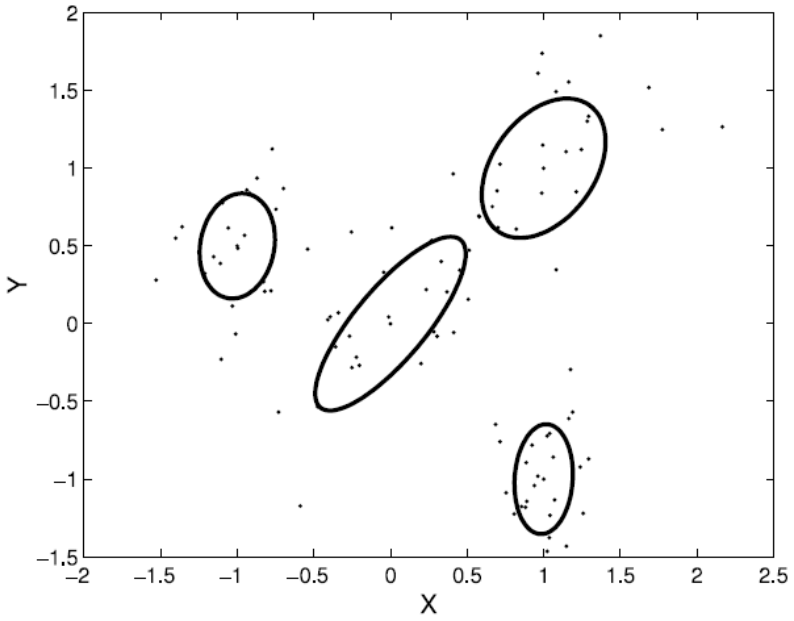


Gaussian Process Prior [Doob, 1944; Rasmussen & Williams, 2006]
+ Gaussian/Sigmoid/Softmax likelihood

Why Be Bayesian Nonparametrics?

Let the data speak for themselves

- Bypass the model selection problem
  - let data determine model complexity (e.g., the number of components in mixture models)
  - allow model complexity to grow as more data observed



Related Tutorials and Materials

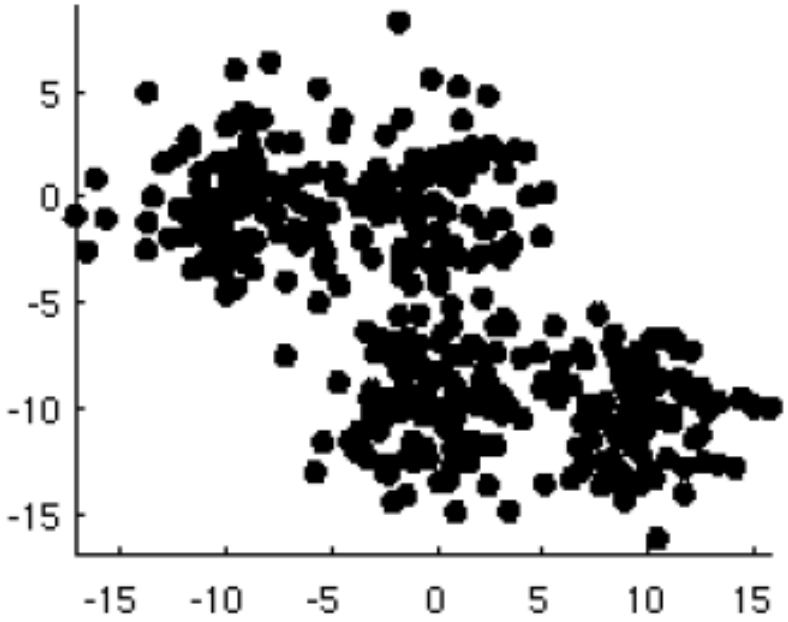
- Tutorial talks:
  - Zoubin Gharamani, ICML 2004. "Bayesian Methods for Machie Learning"
  - Michael Jordan, NIPS 2005. "Nonparametric Bayesian Methods: Dirichlet Processes, Chinese Restaurant Processes and All That"
  - Peter Orbanz, 20009. "Foundations of Nonparametric Bayesian Methods"
  - Yee Whye Teh, 2011. "Modern Bayesian Nonparametrics"

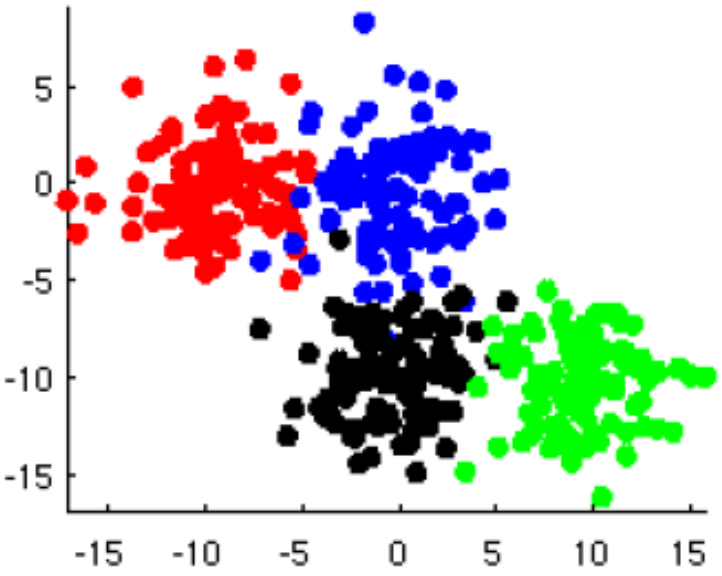
- Tutorial articles:
  - Gershman & Blei. A Tutorial on Bayesian Nonparametric Models. Journal of Mathematical Psychology, 56 (2012) 1-12

Clustering

Clustering

- Given a set of observations
- Each observation belong to exactly one cluster





A Bayesian Approach to Clustering

- We must specify two things:
  - likelihood model (how data is affected by parameters)

$$p(\mathcal{D}|\theta)$$

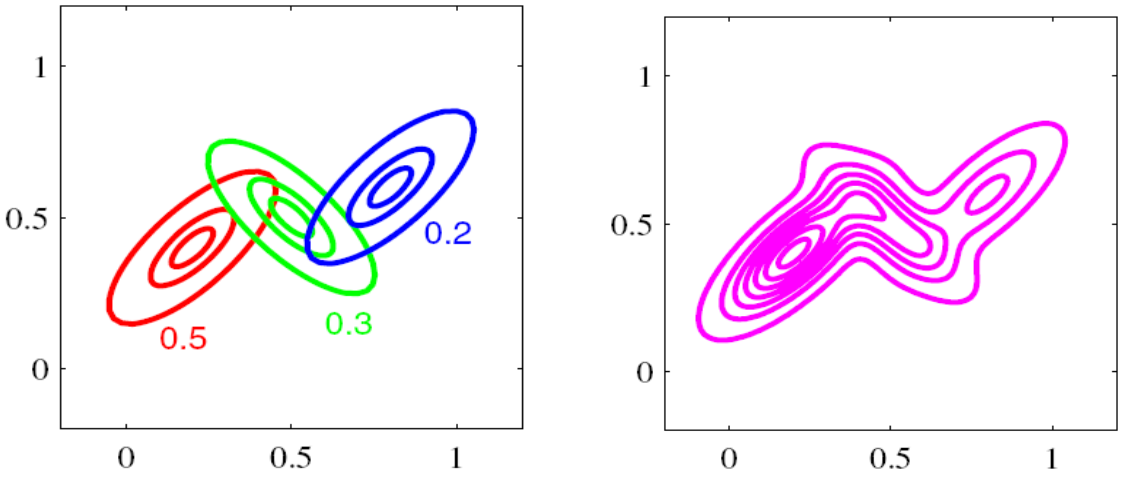
  - prior distribution (the prior belief on the parameters)

$$p(\theta)$$

Clustering – A Parametric Approach

- Guassian Mixture Models with K components
  - a distribution over classes/clusters: $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$
  - each cluster has a mean and covariance $\phi_k = (\mu_k, \Sigma_k)$

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)$$



  - Using EM to maximize the likelihood of the data to estimate $(\boldsymbol{\pi}, \phi)$

[Figure credit: Bishop, 2006]

Clustering – A Parametric Approach

- Guassian Mixture Models with K components
- An alternative definition

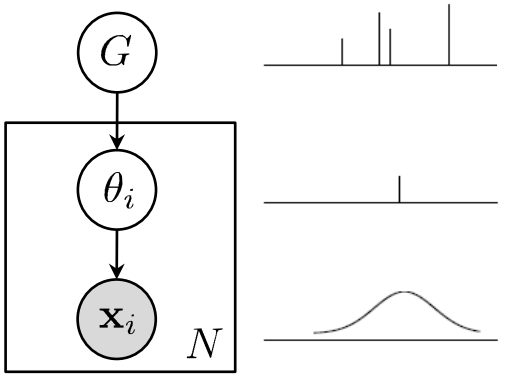
$$G = \sum_{k=1}^{K} \pi_k \delta_{\phi_k}$$

where is δ_{ϕ_k} an *atom* at ϕ_k

- Then,

$$\theta_i \sim G$$

$$\mathbf{x}_i \sim p(\mathbf{x}|\theta_i)$$



Clustering – A Parametric Approach

- Bayesian Gaussian Mixture Models with K mixtures
  - a distribution over classes/clusters $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$

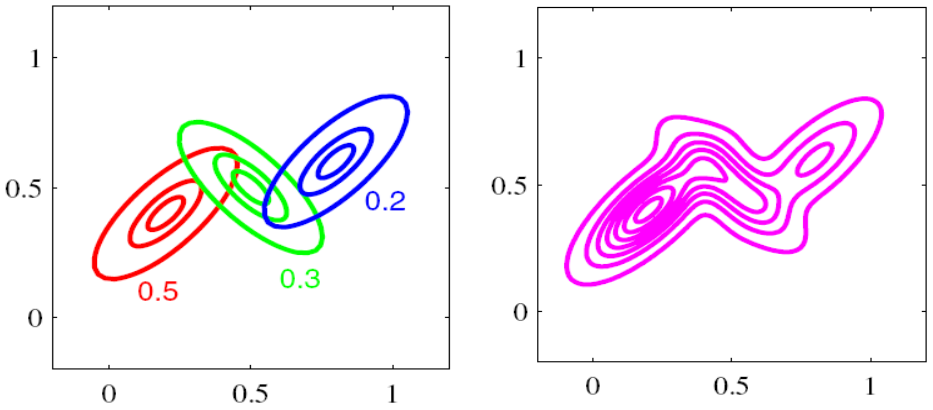
$$\boldsymbol{\pi} \sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K)$$

  - each cluster has a mean and covariance $\phi_k = (\mu_k, \Sigma_k)$

$$(\mu_k, \Sigma_k) \sim \text{Normal-Inverse-Wishart}(\nu)$$

- We still have

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)$$



[Figure credit: Bishop, 2006]

Clustering – A Parametric Approach

- Bayesian Gaussian Mixture Models with K mixtures
- The Alternative Definition
  - G is now a random measure

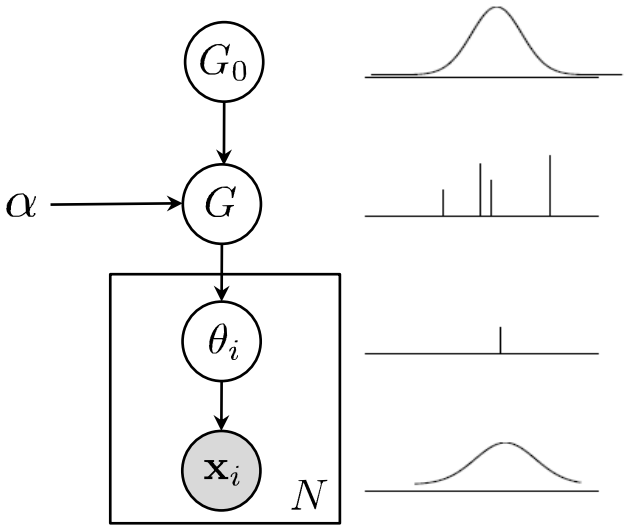
$$\phi_k \sim G_0$$

$$\boldsymbol{\pi} \sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K)$$

$$G = \sum_{k=1}^{K} \pi_k \delta_{\phi_k}$$

$$\theta_i \sim G$$

$$\mathbf{x}_i \sim p(\mathbf{x}|\theta_i)$$



Bayesian Mixture Models

- We no longer want just the maximum likelihood parameters, we want the full posterior:

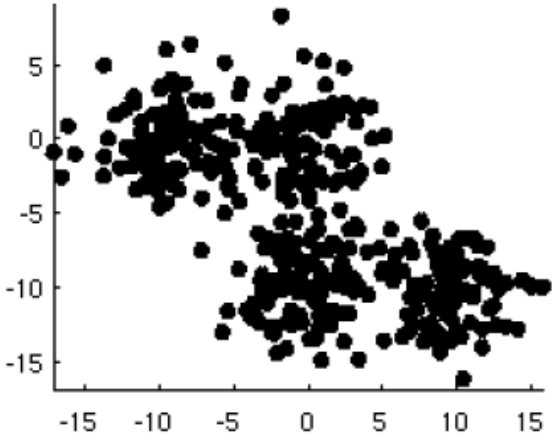
$$p(\pi, \phi|\mathcal{D}) \propto p(\mathcal{D}|\pi, \phi)p(\pi, \phi)$$

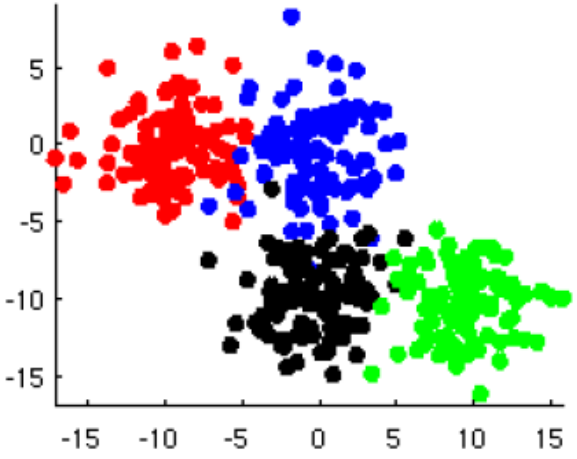
  - unfortunately, this is not analytically tractable

- Two main approaches to approximate inference
  - Markov Chain Monte Carlo (MCMC) methods
  - Variational approximations

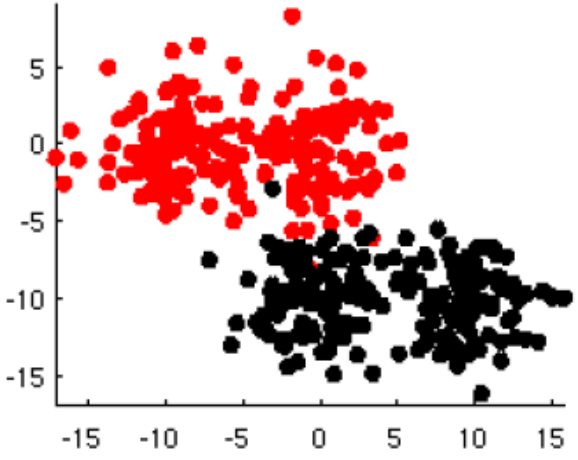
Nonparametric Clustering

- How many clusters?





OR



A Nonparametric Bayesian Approach to Clustering

- We must again specify two things:
  - The likelihood function (how data is affected by the parameters):

$$p(\mathcal{D}|\theta)$$

    Identical to the parametric case.

  - The prior (the prior distribution on the parameters):

$$p(\theta)$$

    The Dirichlet Process!

- Exact posterior inference is still intractable. But we have can derive the Gibbs update equations!

Dirichlet Process

- A flexible, nonparametric prior over an infinite number of clusters/classes as well as the parameters for those classes.

- Dirichlet Process (DP) is a distribution over distributions. We write

$$G \sim DP(\alpha, G_0)$$

to indicate G is a random distribution drawn from the DP

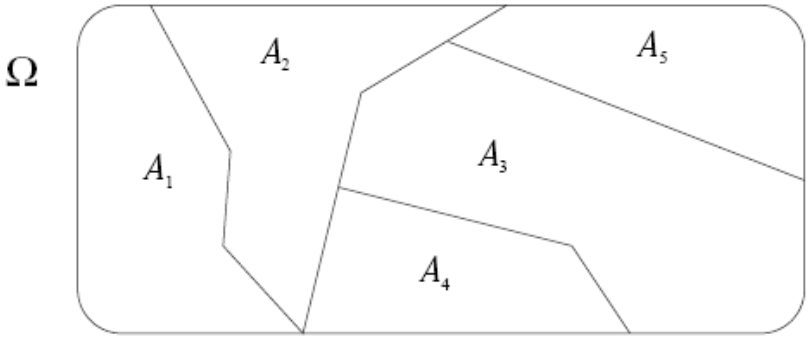
- Parameters:
  - α - the concentration parameter
  - G_0 - the base distribution. A prior for the cluster-specific parameters

Dirichlet Process

- Definition: Let G be a probability measure on the measurable space (Ω, B) and $\alpha \in \mathbb{R}_+$.

- Dirichlet Process $DP(\alpha, G_0)$ is the distribution on probability measure G such that for any finite partition $(A_1, \ldots, A_m)$ of Ω

$$(G(A_1), \ldots, G(A_m)) \sim \text{Dirichlet}(\alpha G_0(A_1), \ldots, \alpha G_0(A_m))$$



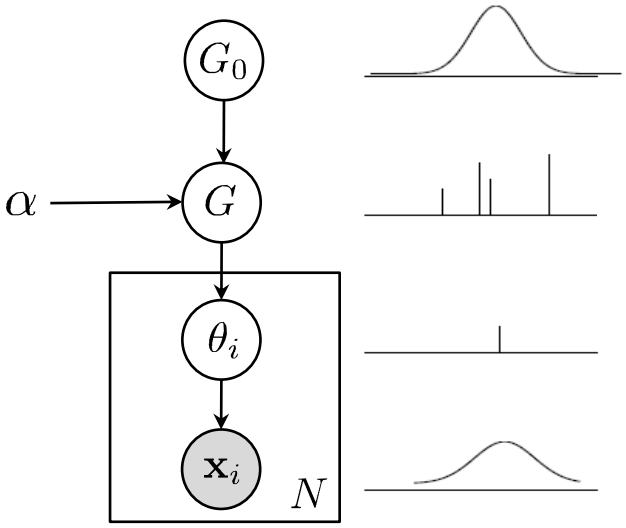
[Ferguson, Annals of Stats., 1973]

Mathematical Property of DP

- With probability 1, a sample $G \sim DP(\alpha, G_0)$ is of the form

$$G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}$$

- This is why DP can used for clustering!



[Sethuraman, Statistica Sinica, 1994]

The Stick-Breaking Process

- Define an infinite sequence of Beta random variables:

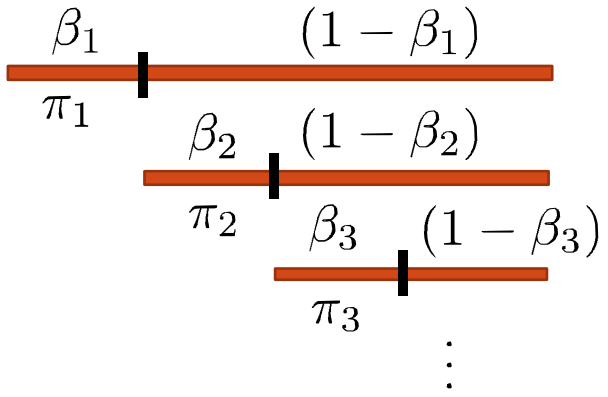
$$\beta_k \sim \text{Beta}(1, \alpha), \;\; k = 1, 2, \ldots$$

- And then define an infinite sequence of mixing proportions as:

$$\pi_1 = \beta_1$$

$$\pi_k = \beta_k \prod_{i=1}^{k-1} (1 - \beta_i), \;\; k = 2, 3, \ldots$$

- This can be viewed as breaking off portions of a stick:



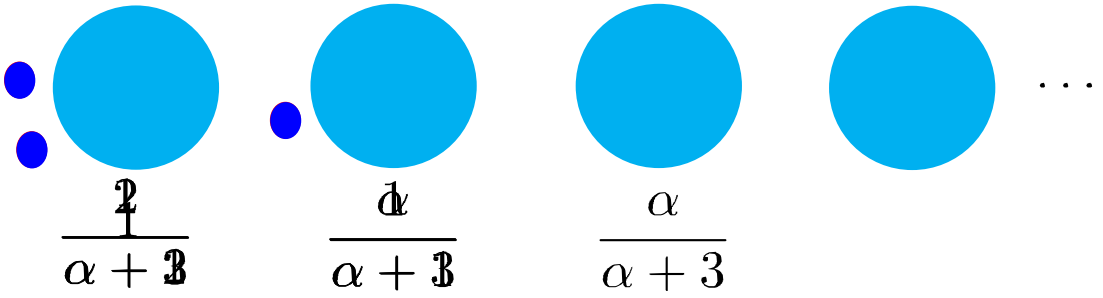
Chinese Restaurant Process (CRP)

- A random process in which n customers sit down in a Chinese restaurant with an infinite number of tables
  - first customer sits at the first table
  - the nth customer chooses a table with probability

$$p(z_i = k) = \frac{n_k}{n - 1 + \alpha}, \text{ for a pre-occupied table } k$$

$$p(z_i = k) = \frac{\alpha}{n - 1 + \alpha}, \text{ for an empty table } k$$

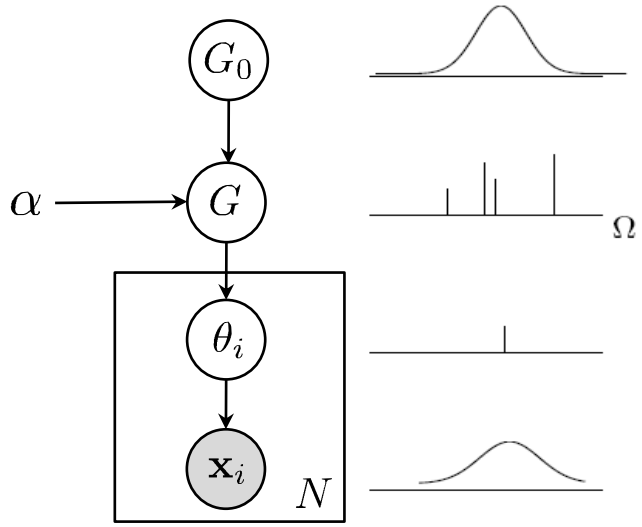
  - where n_k is the number of people sitting at table k.



Relation between CRP and DP

- The Dirichlet Process is the *De Finetti mixing distribution* for the CRP.
- That means, when we integrate out G, we get the CRP

$$p(\theta_1, \ldots, \theta_n) = \int \prod_{i=1}^{n} p(\theta_i|G) dP(G)$$



Inference for DP Mixtures – Gibbs sampler

- Use CRP representation:
  - For the component j with $n_{-i,j} > 0$

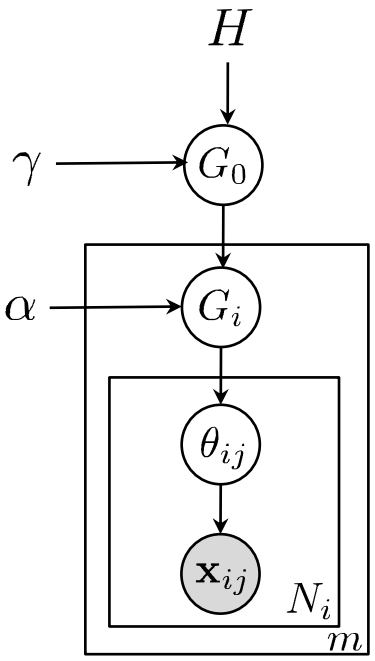
$$p(z_i = j|\mathbf{Z}_{-i}, \theta, X) \propto p(z_i = j|\mathbf{Z}_{-i}, \alpha)p(\mathbf{x}_i|\theta_j)$$
$$= \frac{n_{-i,j}}{N-1+\alpha} p(\mathbf{x}_i|\theta_j)$$

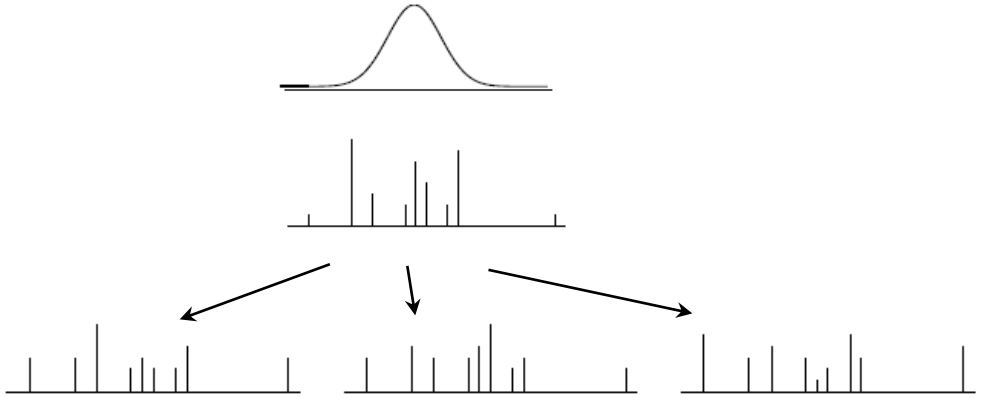
  - For a new component
    - Let $A = \{z_i \neq z_{i'} \text{ for all } i \neq i'\}$

$$p(A|\mathbf{Z}_{-i}, X) = \int p(A, \theta|\mathbf{Z}_{-i}, X)d\theta \propto p(A|\mathbf{Z}_{-i}) \int p_0(\theta)p(\mathbf{x}_i|\theta)d\theta$$
$$\propto \frac{\alpha}{N-1+\alpha} \int p(\mathbf{x}_i|\theta)p_0(\theta)d\theta$$

Extensions of DP Mixtures

- Hierarchical DP mixtures





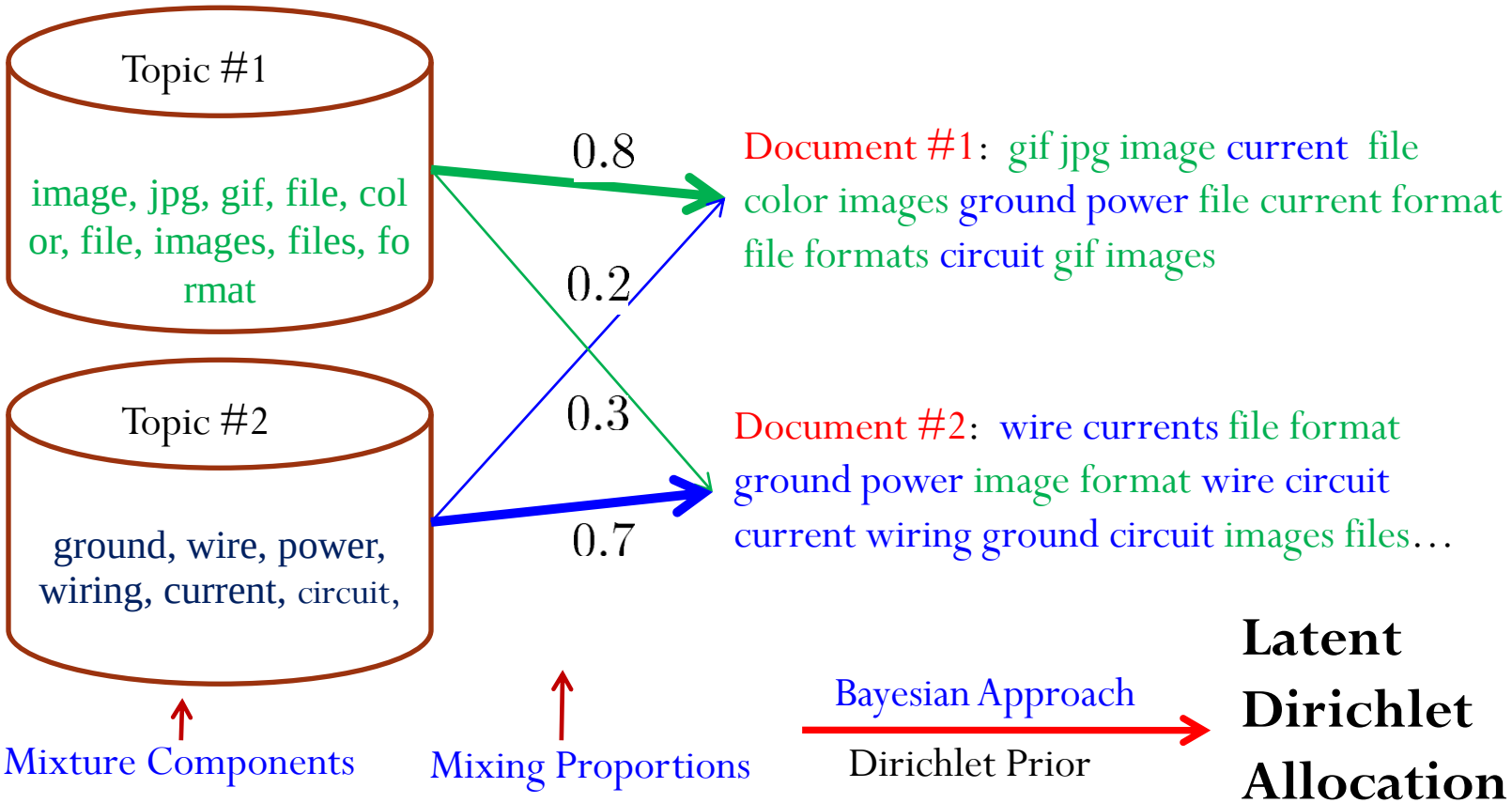
$$G_0 \sim DP(\gamma, H)$$
$$G_i \sim DP(\alpha, G_0)$$
$$\theta_{ij} \sim G_i$$
$$\mathbf{x}_{ij}|\theta_{ij} \sim p(\mathbf{x}_{ij}|\theta_{ij})$$

- Application: modeling *m* data sets from various sources about the same event

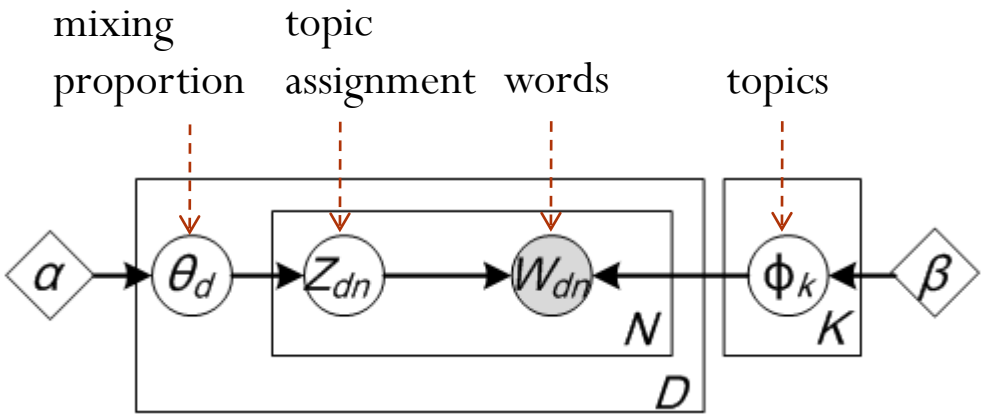
[Teh, Jordan, Beal, & Blei, JASA, 2006]

Basics of Topic Models

- A Bayesian mixture model with topical bases
- Each document is a mixture over topics; Each word is generated by one topic



Bayesian Inference for LDA



$$p(\Theta, \Phi, \mathbf{Z}, \mathbf{W}|\alpha, \beta) = \prod_{k=1}^{K} p(\Phi_k|\beta) \prod_{d=1}^{D} p(\theta_d|\alpha) \Big(\prod_{n=1}^{N} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \Phi) \Big)$$

- Given a set of documents, infer the posterior distribution

$$p(\Theta, \Phi, \mathbf{Z}|\mathbf{W}, \alpha, \beta) = \frac{p(\Theta, \Phi, \mathbf{Z}, \mathbf{W}|\alpha, \beta)}{p(\mathbf{W}|\alpha, \beta)}$$

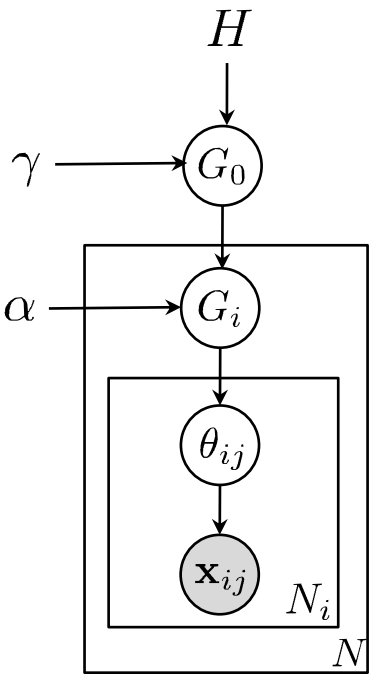
HDP Nonparametric Topic Model

- H – a measure on multinomial probability vectors, e.g., V-dimensional Dirichlet distribution

- G_0 provides a countably infinite collection of multinomial probability vectors (i.e., topics)

- G_i selects a document-specific subset of topics

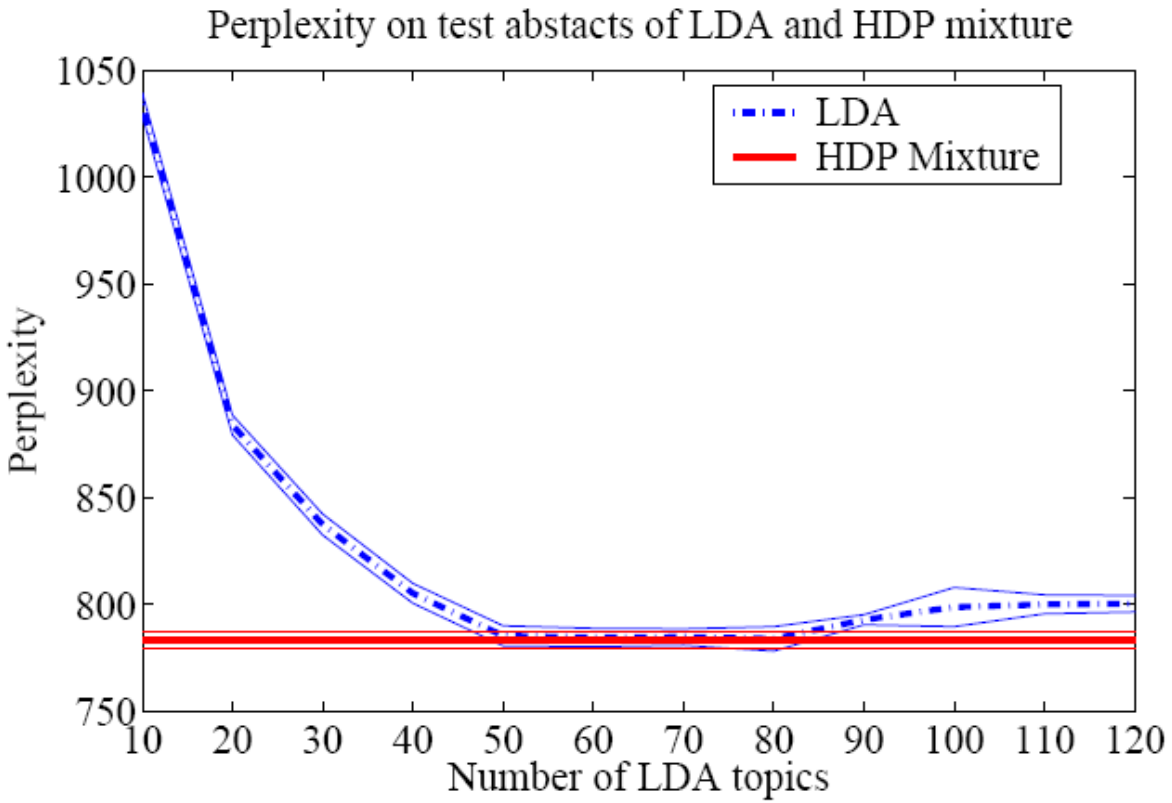
- θ_{ij} is a particular topic



[Teh, Jordan, Beal, & Blei, JASA, 2006]

Example: HDP topic model

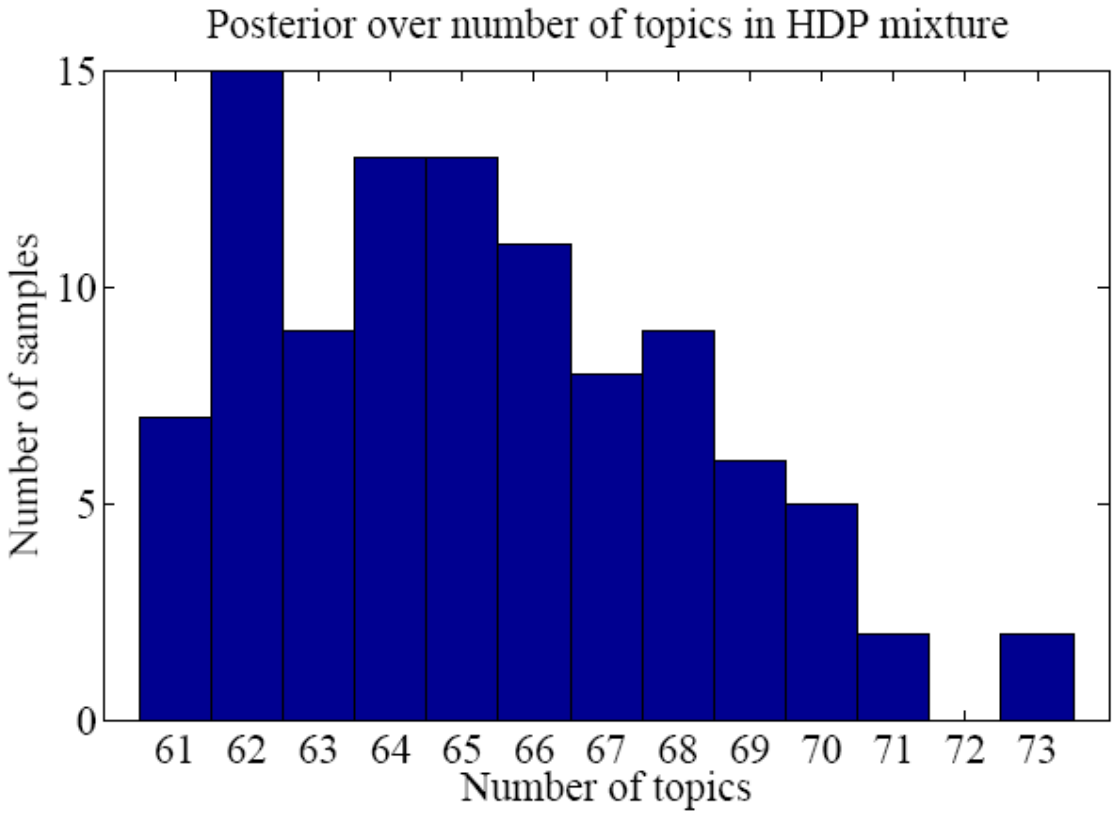
- Results on 5,838 biology abstracts



[Teh, Jordan, Beal, & Blei, JASA, 2006]

Example: HDP topic model

- Results on 5838 biology abstracts

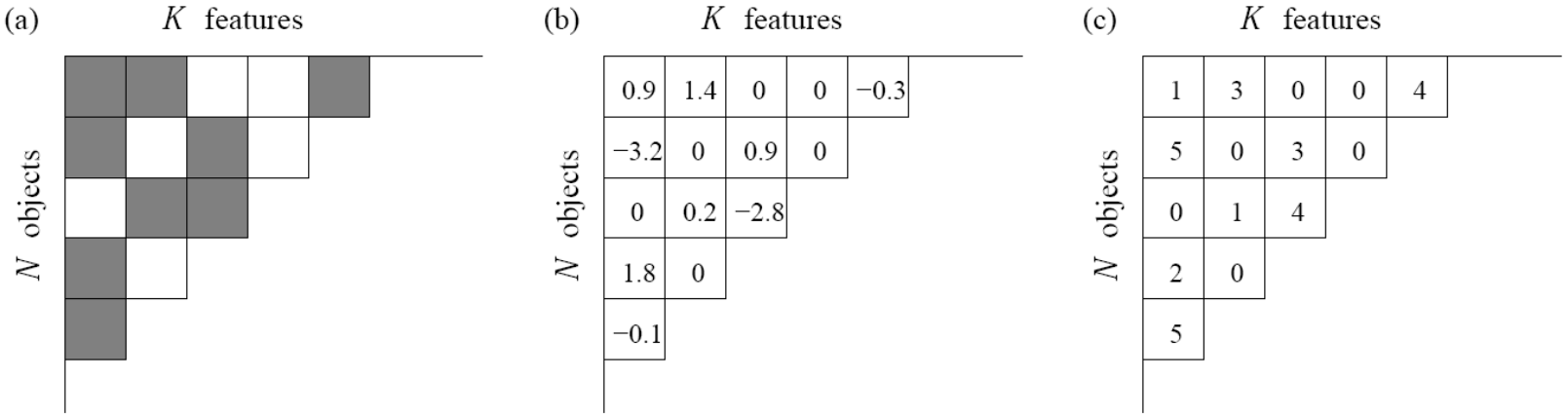


[Teh, Jordan, Beal, & Blei, JASA, 2006]

Feature Representation Learning

Latent Feature Models

- Consider N objects, the latent features form a matrix
- The feature matrix can be decomposed into two components
  - A binary matrix Z indicating which features possessed by each object
  - A matrix V indicating the value of each feature for each object



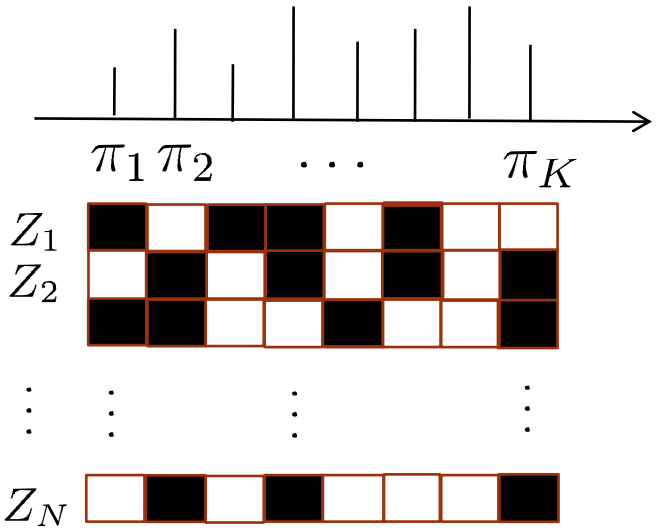
- Sparsity is imposed on the binary matrix Z
- For Bayesian, the prior can be imposed as $p(F) = p(Z)p(V)$
- We will focus on $p(Z)$, which determines the effective dimensionality of latent features

Bayesian Latent Feature Models (finite)

- A finite Beta-Bernoulli latent feature model

$$\pi_k|\alpha \sim \text{Beta}(\frac{\alpha}{K}, 1)$$

$$z_{ik}|\pi_k \sim \text{Bernoulli}(\pi_k)$$



- π_k is the relative probability of each feature being on
- $z_{i.}$ are binary vectors, giving the latent structure that's used to generate the data, e.g.,

$$\mathbf{x}_i \sim \mathcal{N}(\eta^\top z_{i.}, \delta^2)$$

A Finite Latent Feature Model

- The marginal probability of a binary matrix Z is

$$p(Z) = \prod_{k=1}^{K} \int \Big(\prod_{i=1}^{N} p(z_{ik}|\pi_k) \Big) p(\pi_k) d\pi_k$$

$$= \prod_{k=1}^{K} \int \Big(\prod_{i=1}^{N} \pi_k^{z_{ik}} (1-\pi_k)^{1-z_{ik}} \Big) p(\pi_k) d\pi_k$$

$$m_k = \sum_{i=1}^{N} z_{ik}$$

$$= \prod_{k=1}^{K} \int \pi_k^{m_k} (1-\pi_k)^{N-m_k} p(\pi_k) d\pi_k$$

$$= \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}$$

Features are independent!

$$\int_0^1 \pi^{r-1}(1-\pi)^{s-1} d\pi = \frac{\Gamma(r)\Gamma(s)}{\Gamma(r+s)}$$

A Finite Latent Feature Model

- The conditional probability of each feature assignment

$$p(z_{ik}=1|Z_{-(i,k)}) = p(z_{ik}=1|\mathbf{z}_{-(i,k)}) = \frac{p(z_{ik}=1,\mathbf{z}_{-(i,k)})}{p(\mathbf{z}_{-(i,k)})}$$

$$p(z_{ik}=1|Z_{-(i,k)}) = \frac{m_{-(i,k)}+\frac{\alpha}{K}}{N+\frac{\alpha}{N}}$$

$$p(\mathbf{z}_k) = \frac{\frac{\alpha}{K}\Gamma(m_k+\frac{\alpha}{K})\Gamma(N-m_k+1)}{\Gamma(N+1+\frac{\alpha}{K})}$$

$$m_k = \sum_{i=1}^{N} z_{ik} \qquad \Gamma(x+1) = x\Gamma(x)$$

From Finite to Infinite

- **A technical difficulty**: the probability for any particular matrix goes to zero as $K \to \infty$

$$\lim_{K\to\infty} p(Z|\alpha) = 0$$

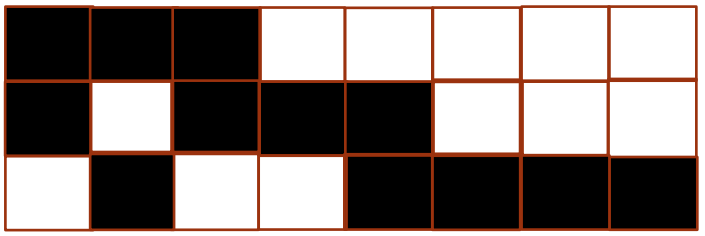
- However, if we consider **equivalence classes of matrices** in left-ordered form obtained by reordering the columns:

$$\lim_{K\to\infty} p([Z]|\alpha) = \exp\{-\alpha H_N\} \frac{\alpha^{K_+}}{\prod_{h>0} K_h!} \prod_{k\le K_+} \frac{(N-m_k)!(m_k-1)!}{N!}$$

  - K_+ is the number of features assigned (i.e. non-zero columns).
  - $H_N = \sum_{n=1}^{N} \frac{1}{n}$ is the Nth harmonic number.
  - K_h are the number of features with history h.

Indian Buffet Process

- A stochastic process on infinite binary feature matrices
- Generative procedure:
  - Customer 1 chooses the first K_1 dishes: $K_1 \sim \text{Poisson}(\alpha)$
  - Customer *i* chooses:
    - Each of the existing dishes with probability $\frac{m_k}{i}$
    - K_i additional dishes, where $K_i \sim \text{Poisson}(\frac{\alpha}{i})$



cust 1: new dishes 1-3

cust 2: old dishes 1,3; new dishes 4-5

cust 3: old dishes 2,5; new dishes 6-8

$$Z \sim \mathcal{IBP}(\alpha)$$

(Griffiths & Ghahramani, 2005)

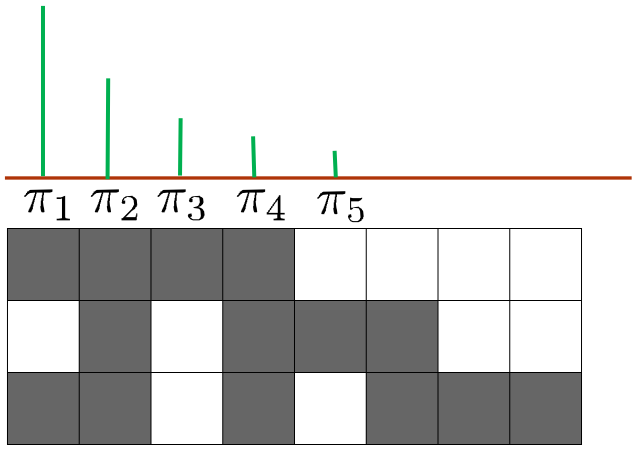
Indian Buffet Process

- A stochastic process on infinite binary feature matrices
- Stick-breaking construction: $Z_i \sim \mathcal{IBP}(\alpha)$

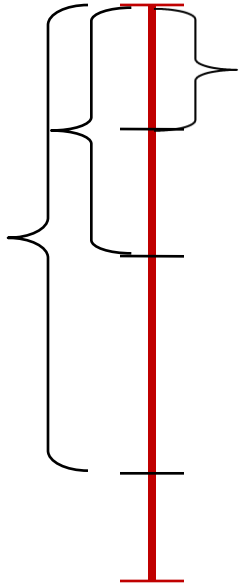
$$z_{nk} \sim \text{Bernoulli}(\pi_k)$$

$$\pi_i(\mathbf{v}) = v_i \pi_{i-1}(\mathbf{v}) = \prod_{j=1}^{i} v_j$$

$$v_i \sim \text{Beta}(\alpha, 1)$$



$\prod_{j=1}^{i-1} v_j$	v_i	π_i
0	0.8	0.8
0.8	0.5	0.4
0.4	0.4	0.16



Inference by Gibbs Sampling

- In the finite Beta-Bernoulli model, we have

$$p(z_{ik} = 1 | Z_{-(i,k)}) = \frac{m_{-(i,k)} + \frac{\alpha}{K}}{N + \frac{\alpha}{N}}$$

- Set limit $K \to \infty$, we have the conditional for infinite model

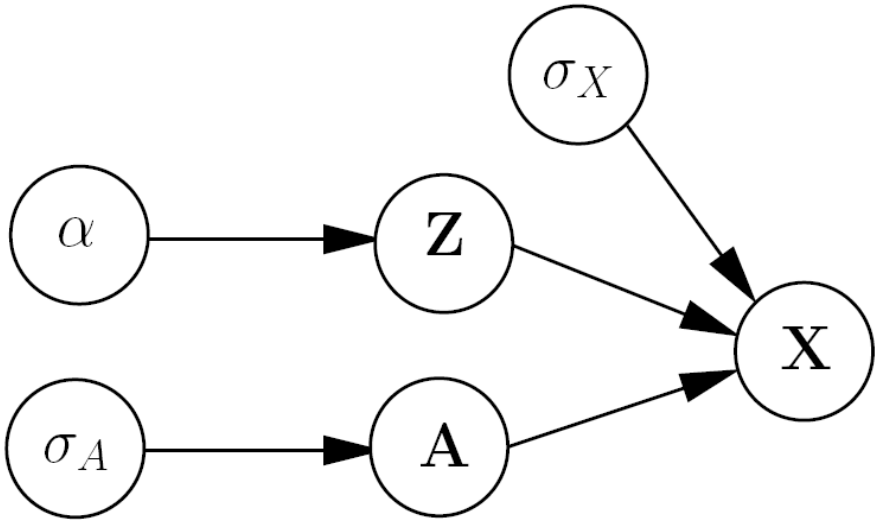
$$p(z_{ik} = 1 | Z_{-(i,k)}) = \frac{m_{-(i,k)}}{N}$$

  - for any k such that $m_{-(i,k)} > 0$
  - The number of new features should be drawn from

$$\text{Poisson}\left(\frac{\alpha}{K}\right)$$

Use with Data

- A linear-Gaussian model with binary features



- Gaussian likelihood $p(X|Z, A, \sigma_X) = \mathcal{N}\left(ZA, \sigma_X^2 I\right)$
- Gaussian prior $p(A|\sigma_A) = \mathcal{N}\left(0, \sigma_A^2 I\right)$

Inference with Gibbs Sampling

- The posterior is

$$p(Z, A|X, \alpha) \propto p(X|Z)p(Z|\alpha)$$

- The conditional for each feature assignment

$$p(z_{nk} = 1|Z_{-(n,k)}, X, \alpha) \propto p(z_{nk} = 1|Z_{-(n,k)}, \alpha)p(X|Z)$$

  - If $m_{-(i,k)} > 0$, $p(z_{ik} = 1|Z_{-(i,k)}) = \frac{m_{-(i,k)}}{N}$
  - For infinitely many k such that $m_{-(i,k)} = 0$: Metropolis steps with truncation to sample from the number of new features for each object

- For linear-Gaussian model, $p(X|Z)$ can be computed

Other Issues

- Sampling methods for non-conjugate models (Wood & Giffiths, 2007; Doshi-Velez & Ghahramani, 2009)
- Variational inference with the stick-breaking representation of IBP (Doshi-Velez et al., 2009)

$$z_{nk} \sim \text{Bernoulli}(\pi_k)$$

$$\pi_i(\mathbf{v}) = v_i \pi_{i-1}(\mathbf{v}) = \prod_{j=1}^{i} v_j \qquad v_i \sim \text{Beta}(\alpha, 1)$$

- Applications to various types of data
  - graph structures (Miller et al., 2009; Zhu, 2012)
  - dyadic data, e.g., user-movie-ratings (Meeds et al., 2006; Xu et al., 2012)
  - time series models (Gael et al., 2009)

Regression

Bayesian Regression Methods

- A noisy observation model

$$y = f(x) + \epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

- Gaussian likelihood function for linear regression $f(x) = \mathbf{w}^\top x$

$$p(\mathbf{y}|X, \mathbf{w}) = \prod_{i=1}^{n} p(y_i|x_i, \mathbf{w}) = \mathcal{N}(X^\top \mathbf{w}, \sigma_n^2 I)$$

- Gaussian prior (Conjugate)

$$\mathbf{w} \sim \mathcal{N}(0, \Sigma_p)$$

- Inference with Bayes' rule
  - Posterior

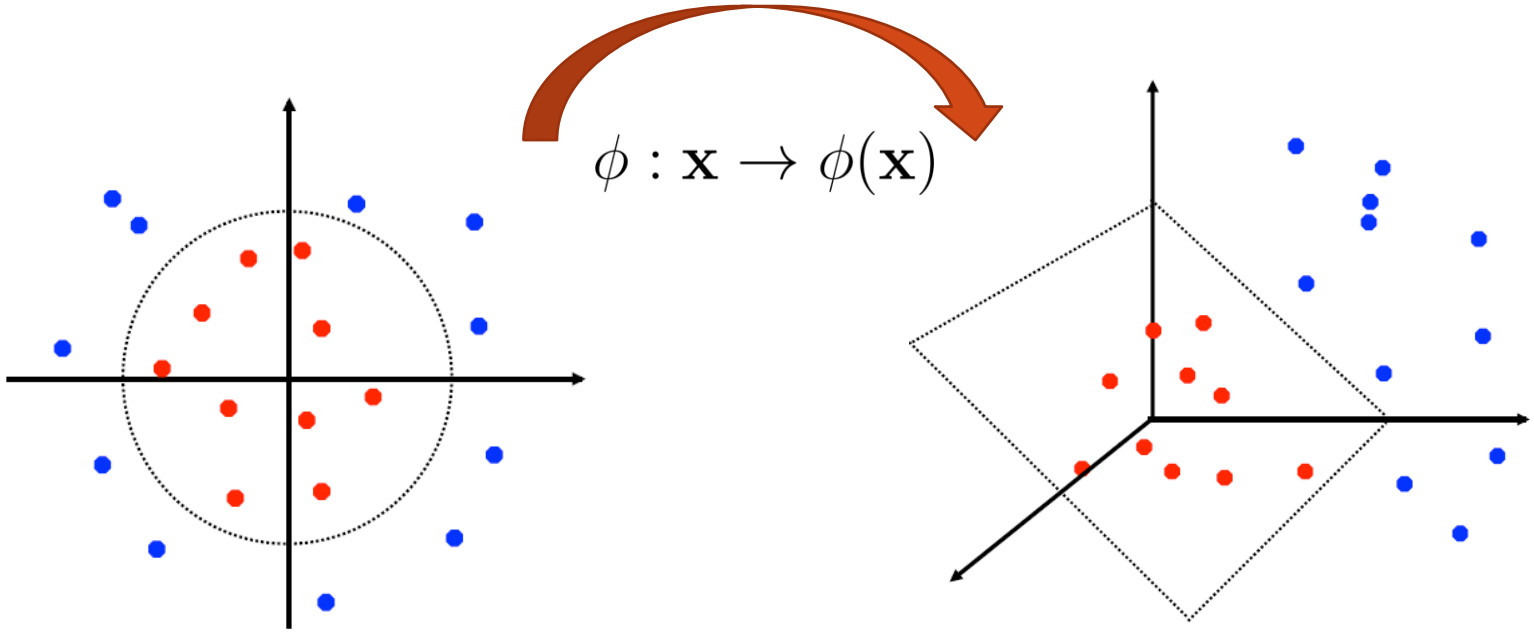
$$p(\mathbf{w}|X, \mathbf{y}) = \mathcal{N}(\frac{1}{\sigma_n^2} A^{-1} X\mathbf{y}, A^{-1}), \text{ where } A = \sigma_n^{-2} XX^\top + \Sigma_p^{-1}$$

  - Prediction

$$p(f_*|\mathbf{x}_*, X, \mathbf{y}) = \int p(f_*|\mathbf{x}_*, \mathbf{w}) p(\mathbf{w}|X, \mathbf{y}) d\mathbf{w} = \mathcal{N}(\frac{1}{\sigma_n^2} \mathbf{x}_*^\top A^{-1} X\mathbf{y}, \mathbf{x}_*^\top A^{-1} \mathbf{x}_*)$$

Generalize to Function Space

- The linear regression model can be too restricted.
- How to rescue?
- … by projections (the kernel trick)



Generalize to Function Space

- A mapping function

$$\phi : \mathcal{X} \to \mathbb{R}^N$$

- Doing linear regression in the mapped space

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}$$

- … everything is similar, with X substituted by $\phi(X)$

$$p(f_*|\mathbf{x}_*, X, \mathbf{y}) = \mathcal{N}\left(\frac{1}{\sigma_n^2}\phi(\mathbf{x}_*)^\top A^{-1}\Phi\mathbf{y}, \phi(\mathbf{x}_*)^\top A^{-1}\phi(\mathbf{x}_*)\right)$$

$$\Phi(X) = [\phi(\mathbf{x}_1) \cdots \phi(\mathbf{x}_n)] \quad A = \sigma_n^{-2}\Phi\Phi^\top + \Sigma_p^{-1}$$

Example 1: fixed basis functions

- Given a set of basis functions $\{\phi_h(\mathbf{x})\}_{h=1}^{H}$

$$\phi(\mathbf{x}) = [\phi_1(\mathbf{x}) \cdots \phi_H(\mathbf{x})]^\top$$

  - E.g. 1:

$$\phi_h(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - c_h\|_2^2}{2r^2}\right)$$

  - E.g. 2:

$$\phi_h(\mathbf{x}) = x_i^p x_j^q$$

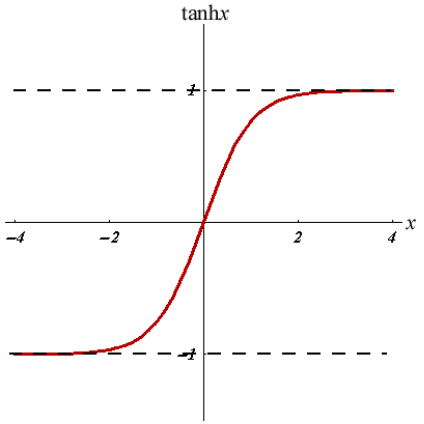
$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}$$

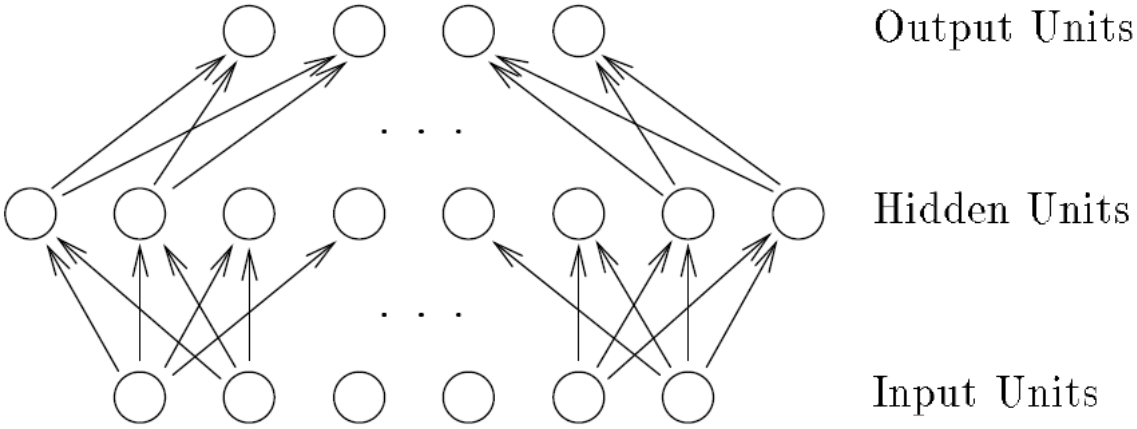
Example 2: adaptive basis functions

- Neural networks to learn a parameterized mapping function
- E.g., a two-layer feedforward neural networks

$$\phi_h(\mathbf{x}) = \tanh\left(\sum_{i=1}^{I} w_{hi}^{(1)} x_i + w_{h0}^{(1)}\right)$$

$$f(\mathbf{x};\mathbf{w}) = \sum_{h=1}^{H} w_h^{(2)} \phi_h(\mathbf{x}) + w_0^{(2)}$$





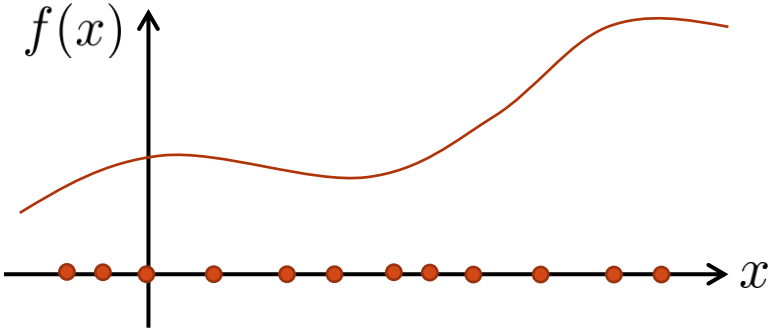
[Figure by Neal]

A Non-parametric Approach

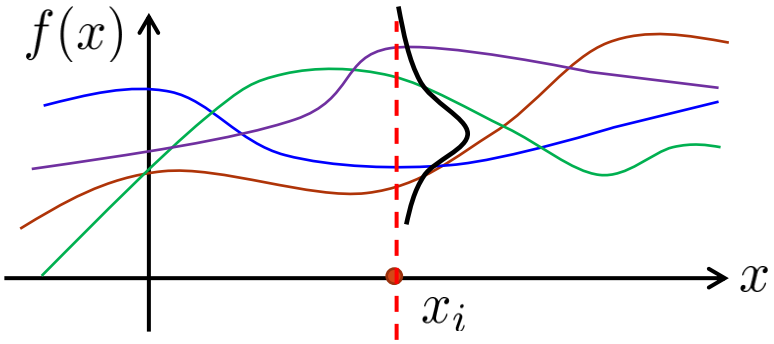
- A non-parametric approach
  - No explicit parameterization of the function
  - Put a prior over all possible functions
  - Higher probabilities are given to functions that are more likely, e.g., of good properties (smoothness, etc.)
  - Manage an uncountably infinite number of functions
  - Gaussian process provides a sophisticated approach with computational tractability

Random Function vs. Random Variable

- A function is represented as an infinite vector with an index set



- For a particular point x_i, $f(x_i)$ is a random variable



Gaussian Process

- A stochastic process is Gaussian iff for every finite set of $x_1, \ldots, x_n$,

$$(f(\mathbf{x}_1), \cdots, f(\mathbf{x}_n))$$

  is a vector-valued Gaussian random variable

- A Gaussian distribution is fully specified by the mean vector and covariance matrix

$$\mathrm{f} = (f_1, \cdots, f_n)^\top \sim \mathcal{N}(\mu, \Sigma)$$

- A Gaussian process is fully specified by a mean function and covariance function

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), \kappa(\mathbf{x}, \mathbf{x}'))$$

  - Mean function $m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$
  - Covariance function

$$\kappa(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]$$

Bayesian Linear Regression is a GP

- Bayesian linear regression with mapping functions

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w} \qquad \mathbf{w} \sim \mathcal{N}(0, \Sigma_p)$$

- The mean and covariance are

$$\mathbb{E}[f(\mathbf{x})] = \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}] = 0$$

$$\kappa(\mathbf{x}, \mathbf{x}') = \mathbb{E}[f(\mathbf{x}) f(\mathbf{x}')] = \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}\mathbf{w}^\top] \phi(\mathbf{x}') = \phi(\mathbf{x}) \Sigma_p \phi(\mathbf{x}')$$

- Therefore,

$$f(\mathbf{x}) \sim \mathcal{GP}(0, \phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}'))$$

Prediction with Noise-free Observations

- For noise-free observations, we know the true function value

$$\{(\mathbf{x}_i, f_i)\}_{i=1}^n$$

- The joint distribution of training output $\mathbf{f}$ and test outputs $\mathbf{f}_*$

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X,X) & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right)$$

$$\mathbf{f}_* | X_*, X, \mathbf{f} \sim \mathcal{N}\Big(K(X_*,X)K(X,X)^{-1}\mathbf{f}, \\ K(X_*,X_*) - K(X_*,X)K(X,X)^{-1}K(X,X_*)\Big)$$

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} A & C \\ C^\top & B \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \tilde{A} & \tilde{C} \\ \tilde{C}^\top & \tilde{B} \end{bmatrix}^{-1}\right)$$

$$\mathbf{x}|\mathbf{y} \sim \mathcal{N}(\mu_x + CB^{-1}(\mathbf{y} - \mu_y),\ A - CB^{-1}C^\top)$$

Prediction with Noisy Observations

- For noisy observations, we don't know true function values

$$\{(\mathbf{x}_i, y_i)\}_{i=1}^n \quad y_i = f(\mathbf{x}_i) + \epsilon \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

➔ $\mathrm{cov}(y_p, y_q) = k(\mathbf{x}_p, \mathbf{x}_q) + \sigma_n^2 \delta_{pq}$ or $\mathrm{cov}(\mathbf{y}) = K(X, X) + \delta_n^2 I$

- The joint distribution of training output $\mathbf{y}$ and test outputs $\mathbf{f}_*$

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X,X) + \delta_n^2 I & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right)$$

$$\mathbf{f}_* | X_*, X, \mathbf{y} \sim \mathcal{N}\Big(K(X_*, X)[K(X,X) + \delta_n^2 I]^{-1}\mathbf{y},$$
$$K(X_*, X_*) - K(X_*, X)[K(X,X) + \delta_n^2 I]^{-1} K(X, X_*)\Big)$$

Other Issues

- GPs for classification (Rasmussen & Williams, 2006)
- GPs for predictive control (Kocijan et al., 2004)
- GPs for relational learning (Chu et al., NIPS 2006)

- More information refers to the GPs Website:

http://www.gaussianprocess.org/

Overview

- **Part I (this morning, 1 hr):**
  - Basics of Bayesian methods
  - Nonparametric Bayesian methods
- **Lunch refill break**
- **Part II (this afternoon, 1.5 hr):**
  - Constrained Bayesian methods
  - Applications

1 Hour Lunch Break

Recent Advances in Bayesian Methods

Jun Zhu

dcszj@mail.tsinghua.edu.cn

Department of Computer Science and Technology

Tsinghua University

ACML 2013, Canberra, Nov 13, 2013

Outline

- Recap. of the morning session
- Constrained Bayesian inference
- Applications to
  - topic modeling
  - classification and multi-task learning
  - link prediction
  - collaborative prediction
- Large-scale Bayesian inference

Bayes' Rule

- Combining the definition of conditional prob. with the product and sum rules, we have Bayes' rule or Bayes' theorem

$$p(\mathcal{M}|X) = \frac{p(X,\mathcal{M})}{p(X)} = \frac{p(\mathcal{M})p(X|\mathcal{M})}{\int p(\mathcal{M})p(X|\mathcal{M})d\mathcal{M}}$$

Thomas Bayes (1702 – 1761)

- "*An Essay towards Solving a Problem in the Doctrine of Chances*" published at Philosophical Transactions of the Royal Society of London in 1763

Why Be Bayesian?

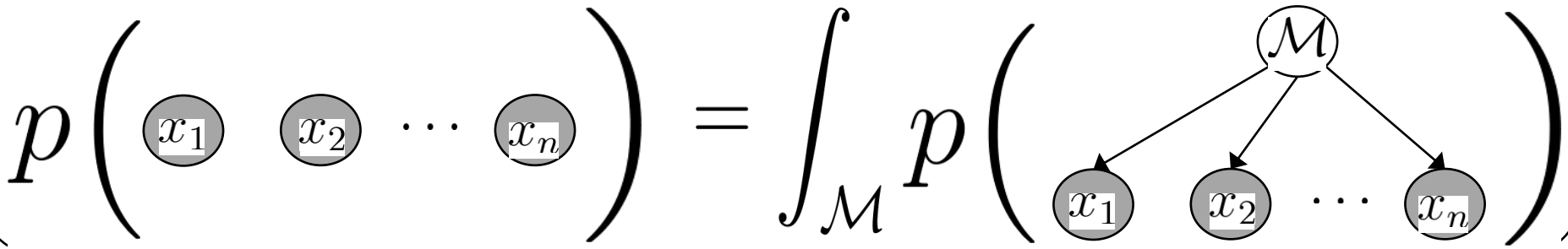
- One of many answers
- Infinite Exchangeability:

$$\forall n,\ \forall \sigma,\ p(x_1, \ldots, x_n) = p(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$$

- De Finetti's Theorem (1955): if $(x_1, x_2, \ldots)$ are *infinitely exchangeable*, then $\forall n$

$$p(x_1, \ldots, x_n) = \int \left(\prod_{i=1}^{n} p(x_i|\mathcal{M}) \right) dP(\mathcal{M})$$

for some random variable $\mathcal{M}$



Parametric Bayesian Inference

$\mathcal{M}$ is represented as a finite set of parameters θ

- A parametric likelihood: $\mathbf{x} \sim p(\cdot|\theta)$
- Prior on θ : $p(\theta)$
- Posterior distribution

$$p(\theta|\mathbf{x}) = \frac{p(\mathbf{x}|\theta)p(\theta)}{p(\mathbf{x})} \propto p(\mathbf{x}|\theta)p(\theta)$$

Examples:

- Gaussian distribution prior + Gaussian likelihood → Gaussian posterior distribution
- Dirichilet distribution prior + Multinomial likelihood → Dirichlet posterior distribution
- Sparsity-inducing priors + some likelihood models → Sparse Bayesian inference

Nonparametric Bayesian Inference

$\mathcal{M}$ is a richer model, e.g., with an infinite set of parameters

- A nonparametric likelihood: $\mathbf{x} \sim p(\cdot|\mathcal{M})$
- Prior on $\mathcal{M}$: $p(\mathcal{M})$
- Posterior distribution

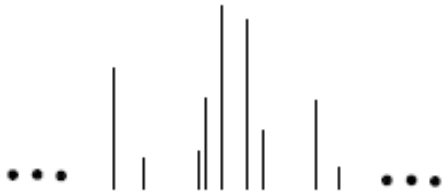
$$p(\mathcal{M}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M})p(\mathcal{M})}{p(\mathbf{x})} \propto p(\mathbf{x}|\mathcal{M})p(\mathcal{M})$$

Examples:

→ see next slide

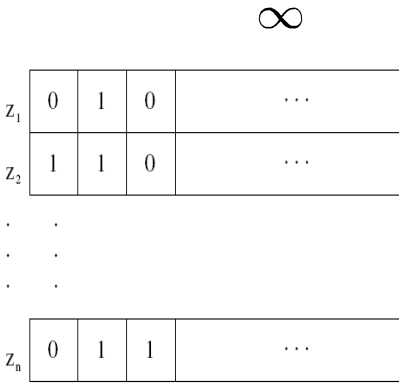
Nonparametric Bayesian Inference

probability measure



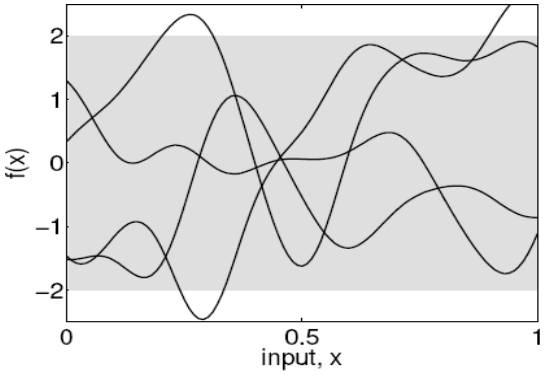
Dirichlet Process Prior [Antoniak, 1974]
+ Multinomial/Gaussian/Softmax likelihood

binary matrix



Indian Buffet Process Prior [Griffiths & Gharamani, 2005]
+ Gaussian/Sigmoid/Softmax likelihood

function

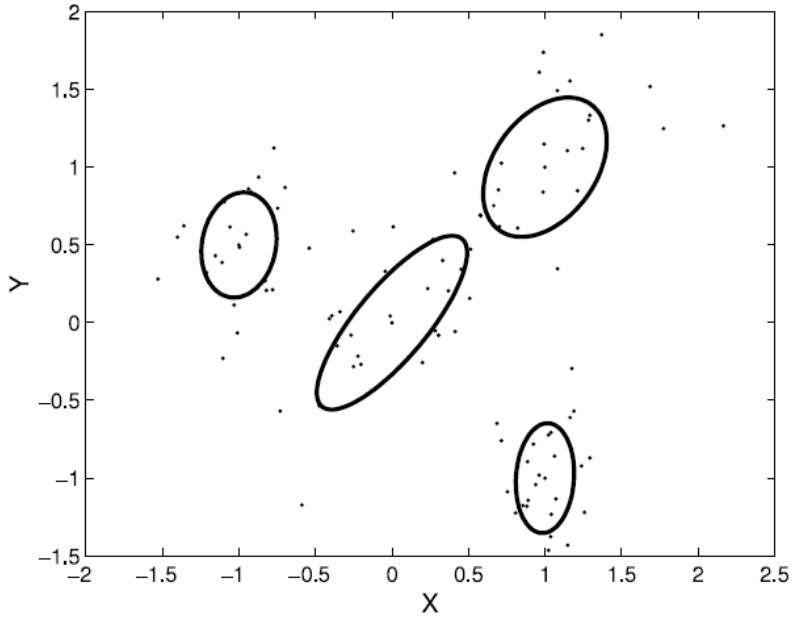


Gaussian Process Prior [Doob, 1944; Rasmussen & Williams, 2006]
+ Gaussian/Sigmoid/Softmax likelihood

Why Bayesian Nonparametrics?

Let the data speak for itself

- Bypass the model selection problem
  - let data determine model complexity (e.g., the number of components in mixture models)
  - allow model complexity to grow as more data observed



Bayesian Inference with Rich Priors

- ***The world is structured and dynamic!***
- Predictor-dependent processes to handle heterogeneous data
  - Dependent Dirichlet Process (MacEachern, 1999)
  - Dependent Indian Buffet Process (Williamson et al., 2010)
  - …
- Correlation structures to relax exchangeability:
  - Processes with hierarchical structures (Teh et al., 2007)
  - Processes with temporal or spatial dependencies (Beal et al., 2002; Blei & Frazier, 2010)
  - Processes with stochastic ordering dependencies (Hoff et al., 2003; Dunson & Peddada, 2007)
  - …

Regularized Bayesian Inference?

posterior — likelihood model — prior

$$p(\mathcal{M}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M})\pi(\mathcal{M})}{\int p(\mathbf{x}|\mathcal{M})\pi(\mathcal{M})d\mathcal{M}}$$

- **Can we directly control the posterior distributions?**
  - An extra freedom to perform Bayesian inference
  - Arguably more direct to control the behavior of models
  - Can be easier and more natural in some examples

Regularized Bayesian Inference?

posterior — likelihood model — prior

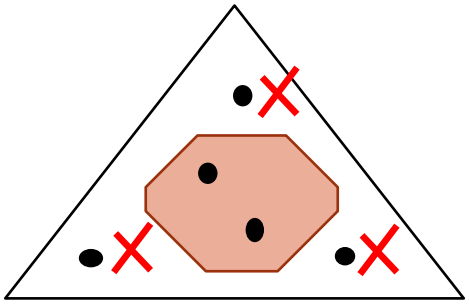
$$p(\mathcal{M}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M})\pi(\mathcal{M})}{\int p(\mathbf{x}|\mathcal{M})\pi(\mathcal{M})d\mathcal{M}}$$

- **Can we directly control the posterior distributions?**

Not obvious!

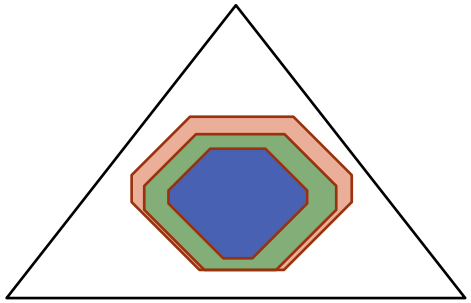
hard constraints

(A single feasible space)



soft constraints

(many feasible subspaces with different complexities/penalties)



Bayesian Inference as an Opt. Problem

Wisdom never forgets that all things have two sides

$$p(\mathcal{M}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{M})\pi(\mathcal{M})}{\int p(\mathbf{x}|\mathcal{M})\pi(\mathcal{M})d\mathcal{M}}$$

- **Bayes' rule is equivalent to solving:**

$$\min_{q(\mathcal{M})} \quad \mathrm{KL}(q(\mathcal{M})\|\pi(\mathcal{M})) - \mathbb{E}_{q(\mathcal{M})}[\log p(\mathbf{x}|\mathcal{M})]$$
$$\text{s.t.}: \quad q(\mathcal{M}) \in \mathcal{P}_{\text{prob}},$$

direct but trivial constraints on posterior distribution

[Zhu et al., NIPS 2011; Zhu et al., arXiv:1210.1766v2, 2013]

Regularized Bayesian Inference

Constraints can encode rich structures/knowledge

- **Bayesian inference with posterior regularization:**

$$\min_{q(\mathcal{M}),\xi} \quad \mathrm{KL}(q(\mathcal{M})\|\pi(\mathcal{M})) - \mathbb{E}_{q(\mathcal{M})}[\log p(\mathbf{x}|\mathcal{M})] + U(\xi)$$
$$\text{s.t.}: \quad q(\mathcal{M}) \in \mathcal{P}_{\text{post}}(\xi),$$

convex function

direct and rich constraints on posterior distribution

- Consider both hard and soft constraints
- Convex optimization problem with nice properties
- Can be effectively solved with convex duality theory

[Zhu et al., NIPS 2011; Zhu et al., arXiv:1210.1766v2, 2013]

Regularized Bayesian Inference

Constraints can encode rich structures/knowledge

- **Bayesian inference with posterior regularization:**

'unconstrained' equivalence:

$$\min_{q(\mathcal{M})} \quad \mathrm{KL}(q(\mathcal{M})\|\pi(\mathcal{M})) - \mathbb{E}_{q(\mathcal{M})}[\log p(\mathbf{x}|\mathcal{M})] + \Omega(q(\mathcal{M}))$$
$$\text{s.t.}: \quad q(\mathcal{M}) \in \mathcal{P}_{\text{prob}},$$

posterior regularization

- Consider both hard and soft constraints
- Convex optimization problem with nice properties
- Can be effectively solved with convex duality theory

[Zhu et al., NIPS 2011; Zhu et al., arXiv:1210.1766v2, 2013]

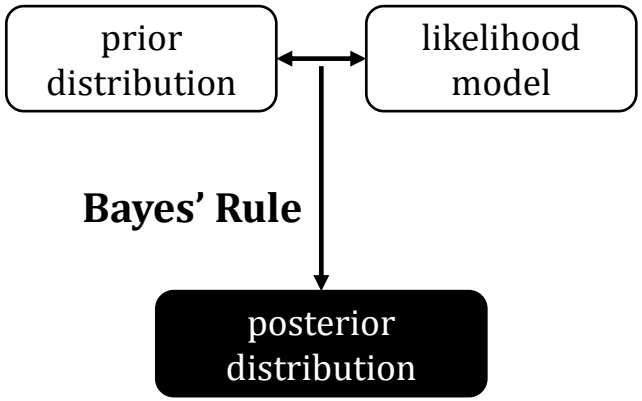
Nonconvex Posterior Regularization

- Convexity is assumed for ease of optimization, but it not the only choice

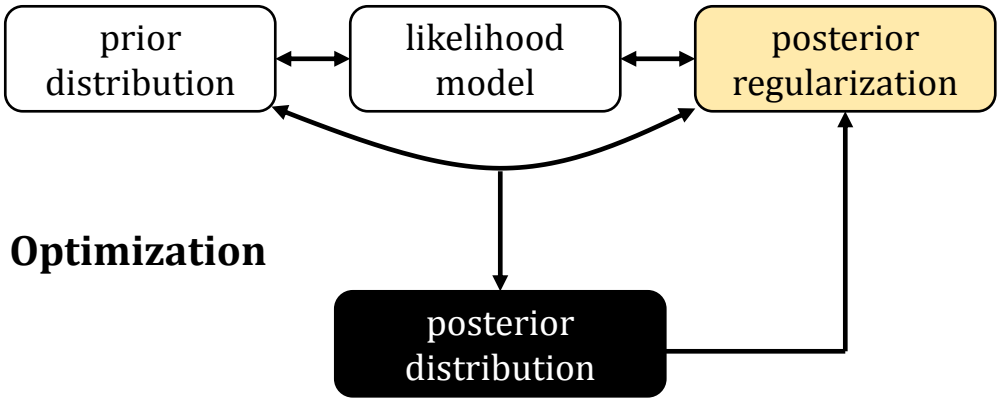
- Some recent work on nonconvex regularization, see the following paper if interested:
  - Koyejo & Ghosh, UAI 2013. Constrained Bayesian Inference for Low Rank Multitask Learning

A High-Level Comparison

Bayes:



RegBayes:



Ways to Derive Posterior Regularization

- **From learning objectives**
  - Performance of posterior distribution can be evaluated when applying it to a learning task
  - Learning objective can be formulated as Pos. Reg.

- **From domain knowledge** (ongoing & future work)
  - Elicit expert knowledge
  - E.g., logic rules

- **Others …** (ongoing & future work)
  - E.g., decision making, cognitive constraints, etc.

PAC-Bayes Theory

- Basic Setup:
  - Binary classification: $\mathbf{x} \in \mathbb{R}^d \quad y \in \mathcal{Y} = \{-1, +1\}$
  - Unknown, true data distribution: $(\mathbf{x}, y) \sim D$
  - Hypothesis space: $\mathcal{H}$
  - Risk, & Empirical Risk:

$$R(h) = \mathbb{E}_{(\mathbf{x},y)\sim D} I(h(\mathbf{x}) \neq y) \quad R_S(h) = \frac{1}{N}\sum_{n=1}^{N} I(h(\mathbf{x}_i) \neq y_i)$$

- Learn a posterior distribution Q
- Bayes/majority-vote classifier:

$$B_Q(\mathbf{x}) = \text{sgn}\left[\mathbb{E}_{h\sim Q} h(\mathbf{x})\right]$$

- Gibbs classifier
  - sample an $h \sim Q$, perform prediction

$$R(G_Q) = \mathbb{E}_{h\sim Q} R(h) \quad R_S(G_Q) = \mathbb{E}_{h\sim Q} R_S(h)$$

PAC-Bayes Theory

- Theorem (Germain et al., 2009):
  - for any distribution D ; for any set $\mathcal{H}$ of classifiers, for any prior P , for any convex function

$$\phi: \ [0,1] \times [0,1] \to \mathbb{R}$$

  - for any posterior Q , for any $\delta \in (0,1]$, the following inequality holds with a high probability ($\geq 1-\delta$)

$$\phi\left(R_S(G_Q), R(G_Q)\right) \leq \frac{1}{N}\left[\mathrm{KL}(Q\|P) + \ln\left(\frac{C(N)}{\delta}\right)\right]$$

  - where $C(N) = \mathbb{E}_{S\sim D^N}\mathbb{E}_{h\sim P}\left[e^{N\phi(R_S(h),R(h))}\right]$

RegBayes Classifiers

- PAC-Bayes theory

$$\phi\left(R_S(G_Q), R(G_Q)\right) \leq \frac{1}{N}\left[\mathrm{KL}(Q\|P) + \ln\left(\frac{C(N)}{\delta}\right)\right]$$

- RegBayes inference

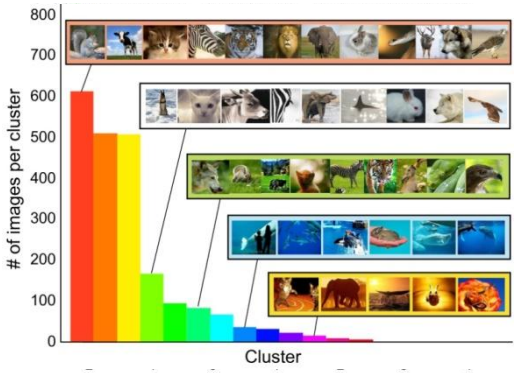
$$\begin{aligned} \min_{q(\mathcal{H})} \quad & \mathrm{KL}(q(\mathcal{H})\|p(\mathcal{H}|\mathbf{x})) + \Omega(q(\mathcal{H})) \\ \text{s.t.}: \quad & q(\mathcal{H}) \in \mathcal{P}_{\text{prob}}, \end{aligned}$$

- Observations:
  - when the posterior regularization equals to (or upper bounds) the empirical risk

$$\Omega(q(\mathcal{H})) \geq R_S(G_q)$$

  - the RegBayes classifiers tend to have PAC-Bayes guarantees.

RegBayes with Max-margin Posterior Regularization

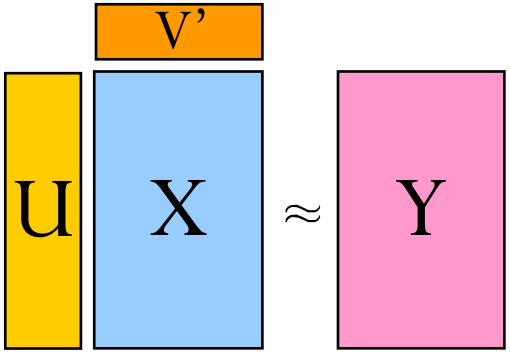


Infinite SVMs

(Zhu, Chen & Xing, ICML'11)

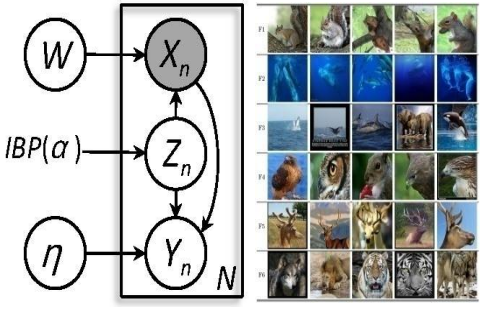
Nonparametric Max-margin Relational Models for Social Link Prediction

(Zhu, ICML'12)



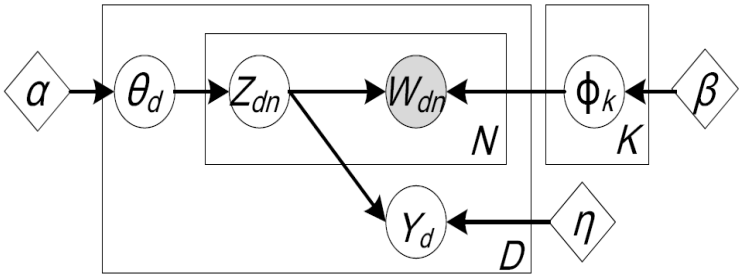
Nonparametric Max-margin Matrix Factorization

(Xu, Zhu, & Zhang, NIPS'12;
Xu, Zhu, & Zhang, ICML'13)



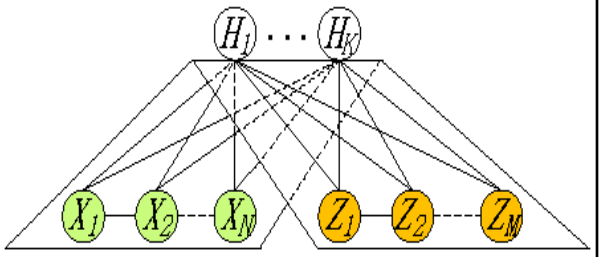
Infinite Latent SVMs

(Zhu, Chen & Xing, NIPS'11;
Zhu, Chen, & Xing, arXiv 2013)



Max-margin Topics and Fast Inference

(Zhu, Ahmed & Xing, ICML'09, JMLR'12;
Jiang, Zhu, Sun & Xing, NIPS'12;
Zhu, Chen, Perkins & Zhang, ICML'13)



Multimodal Representation Learning

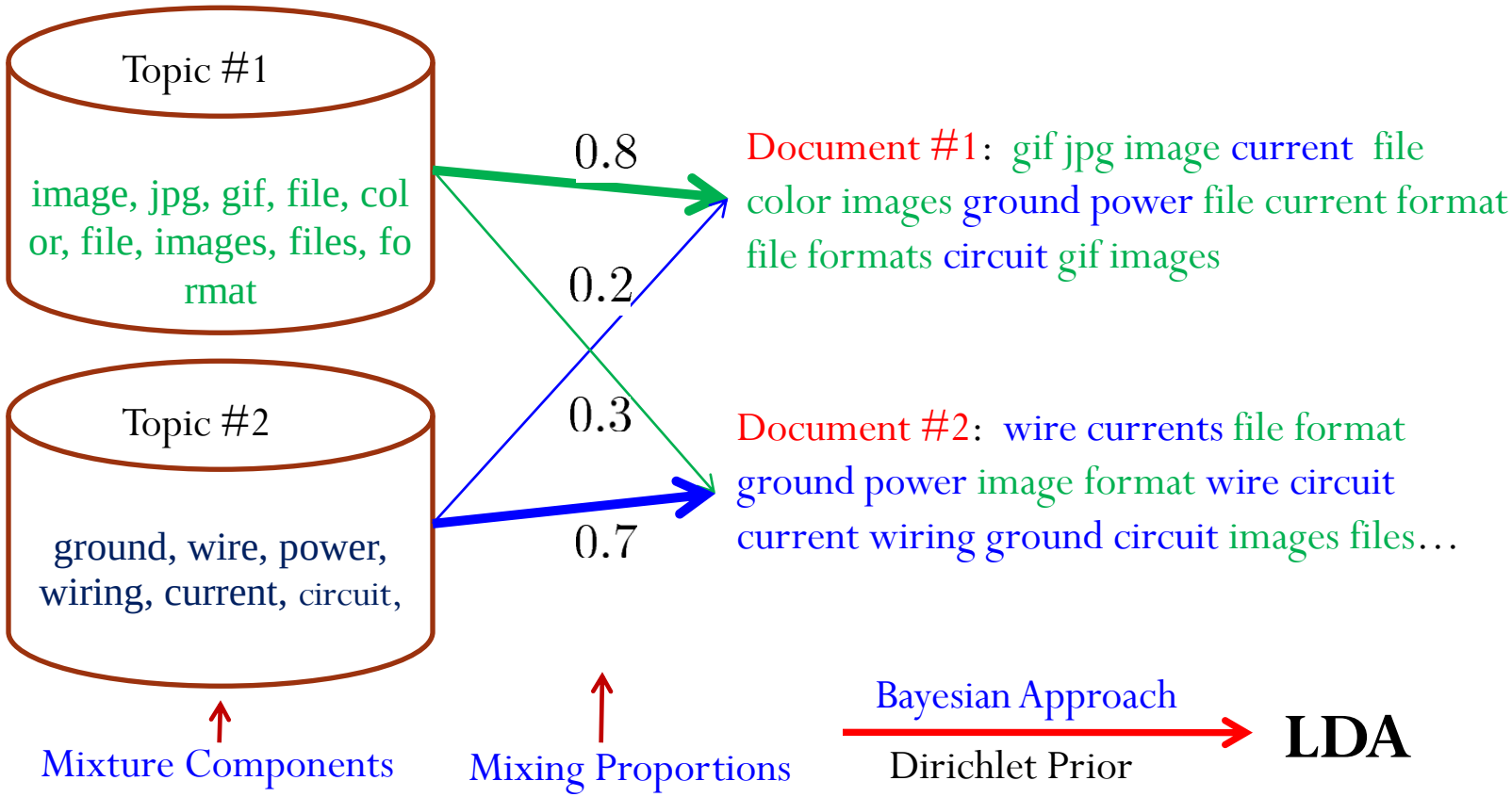
(Chen, Zhu & Xing, NIPS'10,
Chen, Zhu, Sun & Xing, PAMI'12)

*** Works from other groups are not included.**

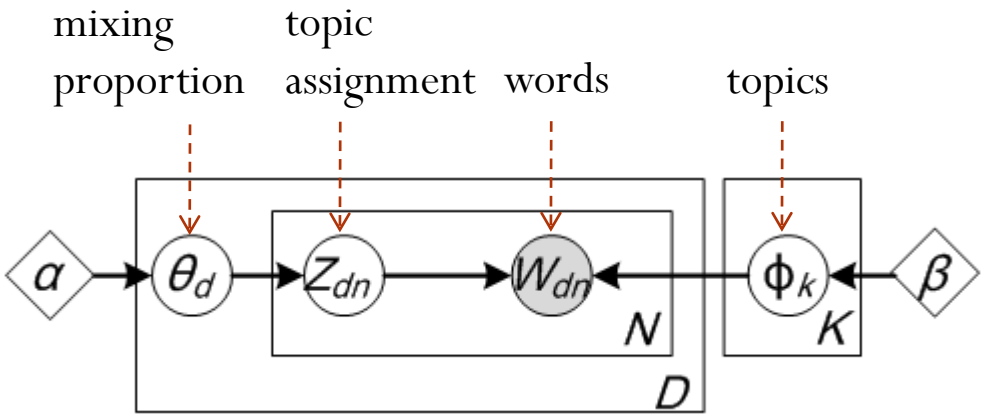
Topic Modeling

Latent Dirichlet Allocation
-- a generative story for documents

- A Bayesian mixture model with topical bases
- Each document is a random mixture over topics; Each word is generated by ONE topic



Bayesian Inference for LDA



$$p(\Theta, \Phi, \mathbf{Z}, \mathbf{W}|\alpha, \beta) = \prod_{k=1}^{K} p(\Phi_k|\beta) \prod_{d=1}^{D} p(\theta_d|\alpha) \Big(\prod_{n=1}^{N} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \Phi) \Big)$$

- Given a set of documents, infer the posterior distribution

$$p(\Theta, \Phi, \mathbf{Z}|\mathbf{W}, \alpha, \beta) = \frac{p(\Theta, \Phi, \mathbf{Z}, \mathbf{W}|\alpha, \beta)}{p(\mathbf{W}|\alpha, \beta)}$$

Optimization Problem for LDA

- Bayes' rule

$$p(\Theta, \Phi, \mathbf{Z}|\mathbf{W}, \alpha, \beta) = \frac{p(\Theta, \Phi, \mathbf{Z}|\alpha, \beta)p(\mathbf{W}|\mathbf{Z}, \Phi)}{p(\mathbf{W}|\alpha, \beta)}$$

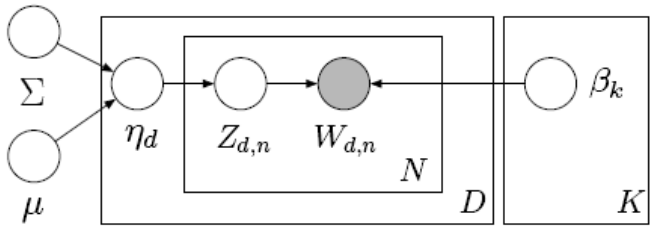
- Optimization problem

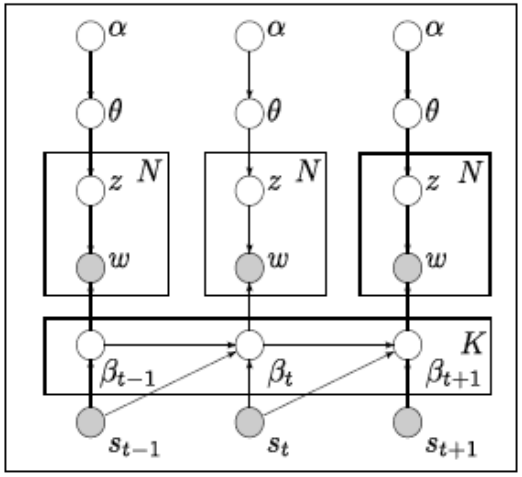
$$\min_{q(\Theta,\Phi,\mathbf{Z})} \mathrm{KL}(q(\Theta, \Phi, \mathbf{Z})||p(\Theta, \Phi, \mathbf{Z}|\alpha, \beta)) - \mathbb{E}_q[\log p(\mathbf{W}|\mathbf{Z}, \Phi)]$$
$$\text{s.t: } q(\Theta, \Phi, \mathbf{Z}) \in \mathcal{P}$$

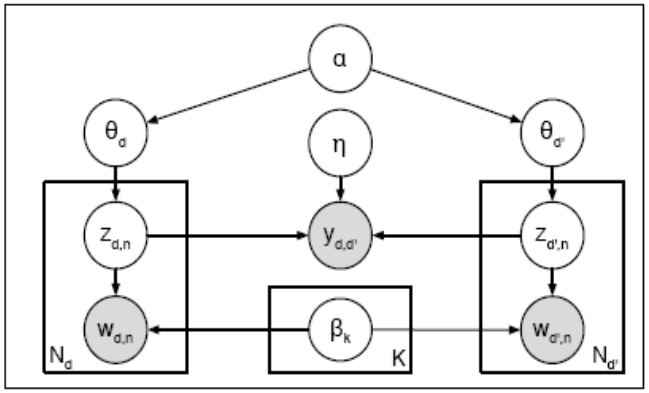
  - Assume q is in the factorized family and solve this problem with co-ordinate descent
    - ➔ variational mean-field algorithm (Blei et al., 2003)
  - Solve this problem, collapse Dirichlet variables and do Gibbs sampling
    - ➔ collapsed Gibbs sampling (Griffiths & Steyvers, 2004)

LDA has been widely extended …

- LDA can **be embedded in more complicated models**, capturing rich structures of the texts
- Extensions are either on
  - **Priors**: e.g., Markov process prior for dynamic topic models, logistic-normal prior for corrected topic models, etc
  - **Likelihood models**: e.g., relational topic models, multi-view topic models, etc.



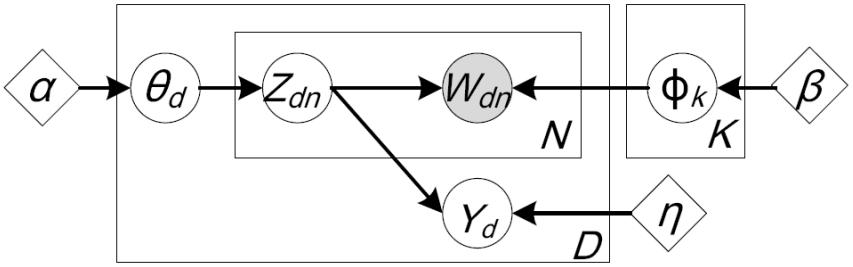




- Tutorials were provide by D. Blei at ICML, SIGKDD, etc. (http://www.cs.princeton.edu/~blei/topicmodeling.html)

Supervised LDA with Rich Likelihood

- Following the standard Bayes' way of thinking, sLDA defines a richer likelihood model



$$p(\mathbf{y}, \mathbf{W}|\mathbf{Z}, \Phi, \eta, \alpha, \beta) = p(\mathbf{y}|\mathbf{Z}, \eta)p(\mathbf{W}|\mathbf{Z}, \Phi, \alpha, \beta)$$

- per-document likelihood $y_d \in \{0, 1\}$

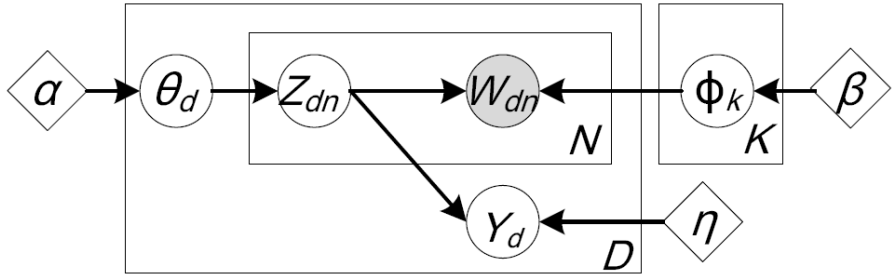
$$p(y_d|\mathbf{z}_d, \eta) = \frac{\{\exp(\eta^\top \bar{\mathbf{z}}_d)\}^{y_d}}{1 + \exp(\eta^\top \bar{\mathbf{z}}_d)} \quad \bar{z}_k = \frac{1}{N}\sum_{n=1}^{N} \mathbb{I}(z_n^k = 1)$$

- both variational and Monte Carlo methods can be developed

(Blei & McAuliffe, NIPS'07; Wang et al., CVPR'09 ; Zhu et al., ACL 2013)

Imbalance Issue with sLDA

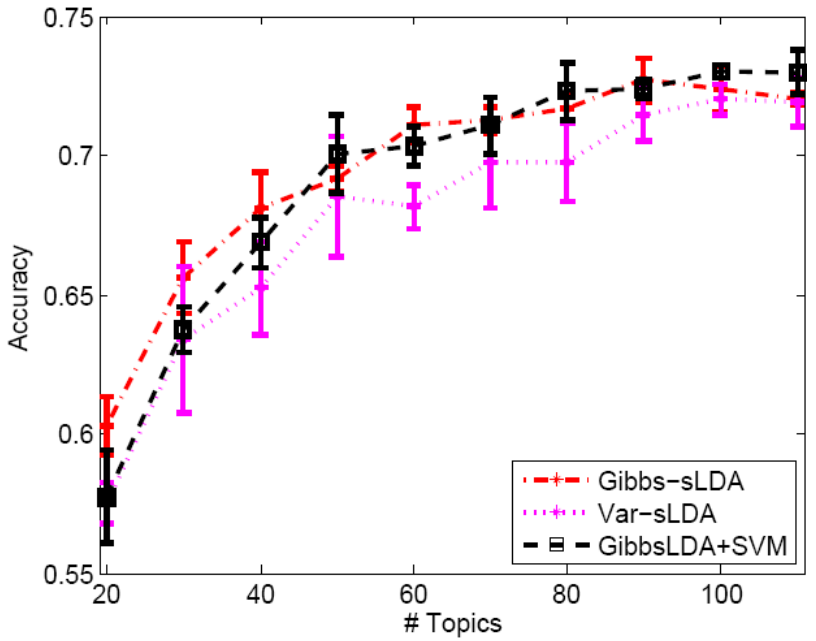
- A document has hundreds of words
- … but only one class label



- Imbalanced likelihood combination

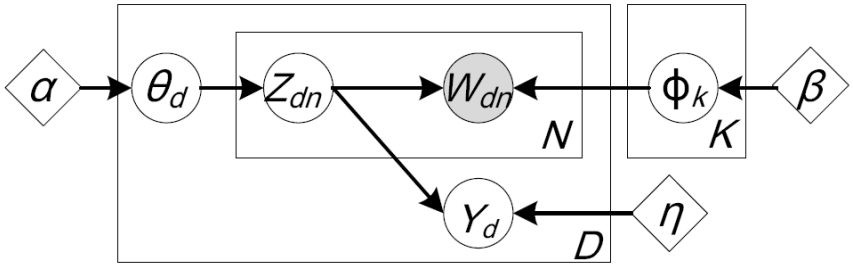
$$p(\mathbf{y}, \mathbf{W}|\mathbf{Z}, \Phi, \eta) = p(\mathbf{y}|\mathbf{Z}, \eta)p(\mathbf{W}|\mathbf{Z}, \Phi)$$

- Too weak influence from supervision



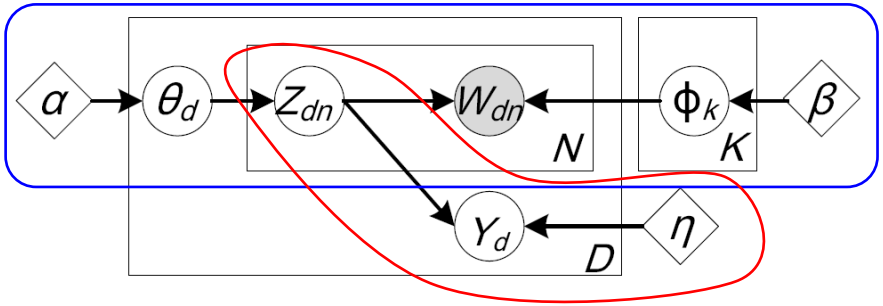
(Halpern et al., ICML 2012; Zhu et al., ACL 2013)

Max-margin Supervised Topic Models



- Can we learn supervised topic models in a max-margin way?
- How to perform posterior inference?
  - Can we do variational inference?
  - Can we do Monte Carlo?
- How to generalize to nonparametric models?

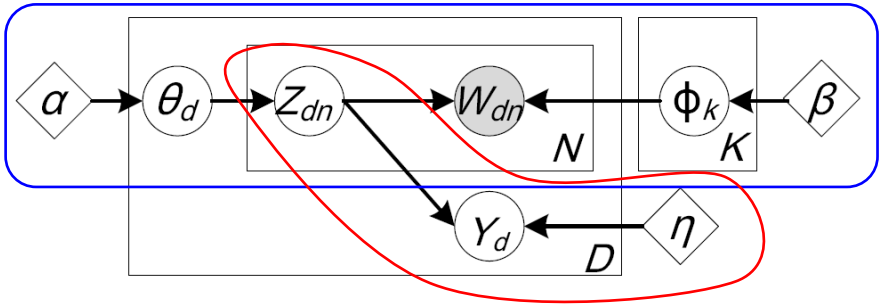
MedLDA:
Max-margin Supervised Topic Models



- Two components
  - An LDA likelihood model for describing word counts
  - An max-margin classifier for considering supervising signal

- Challenges
  - *How to consider uncertainty of latent variables in defining the classifier?*
- Nice work that has inspired our design
  - Bayes classifiers (McAllester, 2003; Langford & Shawe-Taylor, 2003)
  - Maximum entropy discrimination (MED) (Jaakkola, Marina & Jebara, 1999; Jebara's Ph.D thesis and book)

MedLDA: Max-margin Supervised Topic Models



- The averaging classifier
  - The hypothesis space is characterized by (η, Z)
  - Infer the posterior distribution

$$q(\eta, Z|\mathbf{y}, \mathbf{W})$$

  - q-weighted averaging classifier ($y_d \in \{-1, 1\}$)

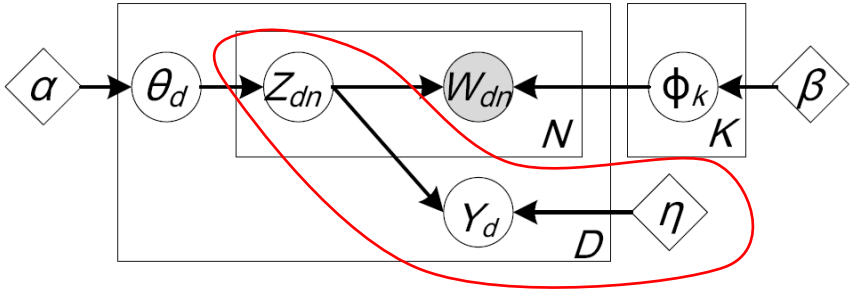
$$\hat{y} = \text{sign} f(\mathbf{w}) = \text{sign}\mathbb{E}_q[f(\eta, \mathbf{z}; \mathbf{w})]$$

  - where

$$f(\eta, \mathbf{z}; \mathbf{w}) = \eta^\top \bar{\mathbf{z}} \qquad \bar{z}_k = \frac{1}{N}\sum_{n=1}^{N} \mathbb{I}(z_n^k = 1)$$

Note: Multi-class classification can be done in many ways, 1-vs-1, 1-vs-all, Crammer & Singer's method

MedLDA:
Max-margin Supervised Topic Models



- Bayesian inference with max-margin posterior constraints

$$\min_{q(\eta,\Theta,\mathbf{Z},\Phi)\in\mathcal{P}} \mathcal{L}(q(\eta,\Theta,\mathbf{Z},\Phi)) + 2c\cdot\mathcal{R}(q)$$

  - objective for Bayesian inference in LDA

$$\mathcal{L}(q) = \mathrm{KL}(q||p_0(\eta,\Theta,\mathbf{Z},\Phi)) - \mathbb{E}_q[\log p(\mathbf{W}|\mathbf{Z},\Phi)]$$

  - posterior regularization is the hinge loss

$$\mathcal{R}(q) = \sum_d \max(0, 1 - y_d f(\mathbf{w}_d))$$

Inference Algorithms

- Regularized Bayesian Inference

$$\min_{q(\eta,\Theta,\mathbf{Z},\Phi)\in\mathcal{P}} \mathcal{L}(q(\eta,\Theta,\mathbf{Z},\Phi)) + 2c\cdot\mathcal{R}(q)$$

- An iterative procedure with $q(\eta,\Theta,\mathbf{Z},\Phi) = q(\eta)q(\Theta,\mathbf{Z},\Phi)$

$$\min_{q(\eta),\xi} \quad \mathrm{KL}(q(\eta)\|p_0(\eta)) + c\sum_d \xi_d$$

$$\forall d,\ \text{s.t.}: \quad y_d\mathbb{E}_q[\eta]^\top\mathbb{E}_q[\bar{\mathbf{z}}_d] \geq 1-\xi_d.$$

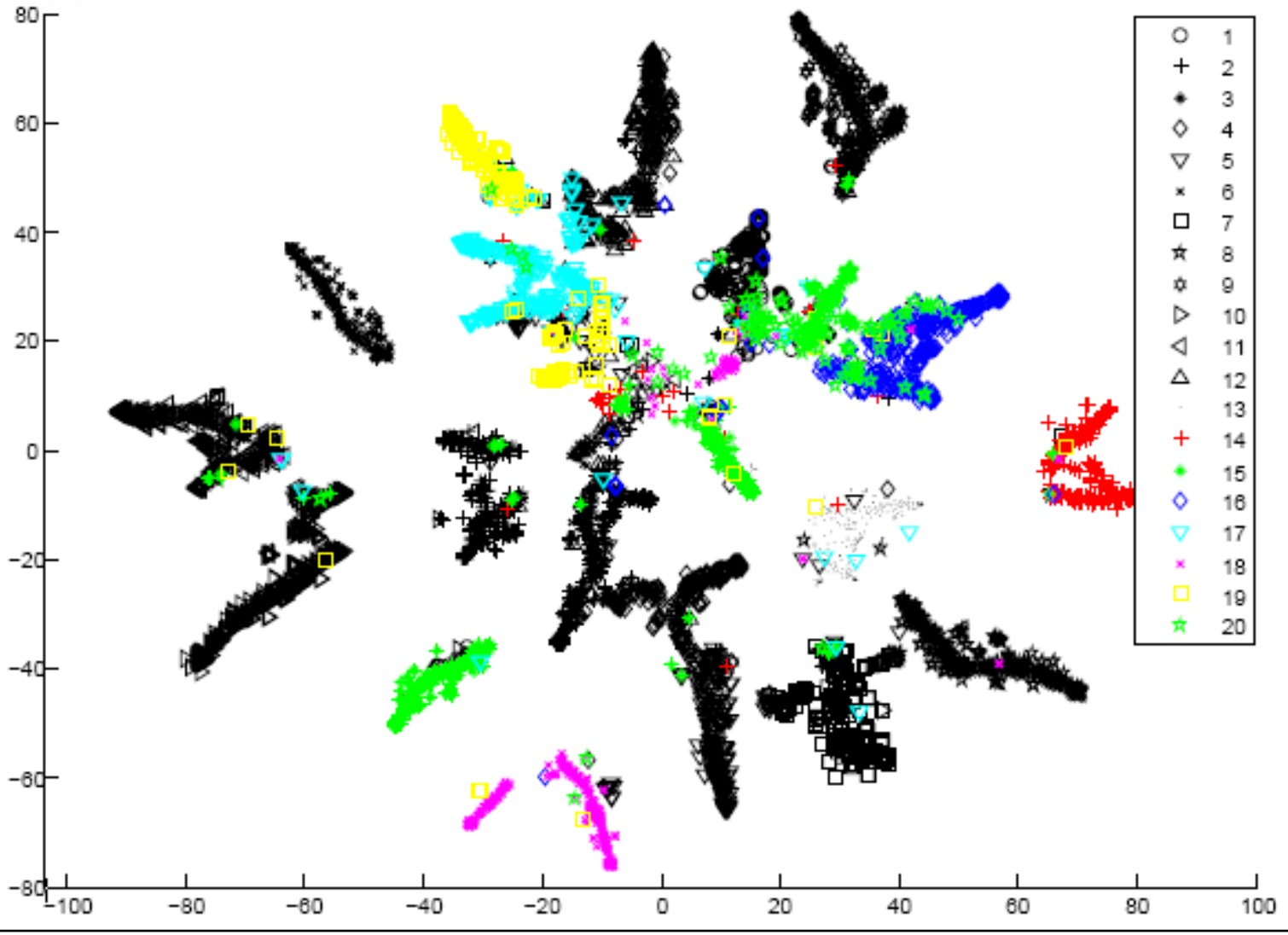
A SVM problem with a normal prior

$$\min_{q(\Theta,\mathbf{Z},\Phi),\xi} \mathcal{L}(q(\Theta,\mathbf{Z},\Phi)) + c\sum_d \xi_d$$

$$\forall d,\ \text{s.t.}: y_d\mathbb{E}_q[\eta]^\top\mathbb{E}_q[\bar{\mathbf{z}}_d] \geq 1-\xi_d.$$

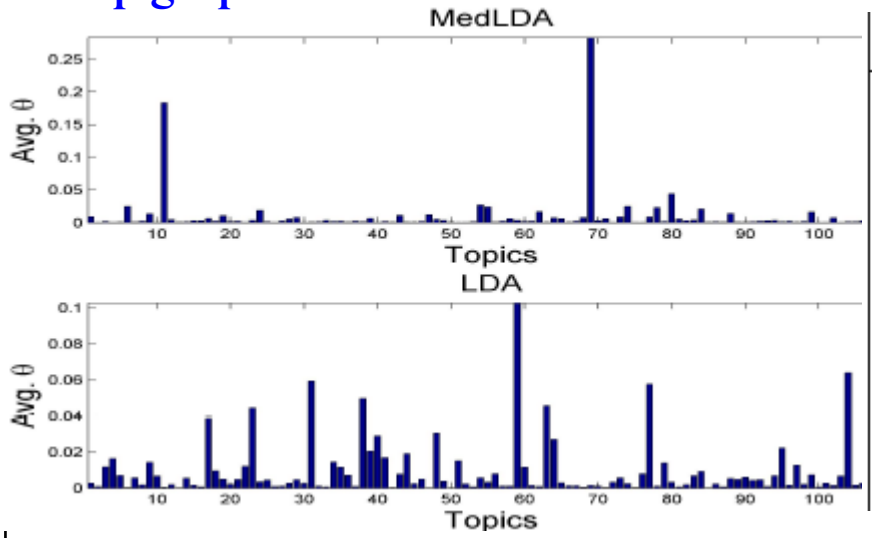
Variational approximation or Monte Carlo methods

Empirical Results on 20Newsgroups



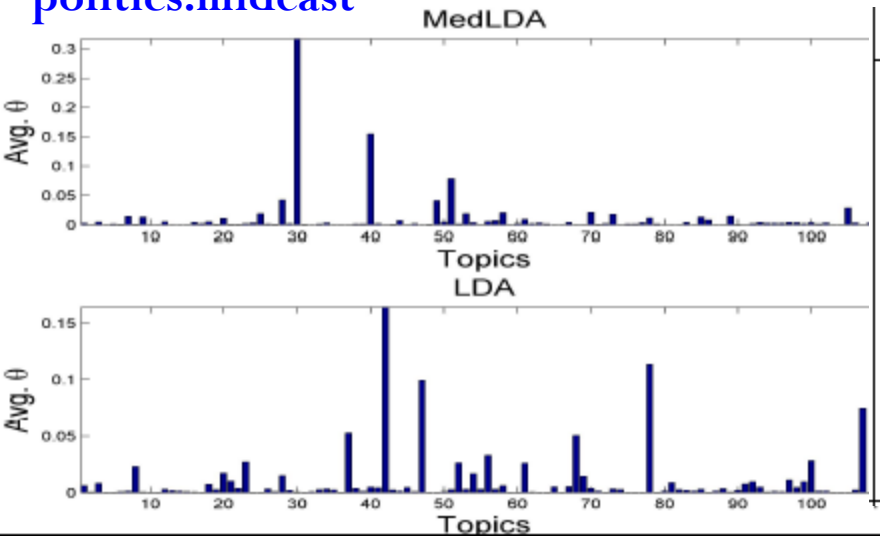
Sparser and More Salient Representations

comp.graphics



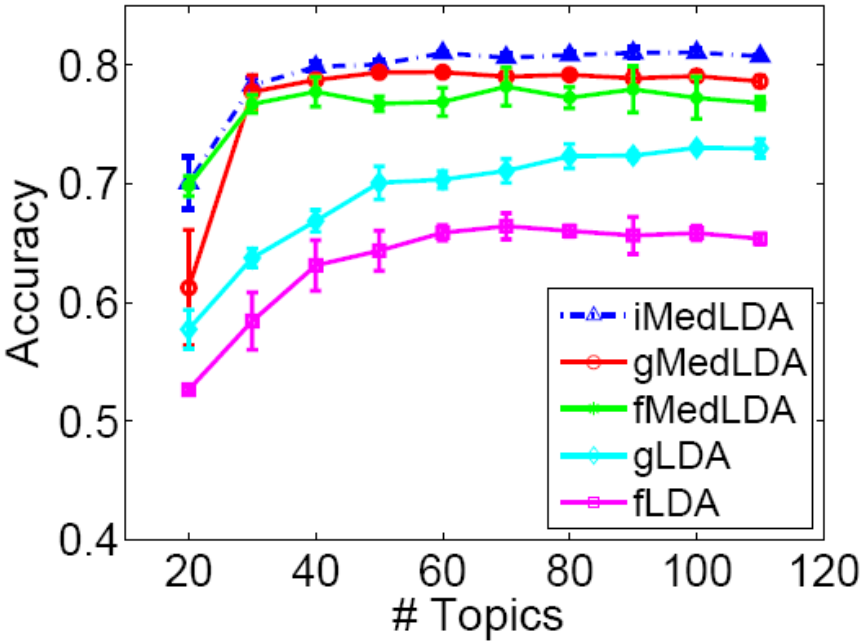
MedLDA			LDA		
T 69	T 11	T 80	T 59	T 104	T 31
image	graphics	db	image	ftp	card
jpeg	image	key	jpeg	pub	monitor
gif	data	chip	color	graphics	dos
file	ftp	encryption	file	mail	video
color	software	clipper	gif	version	apple
files	pub	system	images	tar	windows
bit	mail	government	format	file	drivers
images	package	keys	bit	information	vga
format	fax	law	files	send	cards
program	images	escrow	display	server	graphics

politics.mideast



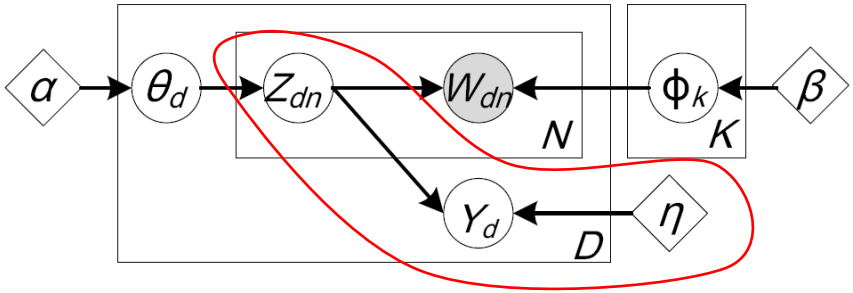
T 30	T 40	T 51	T 42	T 78	T 47
israel	turkish	israel	israel	jews	armenian
israeli	armenian	lebanese	israeli	jewish	turkish
jews	armenians	israeli	peace	israel	armenians
arab	armenia	lebanon	writes	israeli	armenia
writes	people	people	article	arab	turks
people	turks	attacks	arab	people	genocide
article	greek	soldiers	war	arabs	russian
jewish	turkey	villages	lebanese	center	soviet
state	government	peace	lebanon	jew	people
rights	soviet	writes	people	nazi	muslim

Multi-class Classification with Crammer & Singer's Approach



- Observations:
  - Inference algorithms affect the performance;
  - Max-margin learning improves a lot

Gibbs MedLDA



- The Gibbs classifier
  - The hypothesis space is characterized by (η, Z)
  - Infer the posterior distribution

$$q(\eta, Z|\mathbf{y}, \mathbf{W})$$

  - *A Gibbs* classifier

$$\hat{y}|_{\eta,\mathbf{z}} = \text{sign} f(\eta, \mathbf{z}; \mathbf{w}), \text{ where } (\eta, \mathbf{z}) \sim q(\eta, Z|\mathbf{y}, W)$$

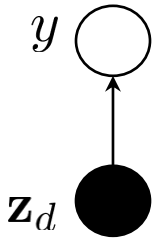
  - where $f(\eta, \mathbf{z}; \mathbf{w}) = \eta^\top \bar{\mathbf{z}} \qquad \bar{z}_k = \frac{1}{N}\sum_{n=1}^{N} \mathbb{I}(z_n^k = 1)$

(Zhu, Chen, Perkins, Zhang, ICML 2013; arXiv:1310.2816, 2013)

Gibbs MedLDA

- The "pseudo-observed" classifier if $(\eta, \mathbf{z})$ are given

$$\hat{y}|_{\eta,\mathbf{z}} = \text{sign} f(\eta, \mathbf{z}; \mathbf{w})$$



  - empirical training error

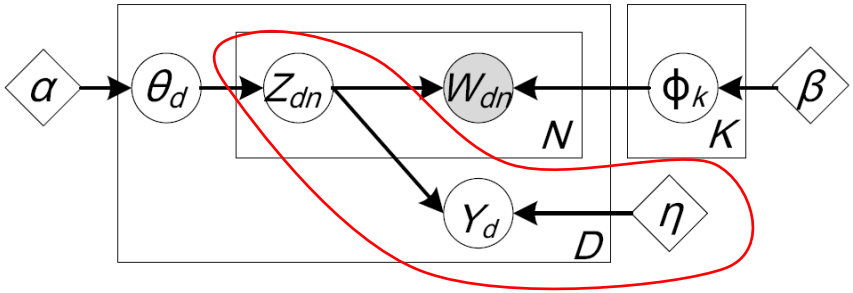
$$\hat{R}(\eta, Z) = \sum_{d=1}^{D} \mathbb{I}(\hat{y}_d|_{\eta,\mathbf{z}_d} \neq y_d)$$

  - A good convex surrogate loss is the hinge loss (an upper bound)

$$\mathcal{R}(\eta, \mathbf{Z}) = \sum_{d=1}^{D} \max(0, \zeta_d), \text{ where } \zeta_d = 1 - y_d \eta^\top \bar{\mathbf{z}}_d$$

- Now the question is how to consider the uncertainty?
  - A Gibbs classifier takes the expectation!

Gibbs MedLDA



- Bayesian inference with max-margin posterior constraints

$$\min_{q(\eta,\Theta,\mathbf{Z},\Phi)\in\mathcal{P}} \mathcal{L}(q(\eta,\Theta,\mathbf{Z},\Phi)) + 2c \cdot \mathcal{R}'(q)$$

  - an upper bound of the expected training error (empirical risk)

$$\mathcal{R}'(q) = \sum_{d=1}^{D} \mathbb{E}_q[\max(0,\zeta_d)] \geq \sum_{d} \mathbb{E}_q[\mathbb{I}(\hat{y}_d \neq y_d)]$$

Gibbs MedLDA vs. MedLDA

- The MedLDA problem

$$\min_{q(\eta,\Theta,\mathbf{Z},\Phi)\in\mathcal{P}} \mathcal{L}(q(\eta,\Theta,\mathbf{Z},\Phi)) + 2c\cdot\mathcal{R}(q)$$

$$\mathcal{R}(q) = \sum_{d} \max(0, 1 - y_d f(\mathbf{w}_d))$$

- Applying Jensen's Inequality, we have

$$\mathcal{R}'(q) \geq \mathcal{R}(q)$$

  - Gibbs MedLDA can be seen as a relaxation of MedLDA

Gibbs MedLDA

- The problem

$$\min_{q(\eta,\Theta,\mathbf{Z},\Phi)\in\mathcal{P}} \mathcal{L}(q(\eta,\Theta,\mathbf{Z},\Phi)) + 2c\cdot\mathcal{R}(q)$$

- Solve with Lagrangian methods

$$q(\eta,\Theta,\mathbf{Z},\Phi) = \frac{p_0(\eta,\Theta,\mathbf{Z},\Phi)p(\mathbf{W}|\mathbf{Z},\Phi)\phi(\mathbf{y}|\mathbf{Z},\eta)}{\psi(\mathbf{y},\mathbf{W})}$$

  - The pseudo-likelihood $\phi(\mathbf{y}|\mathbf{Z},\eta) = \prod_d \phi(y_d|\eta,\mathbf{z}_d)$

$$\phi(y_d|\mathbf{z}_d,\eta) = \exp\{-2c\max(0,\zeta_d)\}$$

Gibbs MedLDA

- **Lemma** [Scale Mixture Rep.] (Polson & Scott, 2011):
  - The pseudo-likelihood can be expressed as

$$\phi(y_d|\mathbf{z}_d,\eta) = \int_0^\infty \frac{1}{\sqrt{2\pi\lambda_d}} \exp\Big(-\frac{(\lambda_d + c\zeta_d)^2}{2\lambda_d}\Big) d\lambda_d$$

- What does the lemma mean?
  - It means:

$$q(\eta,\Theta,\mathbf{Z},\Phi) = \int q(\eta,\lambda,\Theta,\mathbf{Z},\Phi)d\lambda$$

$$\text{where } q(\eta,\lambda,\Theta,\mathbf{Z},\Phi) = \frac{p_0(\eta,\Theta,\mathbf{Z},\Phi)p(\mathbf{W}|\mathbf{Z},\Phi)\phi(\mathbf{y},\lambda|\mathbf{Z},\eta)}{\psi(\mathbf{y},\mathbf{W})}$$

$$\phi(\mathbf{y},\lambda|\mathbf{Z},\eta) = \prod_d \frac{1}{\sqrt{2\pi\lambda_d}} \exp\left(-\frac{(\lambda_d + c\zeta_d)^2}{2\lambda_d}\right)$$

A Gibbs Sampling Algorithm

- Infer the joint distribution

$$q(\eta, \lambda, \Theta, \mathbf{Z}, \Phi) = \frac{p_0(\eta, \Theta, \mathbf{Z}, \Phi) p(\mathbf{W}|\mathbf{Z}, \Phi) \phi(\mathbf{y}, \lambda|\mathbf{Z}, \eta)}{\psi(\mathbf{y}, \mathbf{W})}$$

- A Gibbs sampling algorithm iterates over:
  - Sample $\eta^{t+1} \sim q(\eta|\lambda^t, \Theta^t, \mathbf{Z}^t, \Phi^t) \propto p_0(\eta)\phi(\mathbf{y}, \lambda^t|\mathbf{Z}^t, \eta)$
    - a Gaussian distribution when the prior is Gaussian
  - Sample $\lambda^{t+1} \sim q(\lambda|\eta^{t+1}, \Theta^t, \mathbf{Z}^t, \Phi^t) \propto \phi(\mathbf{y}, \lambda|\mathbf{Z}^t, \eta^{t+1})$
    - a generalized inverse Gaussian distribution, i.e., λ^{-1} follows inverse Gaussian
  - Sample $(\Theta, \mathbf{Z}, \Phi)^{t+1} \sim p(\Theta, \mathbf{Z}, \Phi|\eta^{t+1}, \lambda^{t+1})$
    $\propto p_0(\Theta, \mathbf{Z}, \Phi) p(\mathbf{W}|\mathbf{Z}, \Phi) \phi(\mathbf{y}, \boldsymbol{\lambda}^{t+1}|\mathbf{Z}, \boldsymbol{\eta}^{t+1})$
    - a supervised LDA model with closed-form local conditionals by exploring data independency.

A Collapsed Gibbs Sampling Algorithm

- The collapsed joint distribution

$$q(\eta, \lambda, \mathbf{Z}) = \int q(\eta, \lambda, \Theta, \mathbf{Z}, \Phi) d\Theta d\Phi$$

- A Gibbs sampling algorithm iterates over:
  - Sample $\eta^{t+1} \sim q(\eta|\lambda^t, \mathbf{Z}^t) \propto p_0(\eta)\phi(\mathbf{y}, \lambda^t|\mathbf{Z}^t, \eta)$
    - a Gaussian distribution when the prior is Gaussian
  - Sample $\lambda^{t+1} \sim q(\lambda|\eta^{t+1}, \mathbf{Z}^t) \propto \phi(\mathbf{y}, \lambda|\mathbf{Z}^t, \eta^{t+1})$
    - a generalized inverse Gaussian distribution, i.e., λ^{-1} follows inverse Gaussian
  - Sample $\mathbf{Z}^{t+1} \sim q(\mathbf{Z}|\eta^{t+1}, \lambda^{t+1})$
    $\propto \int p_0(\Theta, \mathbf{Z}, \Phi) p(\mathbf{W}|\mathbf{Z}, \Phi) \phi(\mathbf{y}, \boldsymbol{\lambda}^{t+1}|\mathbf{Z}, \boldsymbol{\eta}^{t+1}) d\Theta d\Phi$
    - closed-form local conditionals

$$q(z_{dn}^k = 1|\mathbf{Z}_\neg, \eta, \lambda, w_{dn} = t)$$

The Collapsed Gibbs Sampling Algorithm

Algorithm 1 Collapsed Gibbs Sampling Algorithm

1: **Initialization:** set $\boldsymbol{\lambda} = 1$ and randomly draw z_{dk} from a uniform distribution.
2: **for** $m = 1$ **to** M **do**
3: draw the classifier from the normal distribution (11)
4: **for** $d = 1$ **to** D **do**
5: **for** each word n in document d **do**
6: draw the topic using distribution (12)
7: **end for**
8: draw λ_d^{-1} (and thus λ_d) from distribution (13).
9: **end for**
10: **end for**

Easy to Parallelize

Some Analysis

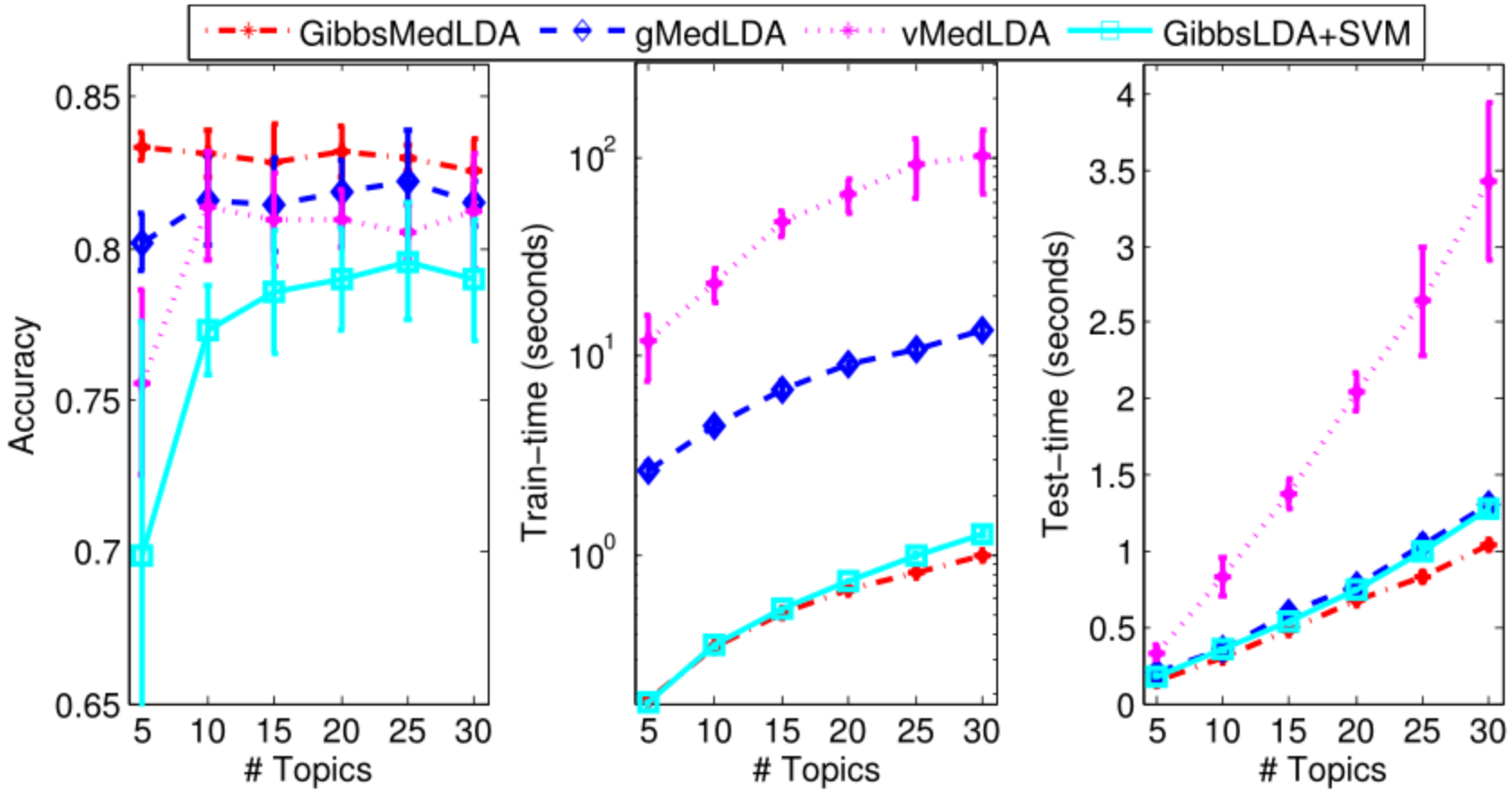
- The Markov chain is guaranteed to converge
- Per-iteration time complexity

$$\mathcal{O}(K^3 + N_{total}K)$$

  - N_{total} the total number of words

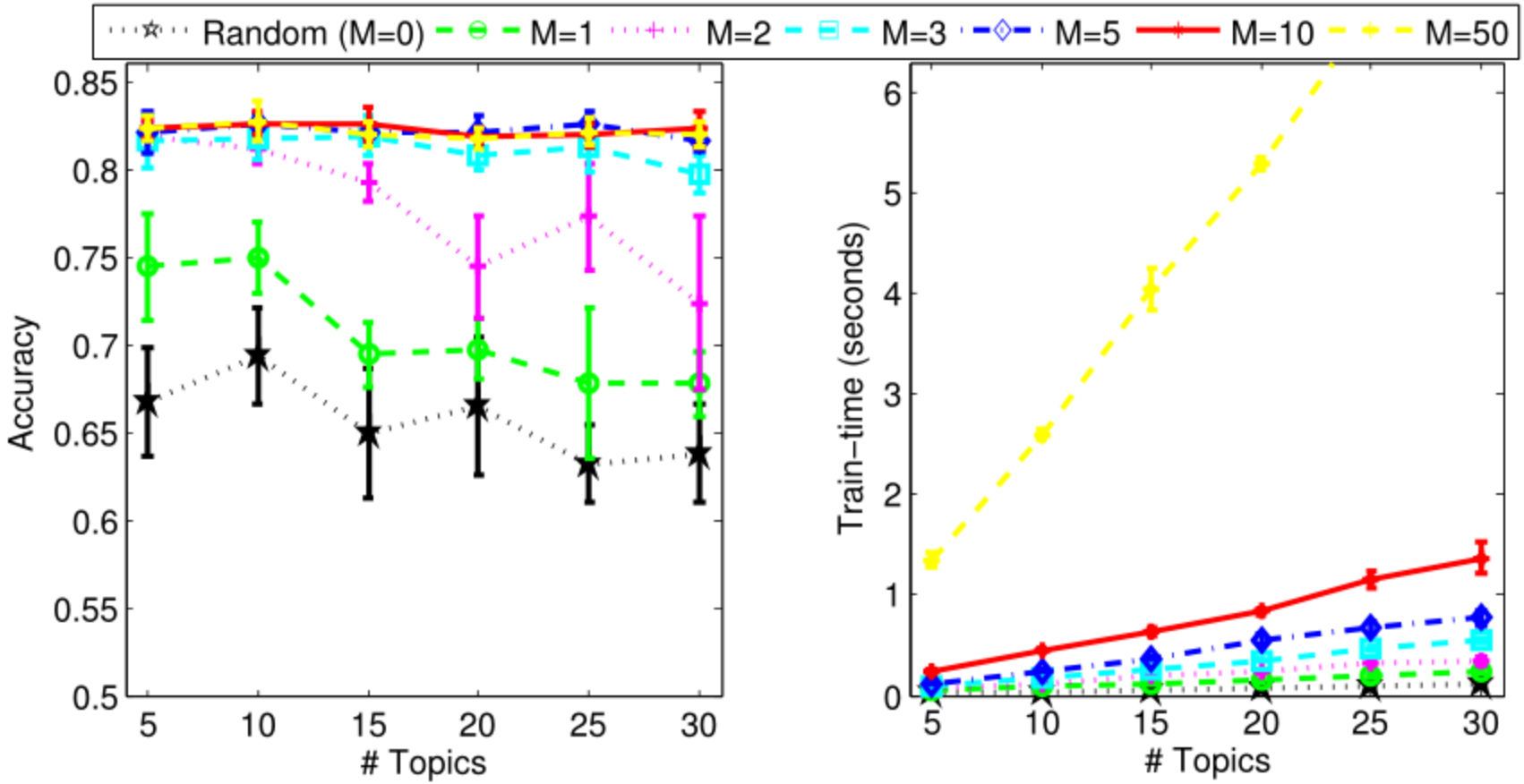
Experiments

- 20Newsgroups binary classification



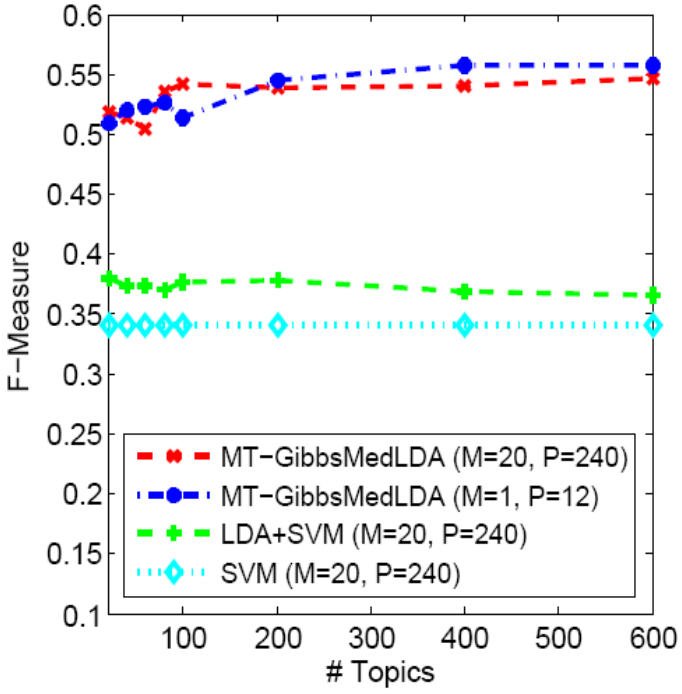
Experiments

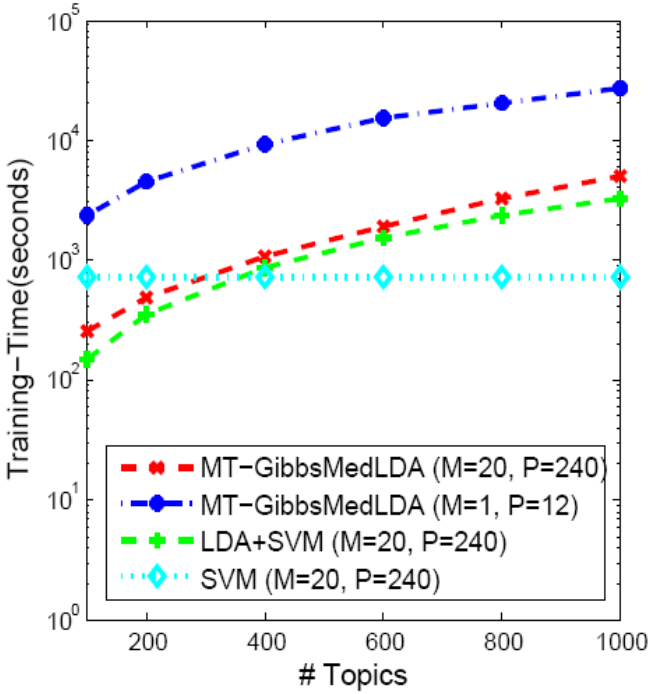
- **Sensitivity to burn-in**: binary classification



Distributed Inference Algorithms

- Leverage big clusters
- Allow learning big models that can't fit on a single machine





- 20 machines;
- 240 CPU cores

- 1.1M multi-labeled Wiki pages
- 20 categories (scale to hundreds/thousands of categories)

[Zhu, Zheng, Zhou, & Zhang, KDD2013]

Summary so far …

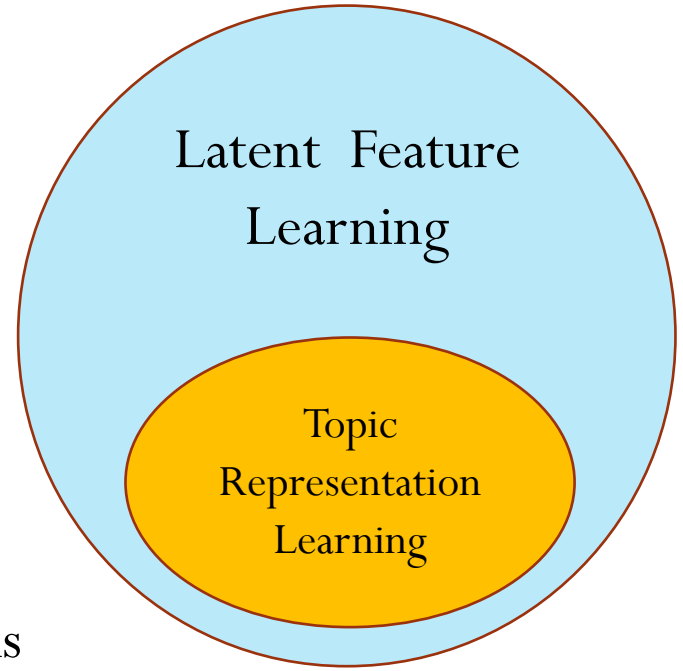
- Regularized Bayesian inference (RegBayes)

- Max-margin supervised topic models
  - A RegBayes model with max-margin posterior constraints
  - Inference can be done with an iterative procedure
  - Both variational and Monte Carlo methods were developed
  - Monte Carlo methods produce better accuracy

Nonparametric Max-margin Latent Feature Learning

- Problems with human-tuned features
  - Needs expert knowledge
  - Time-consuming and expensive
  - Can be incomplete or over-complete
  - Does not generalize to other domains

- Latent feature learning has a large literature
  - Feature representations, deep networks
  - Nonparametric Bayesian methods



清華大學 Tsinghua University

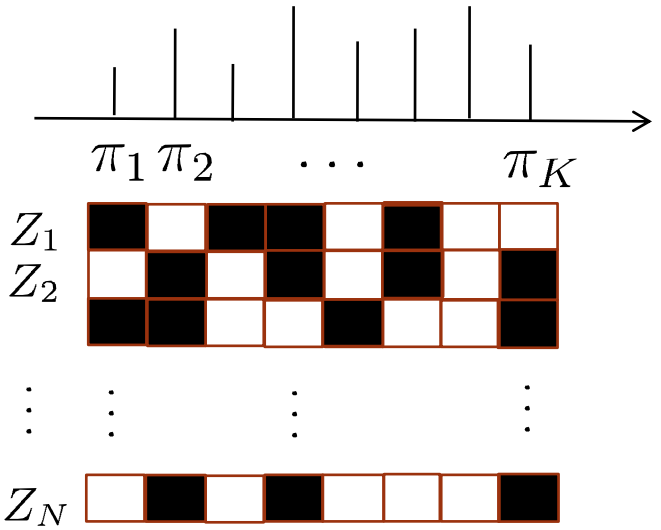
Classification and Multi-task Learning

Bayesian Latent Feature Models (finite)

- A random finite binary latent feature models

$$\pi_k|\alpha \sim \text{Beta}(\frac{\alpha}{K}, 1)$$

$$z_{ik}|\pi_k \sim \text{Bernoulli}(\pi_k)$$



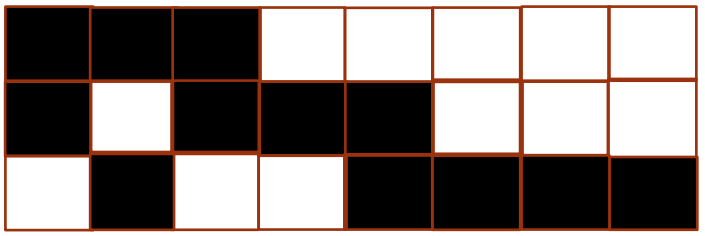
- π_k is the relative probability of each feature being on

- $z_{i.}$ are binary vectors, giving the latent structure that's used to generate the data, e.g.,

$$\mathbf{x}_i \sim \mathcal{N}(\eta^\top z_{i.}, \delta^2)$$

Indian Buffet Process

- A stochastic process on infinite binary feature matrices
- Generative procedure:
  - Customer 1 chooses the first K_1 dishes: $K_1 \sim \text{Poisson}(\alpha)$
  - Customer *i* chooses:
    - Each of the existing dishes with probability $\frac{m_k}{i}$
    - K_i additional dishes, where $K_i \sim \text{Poisson}(\frac{\alpha}{i})$



cust 1: new dishes 1-3

cust 2: old dishes 1,3; new dishes 4-5

cust 3: old dishes 2,5; new dishes 6-8

$$Z \sim \mathcal{IBP}(\alpha)$$

(Griffiths & Ghahramani, NIPS 2005)

Posterior Constraints – classification

- Suppose latent features $\mathbf{z}$ are given, we define *latent discriminant function*:

$$f(\mathbf{x}; \mathbf{z}, \boldsymbol{\eta}) = \boldsymbol{\eta}^\top \mathbf{z}$$

- Define *effective discriminant function* (reduce uncertainty):

$$f(\mathbf{x}; q(\mathbf{Z}, \boldsymbol{\eta})) = \mathbb{E}_{q(\mathbf{Z}, \boldsymbol{\eta})}[f(\mathbf{x}, \mathbf{z}; \boldsymbol{\eta})] = \mathbb{E}_{q(\mathbf{Z}, \boldsymbol{\eta})}[\boldsymbol{\eta}^\top \mathbf{z}]$$

- Posterior constraints with max-margin principle

$$\forall n \in \mathcal{I}_{\text{tr}} : y_n f(\mathbf{x}_n; p(\mathbf{Z}, \boldsymbol{\eta})) \geq 1 - \xi_n$$

- Convex U function

$$U(\xi) = C \sum_{n \in \mathcal{I}_{\text{tr}}} \xi_n$$

The RegBayes Problem

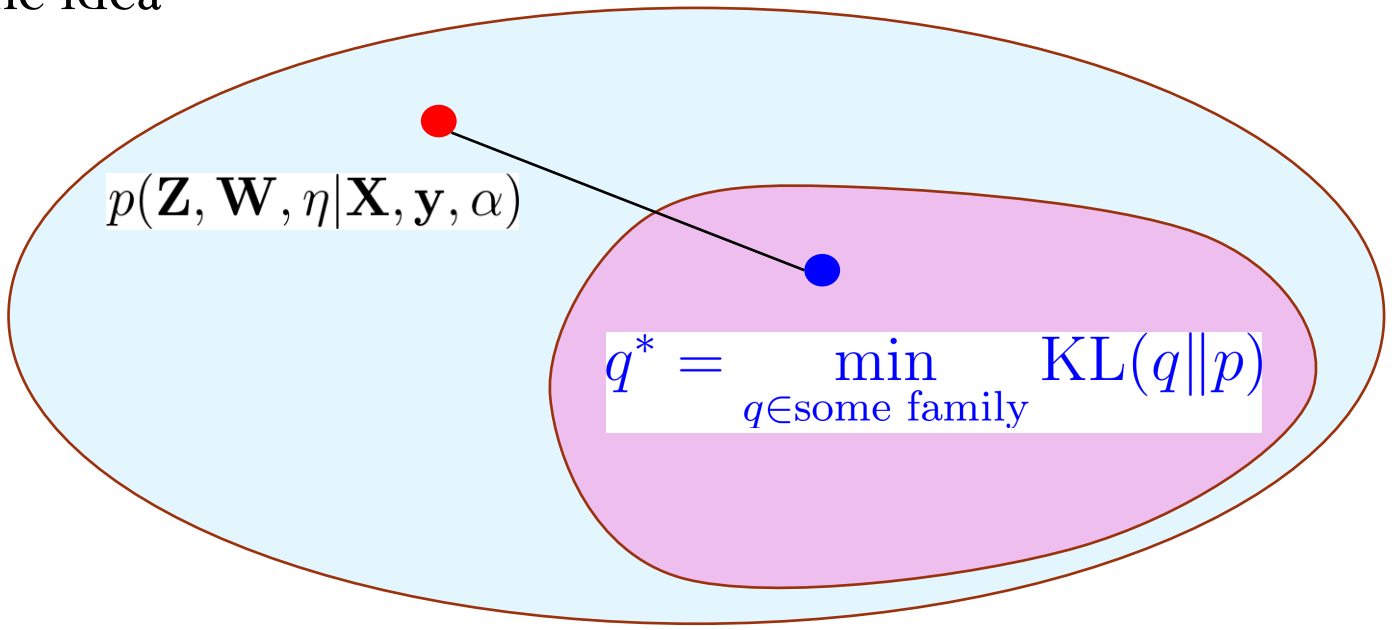
$$\min_{q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta})} \quad \mathcal{L}(q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta}) + 2c \cdot \mathcal{R}(q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta}))$$

- where $\mathcal{L}(q) = \mathrm{KL}(q \| \pi(\mathbf{Z},\mathbf{W},\eta)) - \mathbb{E}_q[\log p(\mathbf{x}|\mathbf{Z},\mathbf{W})]$
- the hinge loss (posterior regularization) is

$$\mathcal{R}(q) = \sum_n \max(0, 1 - y_n f(\mathbf{x}_n; q(\mathbf{Z},\boldsymbol{\eta}))$$

Truncated Variational Inference

- The idea



  - Depends on a stick-breaking representation of IBP (Teh et al., 2007)
  - Truncated mean-field inference with an upper bound of features
  - Works reasonably well in practice

Posterior Regularization with a Gibbs Classifier

- Posterior distribution to learn

$$q(\mathbf{Z}, \boldsymbol{\eta})$$

- Gibbs classifier randomly draws a sample to make prediction

$$(\mathbf{Z}, \boldsymbol{\eta}) \sim q(\mathbf{Z}, \boldsymbol{\eta})$$

  - For classification, we measure the loss of classifier $(\mathbf{Z}, \boldsymbol{\eta})$

$$\mathcal{R}(\mathbf{Z}, \boldsymbol{\eta}) = \sum_n \max(0, 1 - y_n f(\mathbf{x}_n; \mathbf{Z}, \boldsymbol{\eta})$$

  - It minimizes the expected loss

$$\mathcal{R}'(q) = \mathbb{E}_q \left[\sum_n \max(0, 1 - y_n f(\mathbf{x}_n; \mathbf{Z}, \boldsymbol{\eta}) \right]$$

Comparison

- Expected hinge-loss is an upper bound

$$\mathcal{R}'(q) \geq \mathcal{R}(q)$$

- Averaging classifier is suitable for variational inference with truncation (Zhu et al., arXiv, 2013)

- Gibbs classifier is suitable for MCMC without truncation (more details ...)

More Details on MCMC

- RegBayes problem

$$\min_{q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta})} \quad \mathcal{L}(q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta}) + 2c \cdot \mathcal{R}'(q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta}))$$

- The solution is

$$q(\mathbf{Z},\mathbf{W},\boldsymbol{\eta}) = \frac{\pi(\mathbf{Z},\mathbf{W},\boldsymbol{\eta})p(\mathbf{X}|\mathbf{Z},\mathbf{W})\phi(\mathbf{y}|\mathbf{Z},\boldsymbol{\eta})}{\psi(\mathbf{X},\mathbf{y})}$$

  - where

$$\phi(\mathbf{y}|\mathbf{Z},\boldsymbol{\eta}) = \prod_n \phi(y_n|\mathbf{Z},\boldsymbol{\eta}) = \prod_n \exp\{-2c\max(0, 1 - y_n f(\mathbf{x}_n;\mathbf{Z},\boldsymbol{\eta})\}$$

More Details on MCMC

- Scale mixture representation: $\zeta_n = 1 - y_n f(\mathbf{x}_n; \mathbf{Z}, \boldsymbol{\eta})$

$$\phi(y_n|\mathbf{Z}, \eta) = \int_0^\infty \frac{1}{\sqrt{2\pi\lambda_n}} \exp\left(-\frac{(\lambda_n + c\zeta_n)^2}{2\lambda_n}\right) d\lambda_n$$

  - Follows (Polson & Scott, 2011)

- Data augmentation representation

$$q(\mathbf{Z}, \mathbf{W}, \eta) = \int q(\mathbf{Z}, \mathbf{W}, \eta, \lambda) d\lambda$$

$$\text{where } q(\mathbf{Z}, \mathbf{W}, \eta, \lambda) = \frac{\pi(\mathbf{Z}, \mathbf{W}, \eta) p(\mathbf{X}|\mathbf{Z}, \mathbf{W}) \phi(\mathbf{y}, \lambda|\mathbf{Z}, \eta)}{\psi(\mathbf{X}, \mathbf{y})}$$

$$\phi(\mathbf{y}, \lambda|\mathbf{Z}, \eta) = \prod_n \frac{1}{\sqrt{2\pi\lambda_n}} \exp\left(-\frac{(\lambda_n + c\zeta_n)^2}{2\lambda_n}\right)$$

More Details on MCMC

- Data augmented posterior

$$q(\mathbf{Z}, \mathbf{W}, \eta, \lambda) = \frac{\pi(\mathbf{Z}, \mathbf{W}, \eta) p(\mathbf{X}|\mathbf{Z}, \mathbf{W}) \phi(\mathbf{y}, \lambda|\mathbf{Z}, \eta)}{\psi(\mathbf{X}, \mathbf{y})}$$

- A Gibbs sampler is as follows
  - Sample $\boldsymbol{\eta} \sim q(\boldsymbol{\eta}|\mathbf{Z}, \mathbf{W}, \lambda) \propto \pi(\boldsymbol{\eta}) \phi(\mathbf{y}, \lambda|\mathbf{Z}, \boldsymbol{\eta})$
    - a Gaussian distribution if the prior is Gaussian
  - Sample $\lambda \sim q(\lambda|\mathbf{Z}, \mathbf{W}, \eta) \propto \phi(\mathbf{y}, \lambda|\mathbf{Z}, \eta)$
    - a generalized inverse Gaussian distribution
  - Sample $(\mathbf{Z}, \mathbf{W}) \sim q(\mathbf{Z}, \mathbf{W}|\eta, \lambda) \propto \pi(\mathbf{Z}, \mathbf{W}) p(\mathbf{X}|\mathbf{Z}, \mathbf{W}) \phi(\mathbf{y}, \lambda|\mathbf{Z}, \eta)$
    - Similar as the Gaussian infinite latent feature model (Griffiths & Ghahramani, 2005)

Extensions to Multi-task Learning

Multi-task Learning (MTL)

- **[Wikipedia]** MTL is an approach to machine learning that learns a problem together with other related problems, using a ***shared representation***

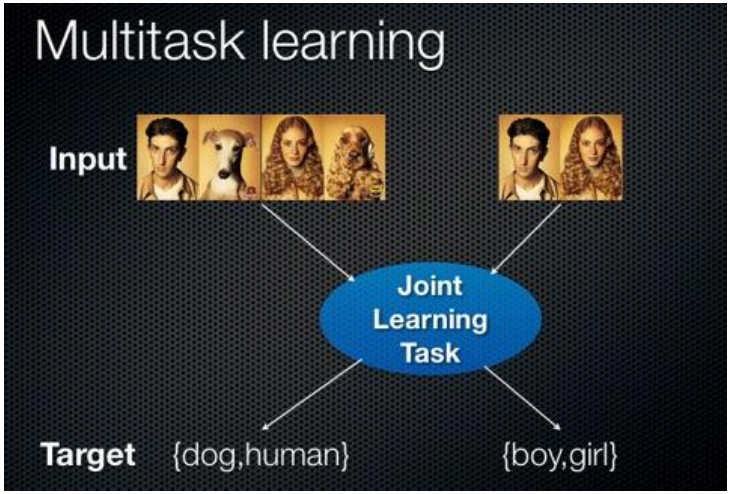


Figure from Wikipedia
Author: Kilian Weinberger

- The goal of MTL is to improve the performance of learning algorithms by learning classifiers for multiple tasks jointly

- It works particularly well if these tasks have some commonality and are generally slightly under sampled

Multi-task Representation Learning

- Assumption:
  - common underlying representation across tasks
- Representative works:
  - ASO (alternating structure optimization): learn a small set of shared features across tasks [Ando & Zhang, 2005]
  - Convex feature learning via sparse norms [Argyriou et al., 2006]

Basic Setup of the Learning Paradigm

- Tasks: $m = 1, \cdots, M$
- N examples per task

$$(\mathbf{x}_{m1}, y_{m1}), \cdots, (\mathbf{x}_{mN}, y_{mN}) \in \mathbb{R}^D \times \mathbb{R}$$

- Estimate

$$f_m : \mathbb{R}^D \to \mathbb{R}, \ \forall m = 1, \cdots, M$$

- Consider features

$$h_1(\mathbf{x}), \ \cdots, h_K(\mathbf{x})$$

- Predict using functions

$$f_m(\mathbf{x}) = \sum_{k=1}^{K} \eta_{mk} h_k(\mathbf{x})$$

Learning a Projection Matrix

- Tasks: $m = 1, \cdots, M$
- N examples per task

$$(\mathbf{x}_{m1}, y_{m1}), \cdots, (\mathbf{x}_{mN}, y_{mN}) \in \mathbb{R}^D \times \mathbb{R}$$

- Estimate

$$f_m : \mathbb{R}^D \to \mathbb{R}, \ \forall m = 1, \cdots, M$$

- Consider features

$$h_k(\mathbf{x}) = \mathbf{z}_k^\top \mathbf{x}, \ k = 1, \cdots, \infty$$

- Predict using functions ($\mathbf{Z}$ is a $D \times \infty$ projection matrix)

$$f_m(\mathbf{x}; \mathbf{Z}, \boldsymbol{\eta}) = \sum_{k=1}^{\infty} \eta_{mk} (\mathbf{z}_k^\top \mathbf{x}) = \boldsymbol{\eta}_m^\top (\mathbf{Z}^\top \mathbf{x})$$

Max-margin Posterior Regularizations

- Similar as in infinite latent SVMs
  - Averaging classifier

$$y_{mn}\mathbb{E}_q[f_m(\mathbf{x}_{mn};\mathbf{Z},\boldsymbol{\eta})] \geq 1-\xi_{mn}$$

    - The hinge loss

$$\mathcal{R} = \sum_{m,n\in\mathcal{I}_{\text{tr}}^m} \max\left(0, 1-y_{mn}\mathbb{E}_q[f_m(\mathbf{x}_{mn};\mathbf{Z}\boldsymbol{\eta})]\right)$$

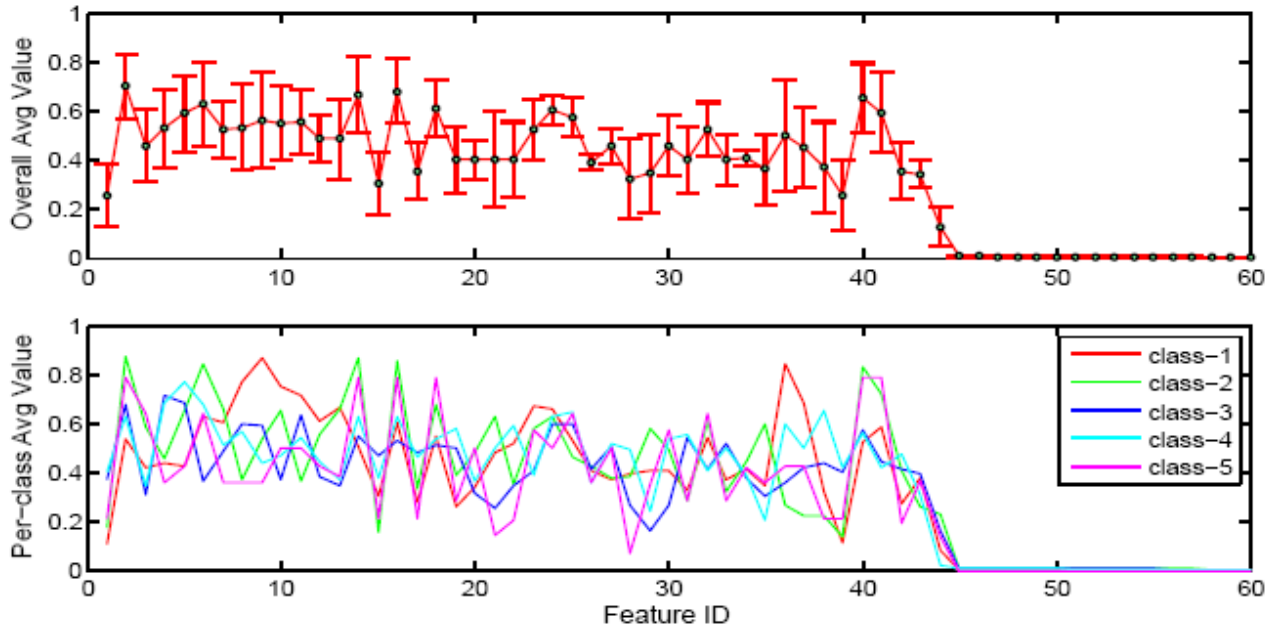
  - Gibbs classifier

$$\mathcal{R}' = \mathbb{E}_q\left[\sum_{m,n\in\mathcal{I}_{\text{tr}}^m} \max\left(0, 1-y_{mn}f_m(\mathbf{x}_{mn};\mathbf{Z}\boldsymbol{\eta})\right)\right]$$

Experimental Results

- Classification
  - Accuracy and F1 scores on TRECVID2003 and Flickr image datasets

Model	TRECVID2003		Flickr	
	Accuracy	F1 score	Accuracy	F1 score
EFH+SVM	0.565 ± 0.0	0.427 ± 0.0	0.476 ± 0.0	0.461 ± 0.0
MMH	$\mathbf{0.566} \pm 0.0$	0.430 ± 0.0	$\mathbf{0.538} \pm 0.0$	$\mathbf{0.512} \pm 0.0$
IBP+SVM	0.553 ± 0.013	0.397 ± 0.030	0.500 ± 0.004	0.477 ± 0.009
iLSVM	0.563 ± 0.010	$\mathbf{0.448} \pm 0.011$	0.533 ± 0.005	0.510 ± 0.010



Experimental Results

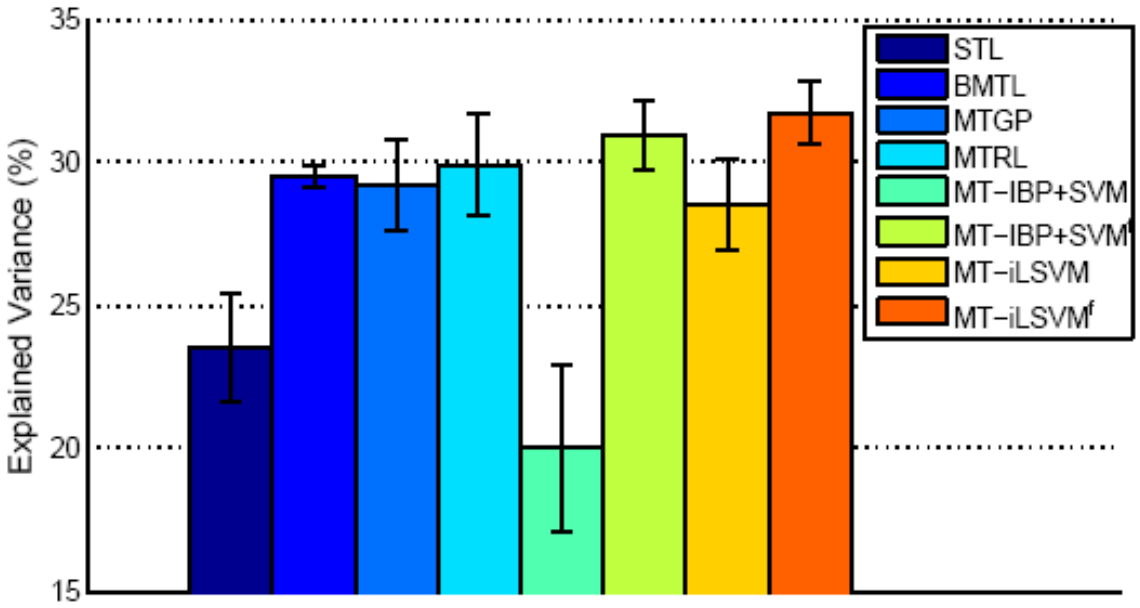
- Multi-label Classification (multiple binary classification)
  - Accuracy and F1 scores (Micro & Macro) on Yeast and Scene datasets

Model	Acc	F1-Macro	F1-Micro
YaXue [Xue et al., 2007]	0.5106	0.3897	0.4022
Piyushrai [Piyushrai et al., 2010]	0.5424	0.3946	0.4112
MT-iLSVM	0.5792 ± 0.003	0.4258 ± 0.005	0.4742 ± 0.008
Gibbs MT-iLSVM	0.5851 ± 0.005	0.4294 ± 0.005	0.4763 ± 0.006

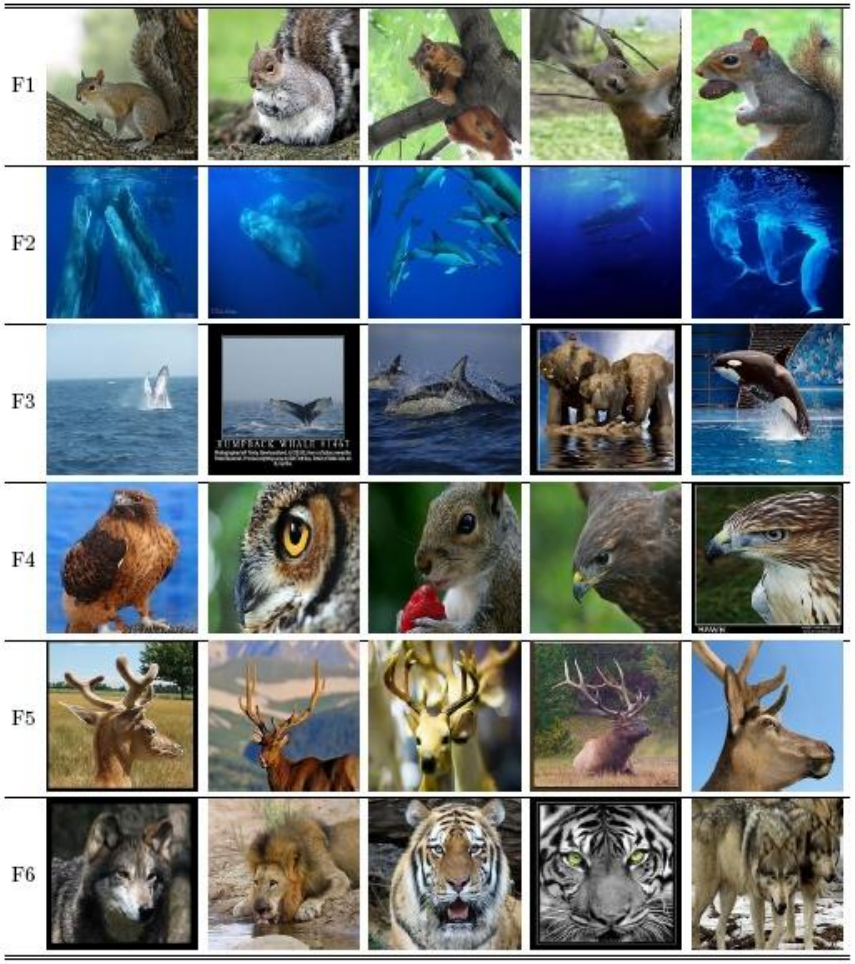
Model	Acc	F1-Macro	F1-Micro
YaXue [Xue et al., 2007]	0.7765	0.2669	0.2816
Piyushrai [Piyushrai et al., 2010]	0.7911	0.3214	0.3226
MT-iLSVM	0.8752 ± 0.004	0.5834 ± 0.026	0.6148 ± 0.020
Gibbs MT-iLSVM	0.8855 ± 0.004	0.6494 ± 0.011	0.6458 ± 0.011

Experimental Results

- Multi-task Regression
  - School dataset (139 regression tasks) – a standard dataset for evaluating multi-task learning
  - Percentage of *explained variance* (higher, better)



Example Features



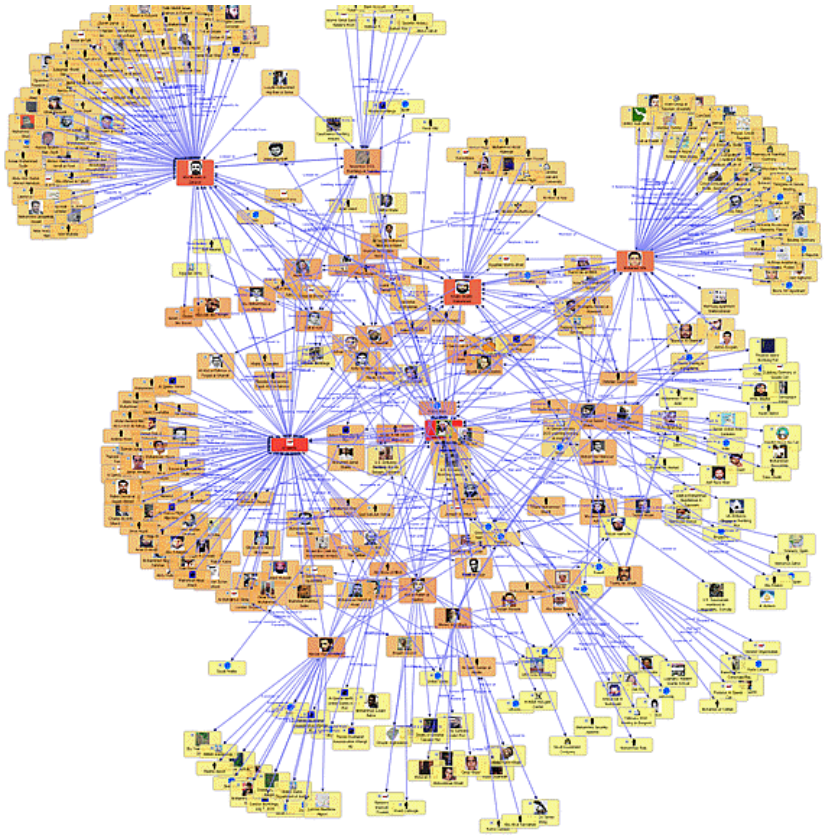
Link Prediction

Link Prediction

- We're living in social networks

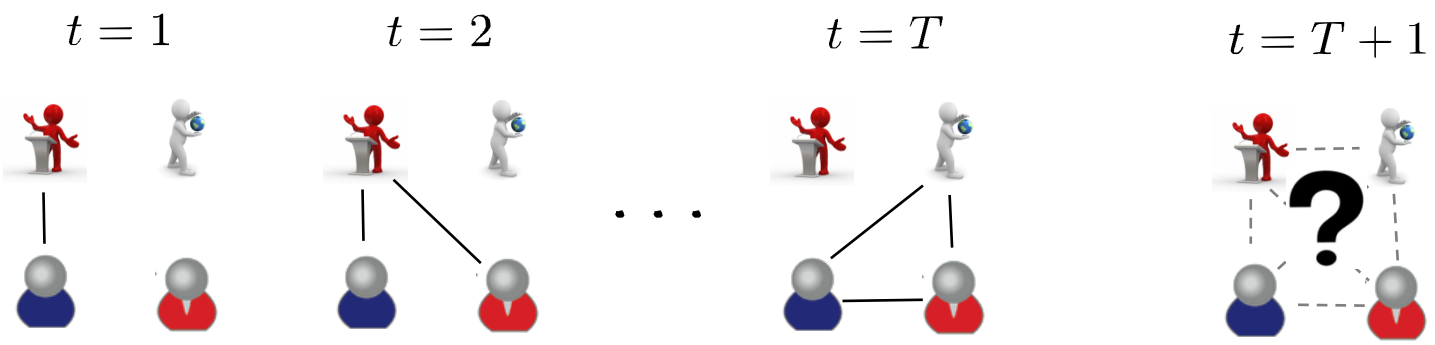
Link Prediction

- But network structures are usually unclear, unobserved, or corrupted with noise

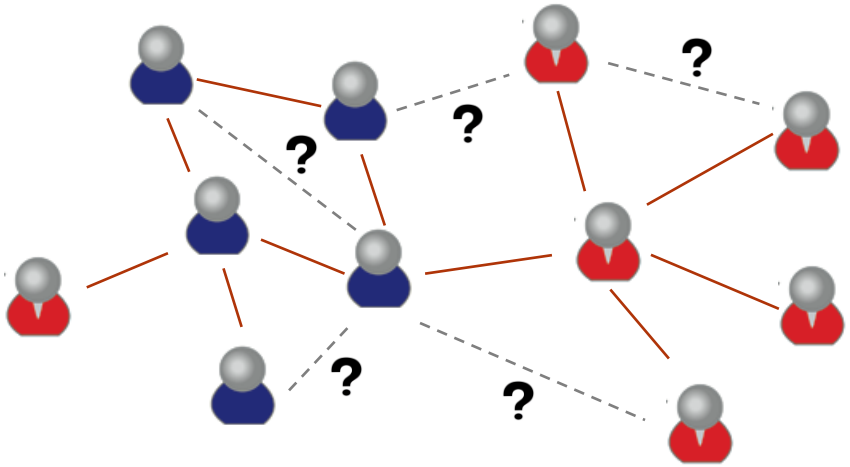


Link Prediction – task

- Dynamic networks



- Static networks



We treat it as a supervised learning task with 1/-1 labels

Link Prediction as Supervised Learning

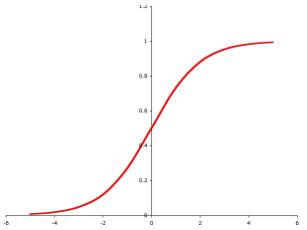
- Building classifiers with manually designed features from networks
  - Topological features
    - Shortest distance, number of common neighbors, Jaccard's coefficient, etc.
  - Attributes about individual entities
    - E.g., the papers an authors has published
  - Proximity features
    - E.g., two authors are close, if their research work evolves around a large set of identical keywords
  - …

[Hasan et al., 2006]

Link Prediction as Supervised Learning

- Latent feature relational models
  - Each entity is associated with a point $\mu_i \in \mathbb{R}^K$ in a latent feature space
  - Then, a link likelihood is generally defined

$$p(Y_{ij} = 1 | X_{ij}, \mu_i, \mu_j) = \Phi\Big(\mu + \beta^\top X_{ij} + \psi(\mu_i, \mu_j)\Big)$$



Latent distance model (Hoff et al., 2002)

$$\psi(\mu_i, \mu_j) = -d(\mu_i, \mu_j) = -\|\mu_i - \mu_j\|$$

generalize

Latent eigenmodel (Hoff, 2007)

$$\psi(\mu_i, \mu_j) = \mu_i^\top D \mu_j$$

D *is diagonal*

generalize generalize

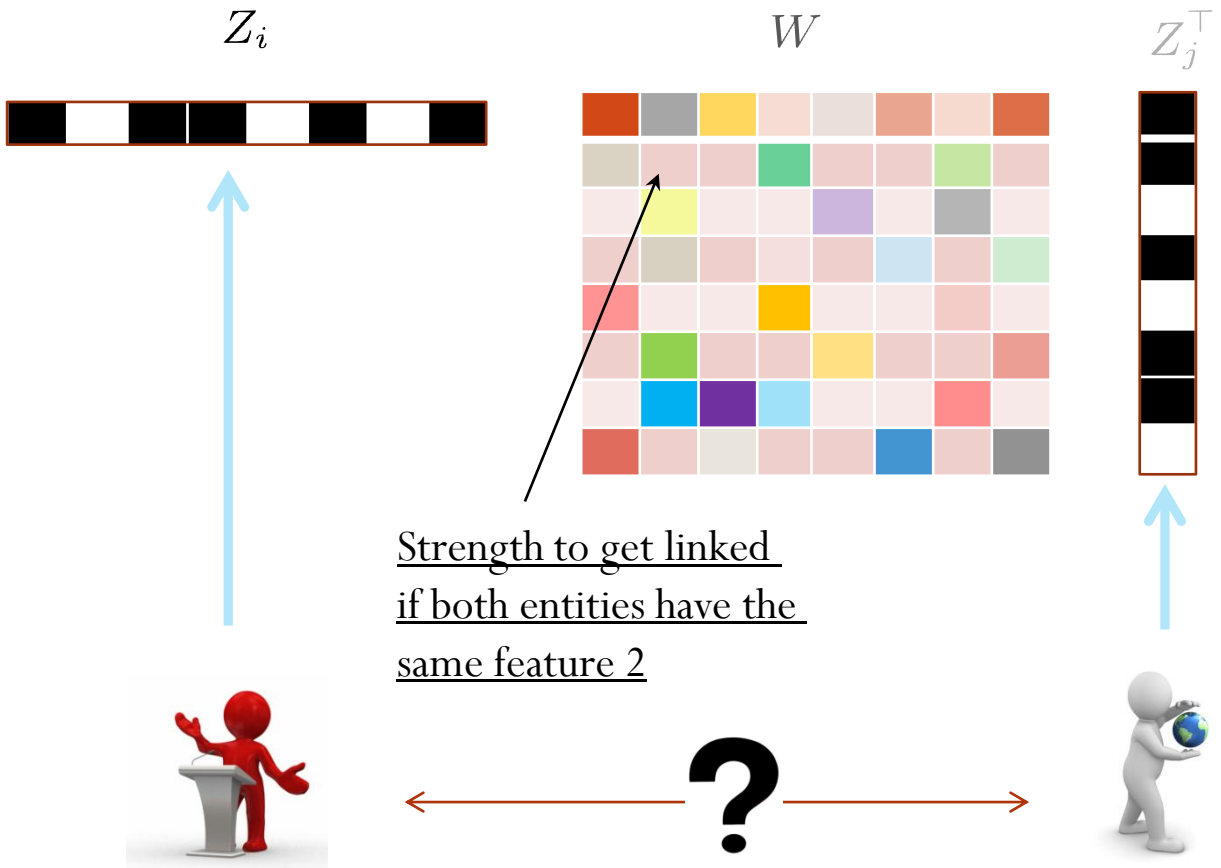
Nonparametric latent feature relational model (LFRM)
(Miller, Griffiths, & Jordan, 2009)

$$\psi(\mu_i, \mu_j) = \mu_i^\top W \mu_j, \ \ \textit{where } \mu_i \in \{0, 1\}^\infty$$

More at ICML 2012

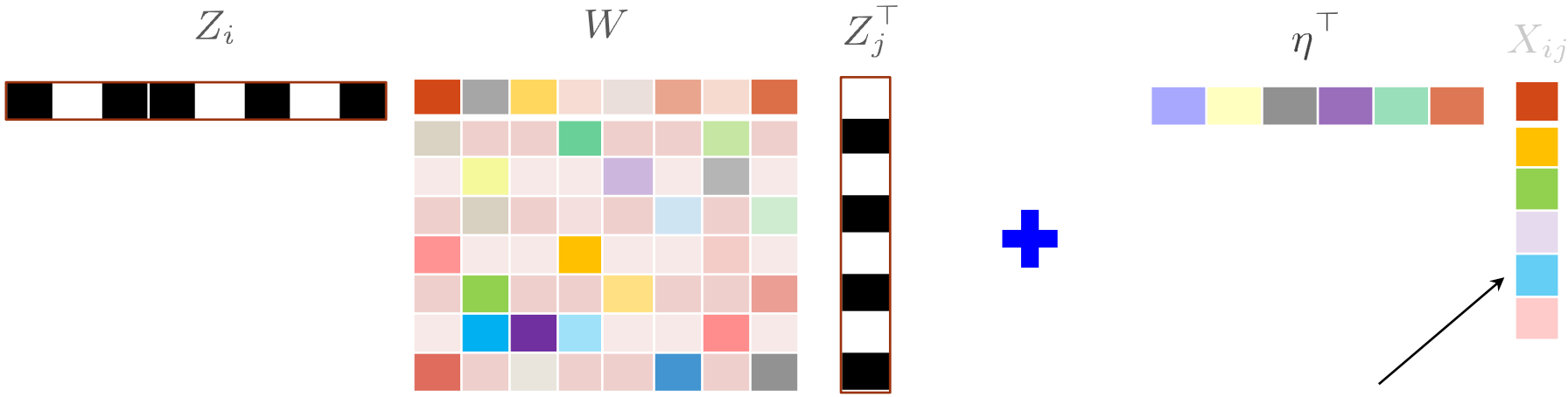
Discriminant Function with Latent Features

$$f(Z_i, Z_j; W) = Z_i W Z_j^\top$$



Discriminant Function with Latent Features

$$f(Z_i, Z_j; X_{ij}, W, \eta) = Z_i W Z_j^\top + \eta^\top X_{ij}$$



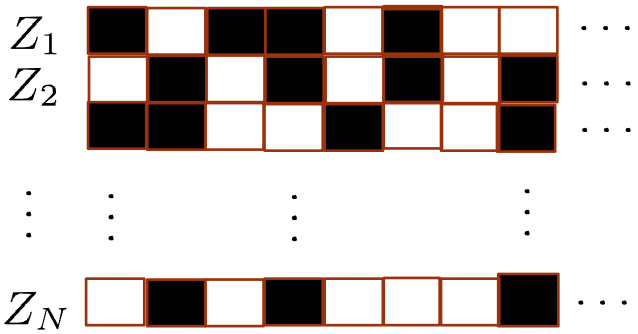
$$X_{ij} = [g(X_i); g(X_j); h(X_i, X_j)]$$

Ex: [red, table; gray, ball; neighbors…]

Latent Features ***Observable Features***

Infinite Latent Feature Matrix

- N entities → a latent feature matrix Z



- How many columns (i.e., features) are sufficient?

  → a stochastic process to infer from data – Indian buffet process (IBP) (Griffiths & Ghahramani, 2006)

- What learning principle is good?

  → max-margin learning – (Vapnik, 1995; Taskar et al., 2003)

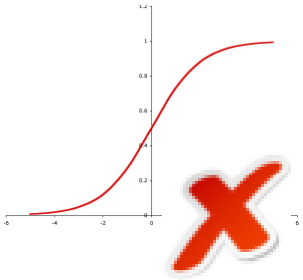
MedLFRM (max-margin latent feature relational model)

Max-margin Latent Feature Relational Model

- Effective discriminant function using linear expectation operator

$$f(X_{ij}) = \mathbb{E}_{p(Z,\Theta)}[f(Z_i, Z_j; X_{ij}, \Theta)]$$

Observable function — Latent function



- Averagingclassifier

$$\hat{Y}_{ij} = \text{sign} f(X_{ij})$$

- The learning problem:

$$\min_{p(Z,\Theta)} \ \text{KL}(p(Z,\Theta)||p_0(Z,\Theta)) + C\mathcal{R}_\ell(p(Z,\Theta))$$
$$\text{s.t.}: \ p(Z,\Theta) \in \mathcal{P}.$$

  - hinge loss of the expectation rule (piecewise linear of p)

$$\mathcal{R}_\ell(p(Z,\Theta)) = \sum_{(i,j)\in\mathcal{I}} \max(\ell, Y_{ij} f(X_{ij}))$$

Augmented with Stick-breaking Variables

- The IBP prior in a hierarchical form (Teh, Gorur, & Ghahramani, 2007):

$$v_i \sim \text{Beta}(\alpha, 1)$$

$$\pi_i(\mathbf{v}) = v_i \pi_{i-1}(\mathbf{v}) = \prod_{j=1}^{i} v_j$$

Conditional Independence

$$Z_{ik} \sim \text{Bernoulli}(\pi_k)$$

- Augmented learning problem:

$$\min_{p(\boldsymbol{\nu}, Z, \Theta)} \text{KL}(p(\boldsymbol{\nu}, Z, \Theta) \| p_0(\boldsymbol{\nu}, Z, \Theta)) + C\mathcal{R}_\ell(p(Z, \Theta))$$

$$\text{s.t.} : p(\boldsymbol{\nu}, Z, \Theta) \in \mathcal{P}.$$

Inference (outline)

- Simplifying a

$$\min \quad \mathrm{KL}(p(\Theta)\|p_0(\Theta)) + C \sum \xi_{ij}$$

$$p(\boldsymbol{\nu},$$

$$\psi_{ik} = \Phi\Big(\sum_{j=1}^{k} \mathbb{E}_p[\log \nu_j] - \mathcal{L}_k^{\nu} + C\partial_{\psi_{ik}}\mathcal{R}_\ell\Big)$$

  - $K-$ trunca

$$p(\nu_k|\gamma_k)$$

$$\partial_{\psi_{ik}}\mathcal{R}_\ell = -\sum_{j\in\mathcal{I}_i} Y_{ij}\Lambda_{k\cdot}\psi_j^\top - \sum_{j\in\mathcal{I}_i'} Y_{ji}\psi_j\Lambda_{\cdot k}$$

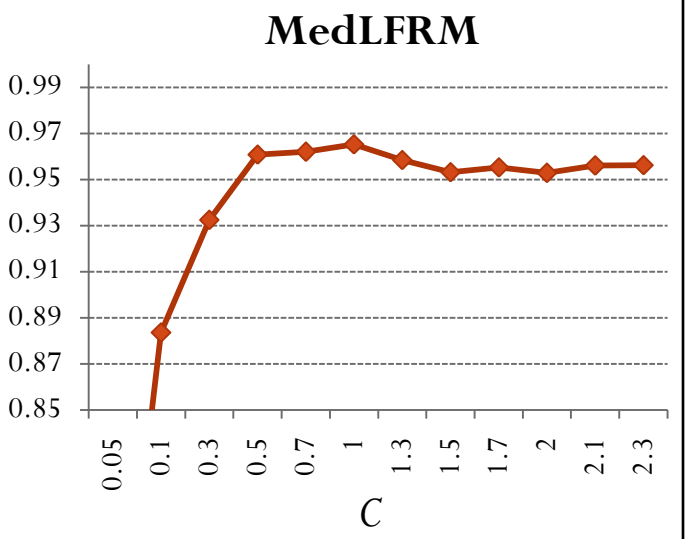
$$-\mathbb{I}(Y_{ii}f(X_{ii}) \le \ell)Y_{ii}(\Lambda_{kk}(1-\psi_{ik}) + \Lambda_{k\cdot}\psi_i^\top),$$

- Alternating min

  - *p(Θ) – learn an "ob ed" binary SVM with expected features*
  - *p(Z) – closed-form update rules using sub-gradient methods*
  - *p(v) – same update rules as in IBP latent factor models*

Bayesian MedLFRM

- One problem with MedLFRM is the tuning of C, e.g., using CV
- Hierarchical Bayesian ideas to infer it from data



- The Normal-Gamma hyper-prior:
  - Prior of model parameters with common mean and variance

$$p_0(\Theta|\mu,\tau) = \prod_{kk'} \mathcal{N}(\mu,\tau^{-1}) \prod_{d} \mathcal{N}(\mu,\tau^{-1})$$

  - The hyper-prior

$$p_0(\mu|\tau) = \mathcal{N}(\mu_0,(n_0\tau)^{-1}),\ p_0(\tau) = \mathcal{G}(\frac{\nu_0}{2},\frac{2}{S_0})$$

    - a weak hyper-prior suffices, e.g., $\mu_0 = 0,\ n_0 = 1,\ \nu_0 = 2,\ S_0 = 1$

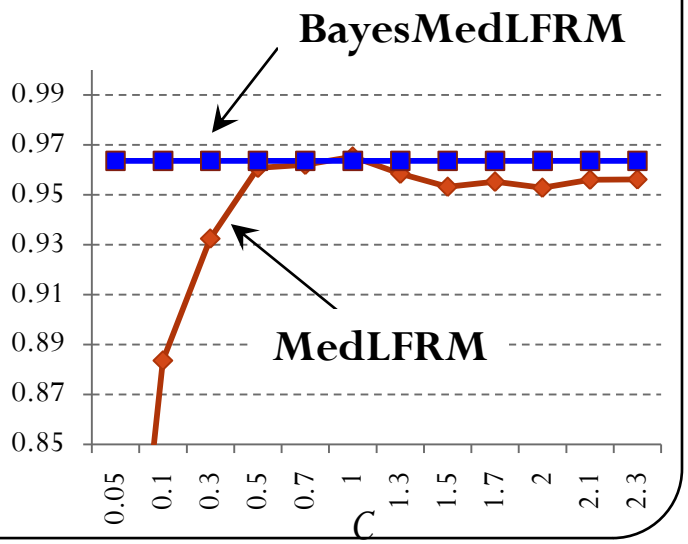
Bayesian MedLFRM

- Learning problem

$$\min_{p(\boldsymbol{\nu}, Z, \mu, \tau, \Theta)} \mathrm{KL}(p(\boldsymbol{\nu}, Z, \mu, \tau, \Theta) \| p_0(\boldsymbol{\nu}, Z, \mu, \tau, \Theta)) + \mathcal{R}_\ell(p(Z, \Theta))$$
$$\text{s.t.}: \quad p(\boldsymbol{\nu}, Z, \mu, \tau, \Theta) \in \mathcal{P}.$$

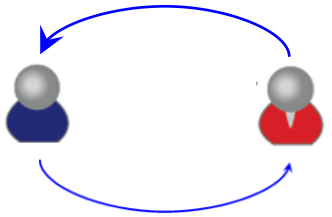
- Inference – similar iterative procedure (outline)
  - The step of inferring $p(\boldsymbol{\nu}, Z)$ doesn't change
  - For $p(\Theta)$, we solve a binary SVM
  - For $p(\mu, \tau)$, we have closed-form rule

$$\frac{1}{C} = \lambda = \mathbb{E}[\tau] = \frac{\tilde{\nu}}{\tilde{S}}$$

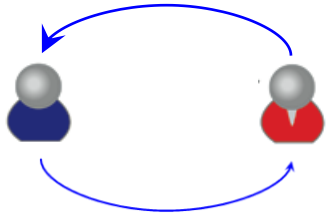


Datasets & Classification Setups

- Countries
  - 14 countries, 56 relations
  - 90 observable attributes about countries
  - predict the existence/non-existence (1/-1 classification) of each relation for each pair of country

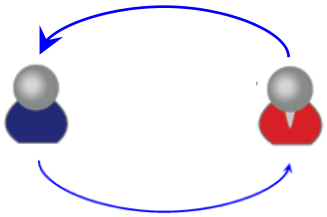


- Kinship
  - 104 people, 26 kinship relations
  - predict the existence/non-existence (1/-1 classification) of each relation for each pair of people

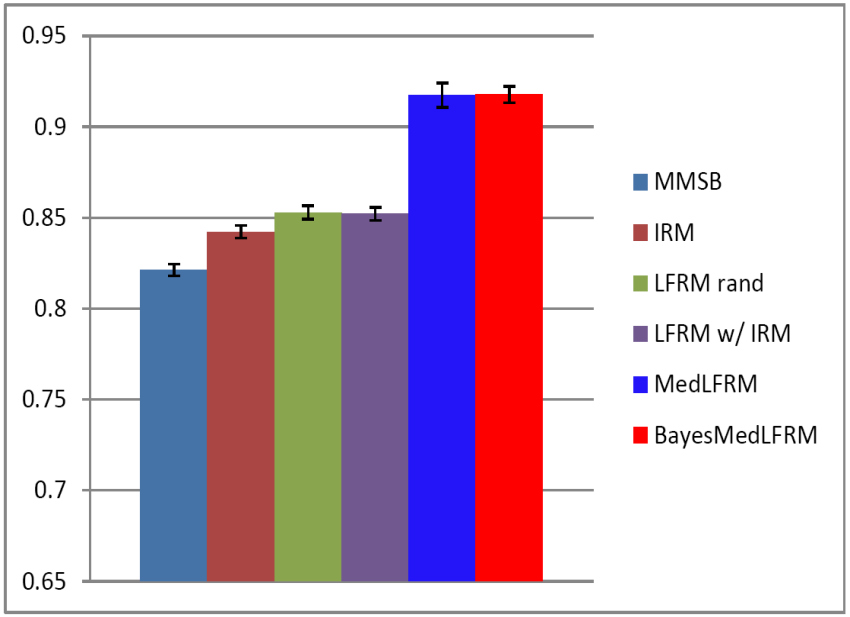


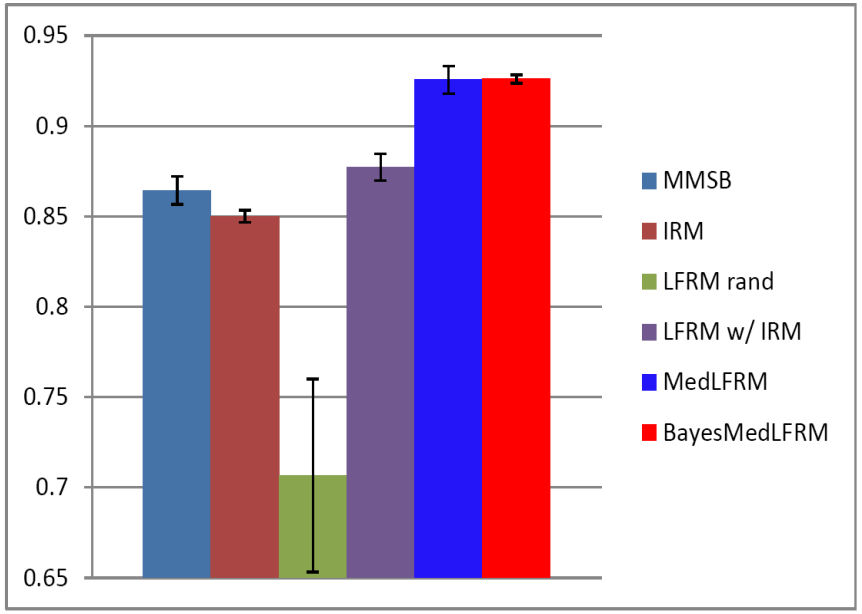
- Coauthor Networks:
  - 234 authors
  - 80% pairs for training; 20% for testing
  - Positive – author pairs that published papers together in train years;
  - Negative – author pairs that didn't publish any papers together in train years

Results on Multi-relation Data



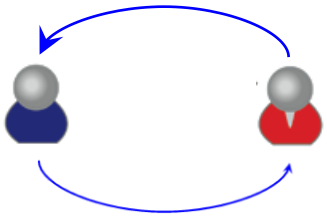
- AUC – area under ROC curve (higher, better)
- Two evaluation settings
  - Single – learn separate models for different relations, and average the AUC scores;
  - Global – learn one common model (i.e., features) for all relations



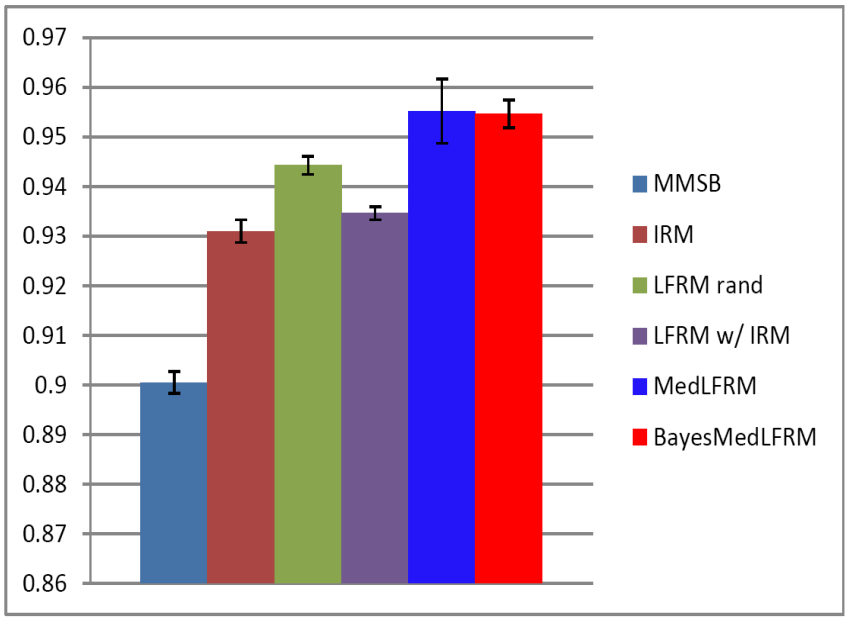


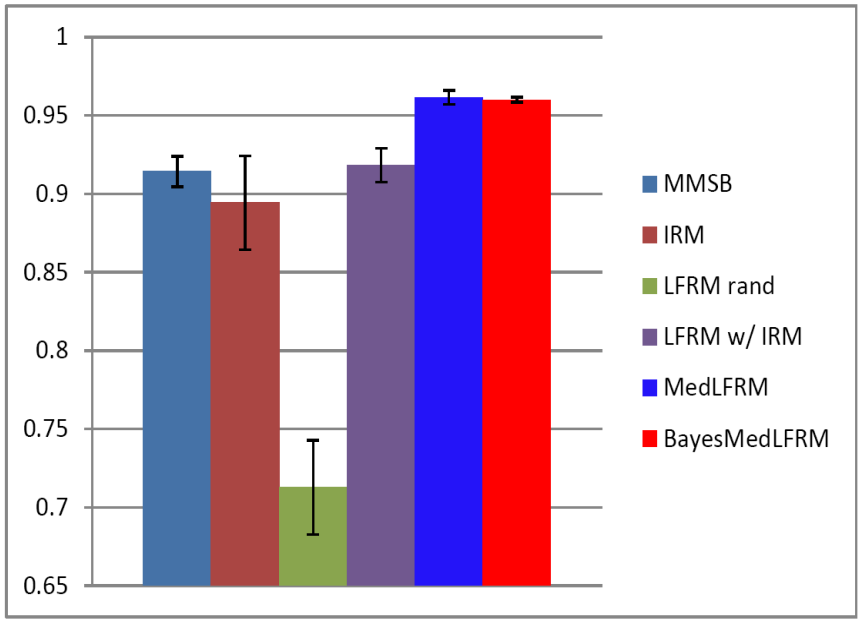
Country Relationships

Results on Multi-relation Data



- AUC – area under ROC curve (higher, better)
- Two evaluation settings
  - Single – learn separate models for different relations, and average the AUC scores;
  - Global – learn one common model (i.e., features) for all relations





Kinship Relationships

Results on Coauthor Networks

- Two model settings:
  - Symmetric – W is a symmetric matrix:

$$Z_i W Z_j^\top = Z_j W Z_i^\top$$

  - Asymmetric – no above constraint

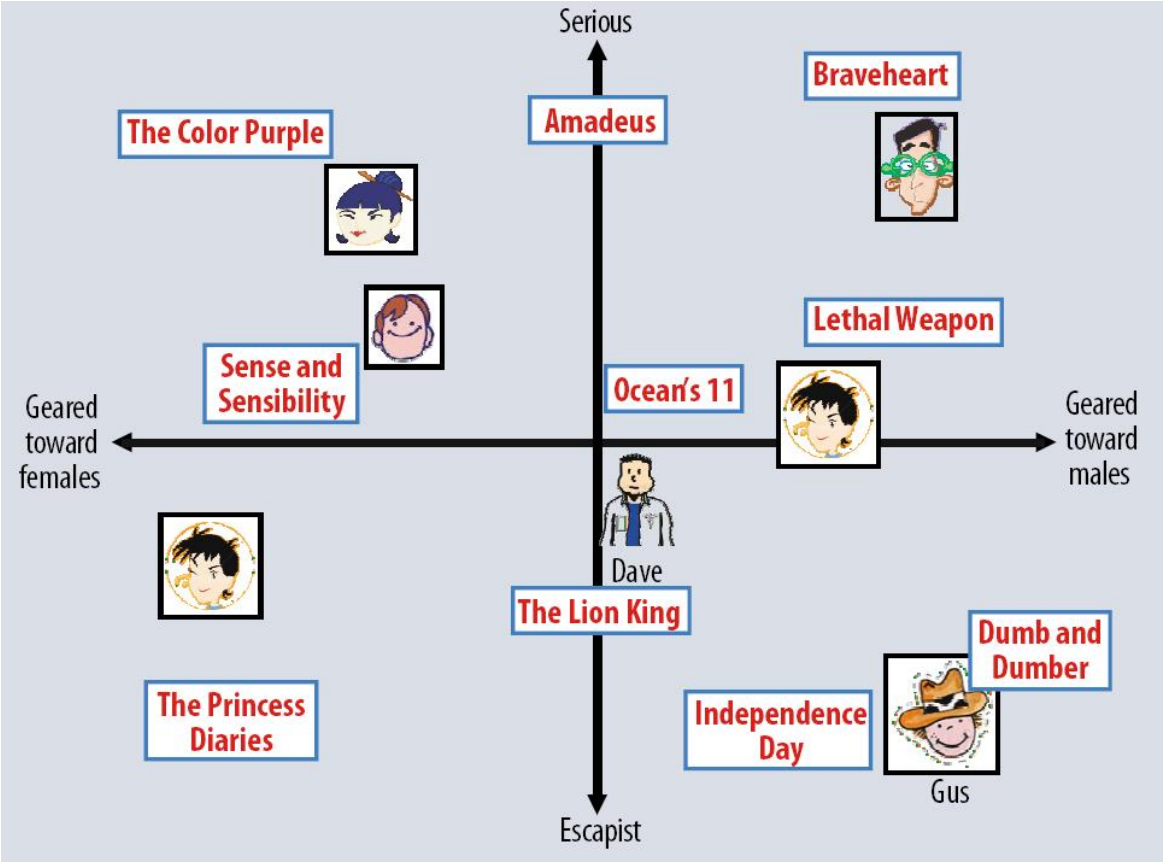
MMSB	0.8705 ± 0.0130
IRM	$0.8906 \pm$ —
LFRM rand	$0.9466 \pm$ —
LFRM w/ IRM	$0.9509 \pm$ —
MedLFRM	$\mathbf{0.9642} \pm 0.0026$
BayesMedLFRM	$\mathbf{0.9636} \pm 0.0036$
Asymmetric MedLFRM	0.9140 ± 0.0130
Asymmetric BayesMedLFRM	0.9146 ± 0.0047

Collaborative Prediction

Collaborative Filtering in Our Life

Latent Factor Methods

- Characterize both items & users on say 20 to 100 factors inferred from the rating patterns

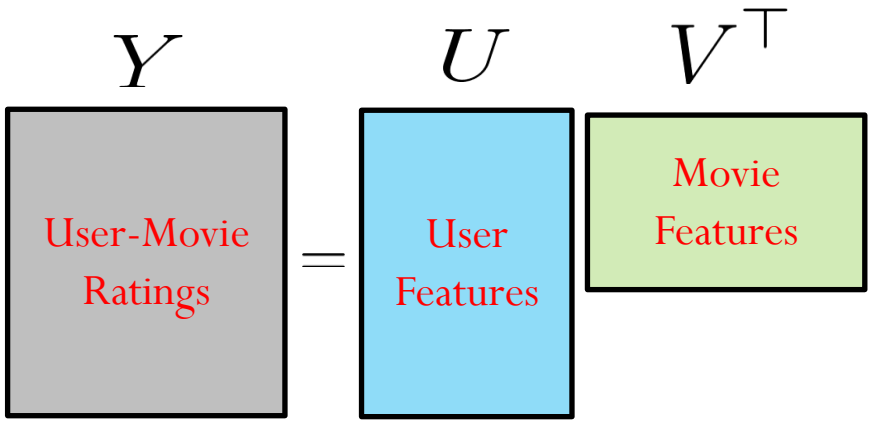


[Y. Koren, R. Bell & C. Volinsky, IEEE, 2009]

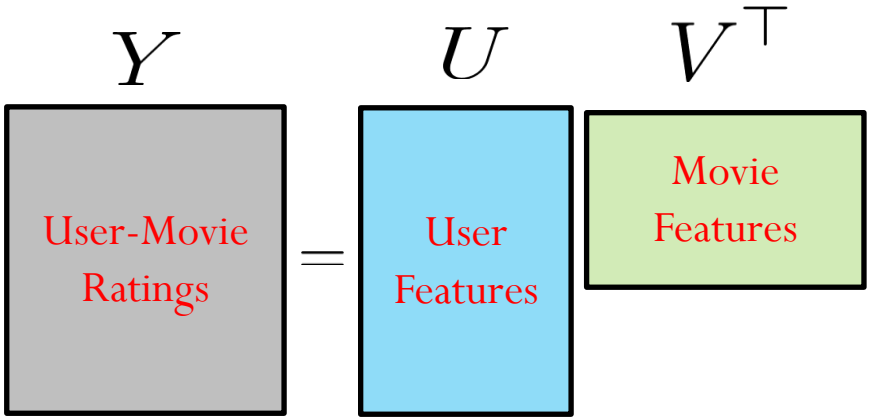
Matrix Factorization

- Some of the most successful latent factor models are based on matrix factorization

item / user	1	2	3	4	...
1	★★★★	?	★★★	★	
2	★★★	?	?	★★	
3	★★★★	★	?	?	
⋮					



Two Key Issues



- How many columns (i.e., features) are sufficient?
  → a stochastic process to infer it from data
- What learning principle is good?
  → large-margin principle to learn classifiers

Nonparametric Max-margin Matrix Factorization for Collaborative Prediction

[Xu, Zhu, & Zhang, NIPS 2012]

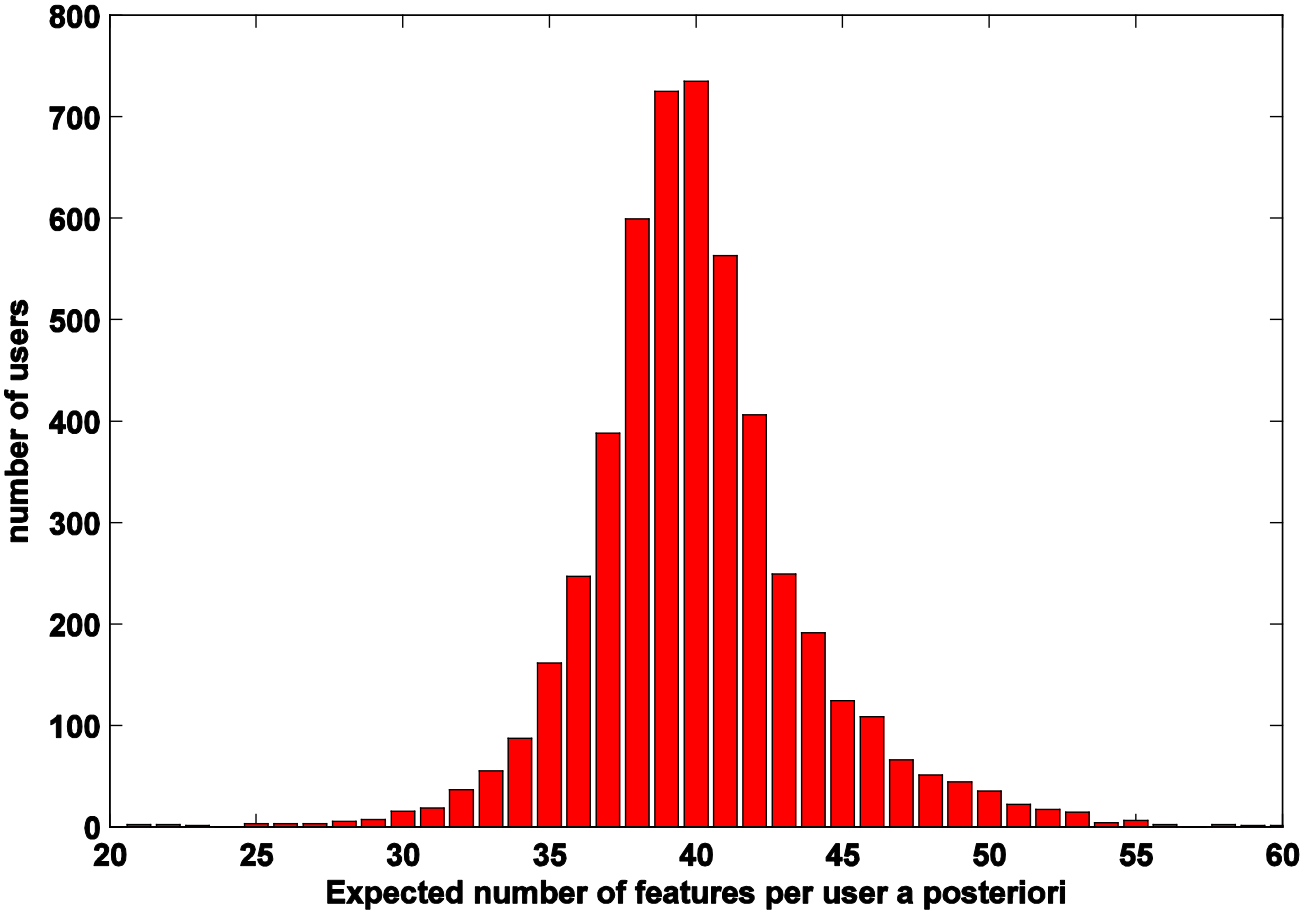
Experiments

- Data sets:
  - **MovieLens**: 1M anonymous ratings of 3,952 movies made by 6,040 users
  - **EachMovie**: 2.8M ratings of 1,628 movies made by 72,916 users
- Overall results on **Normalized Mean Absolute Error (NMAE) (the lower, the better)**

Table 1: NMAE performance of different models on MovieLens and EachMovie.

	MovieLens		EachMovie	
Algorithm	**weak**	**strong**	**weak**	**strong**
M^3F [11]	.4156 ± .0037	.4203 ± .0138	.4397 ± .0006	.4341 ± .0025
PMF [13]	.4332 ± .0033	.4413 ± .0074	.4466 ± .0016	.4579 ± .0016
BPMF [12]	.4235 ± .0023	.4450 ± .0085	.4352 ± .0014	.4445 ± .0005
M^3F^*	.4176 ± .0016	.4227 ± .0072	.4348 ± .0023	.4301 ± .0034
iPM^3F	**.4031** ± .0030	.4135 ± .0109	**.4211** ± .0019	**.4224** ± .0051
$iBPM^3F$	.4050 ± .0029	**.4089** ± .0146	.4268 ± .0029	.4403 ± .0040

Expected Number of Features per User



Fast Sampling Algorithms

- Data augmentation can be used for Gibbs classifiers
- More details are in [Xu, Zhu, & Zhang, ICML2013]

	MovieLens		EachMovie	
Algorithm	**weak**	**strong**	**weak**	**strong**
M^3F	$.4156 \pm .0037$	$.4203 \pm .0138$	$.4397 \pm .0006$	$.4341 \pm .0025$
bcd M^3F	$.4176 \pm .0016$	$.4227 \pm .0072$	$.4348 \pm .0023$	$.4301 \pm .0034$
Gibbs M^3F	$.4037 \pm .0005$	$.4040 \pm .0055$	$.4134 \pm .0017$	$.4142 \pm .0059$
iPM^3F	$.4031 \pm .0030$	$.4135 \pm .0109$	$.4211 \pm .0019$	$.4224 \pm .0051$
Gibbs iPM^3F	$.4080 \pm .0013$	$.4201 \pm .0053$	$.4220 \pm .0003$	$.4331 \pm .0057$

Algorithm	**MovieLens**	**EachMovie**	**Iters**
M^3F	5h	15h	100
bcd M^3F	4h	10h	50
Gibbs M^3F	0.11h	0.35h	50
iPM^3F	4.6h	5.5h	50
Gibbs iPM^3F	0.68h	0.70h	50

30 times faster!

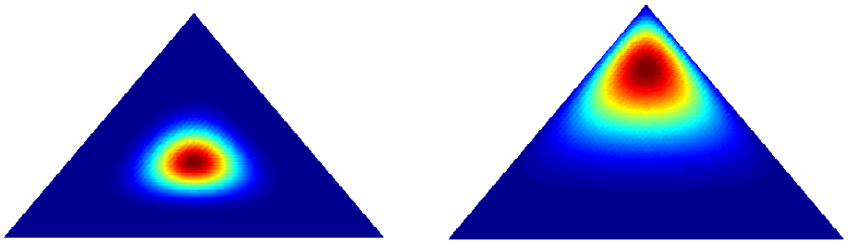
8 times faster!

清華大學 Tsinghua University

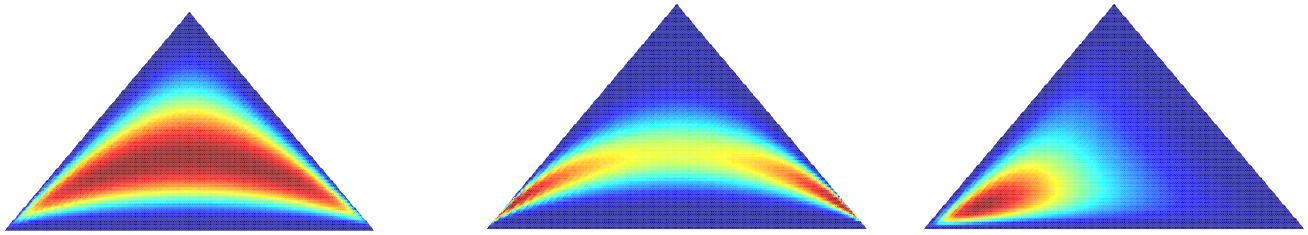
Large-scale Bayesian Inference

Distributed Inference Algorithms

- Logistic-normal prior distribution (Aitchison & Shen, 1980)
  - Dirichlet prior is conjugate but can be too simple



  - Logisitc-normal prior can capture the correlationships

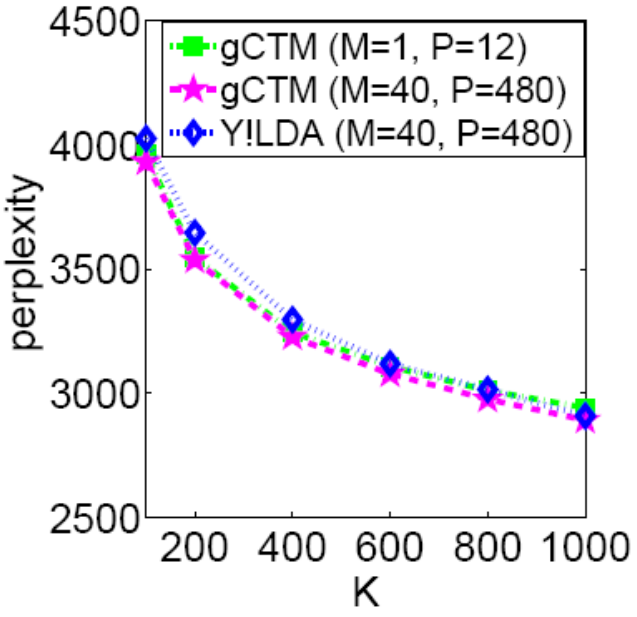


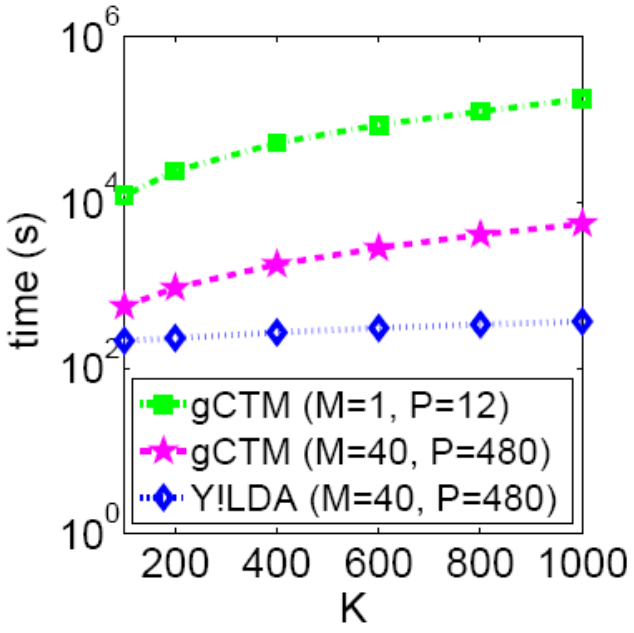
  - But it is non-conjugate to a multinomial likelihood !
  - Variational approximation not scalable (Blei & Lafferty, 2007)

Distributed Inference Algorithms

- Leverage big clusters
- Allow learning big models that can't fit on a single machine

An example of scaling up logistic-normal topic models:





- 40 machines;
- 480 CPU cores

- 0.285M NYTimes pages
- K = 200 ~ 1000

[Chen, Zhu, Wang, Zheng, & Zhang, NIPS 2013]

Distributed Inference Algorithms

- Leverage big clusters
- Allow learning big models that can't fit on a single machine

An example of scaling up logistic-normal topic models:

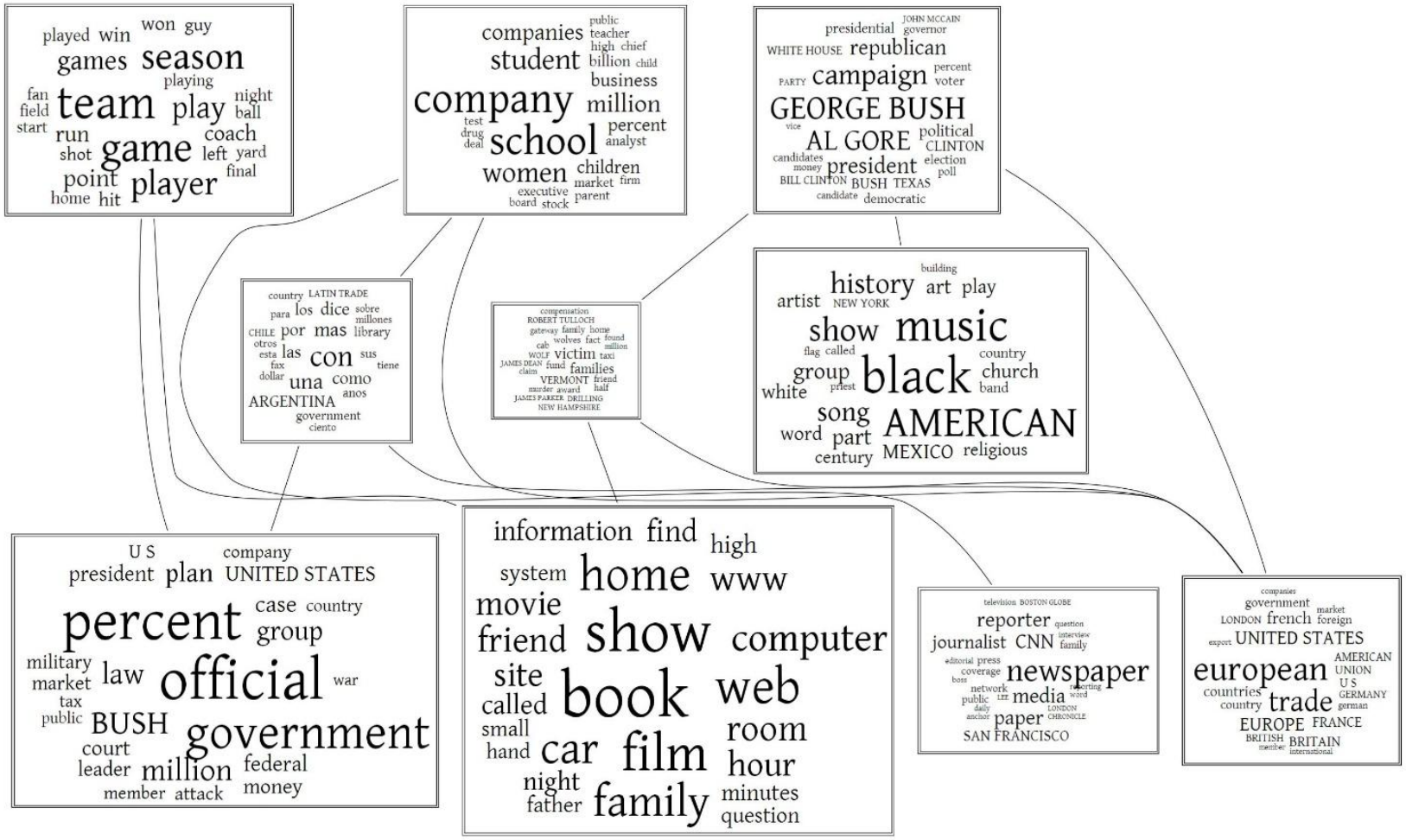
data set	D	K	vCTM	gCTM
NIPS	1.2K	100	1.9 hr	8.9 min
20NG	11K	200	16 hr	9 min
NYTimes	285K	400	N/A*	0.5 hr
Wiki	6M	1000	N/A*	17 hr

*not finished within 1 week.

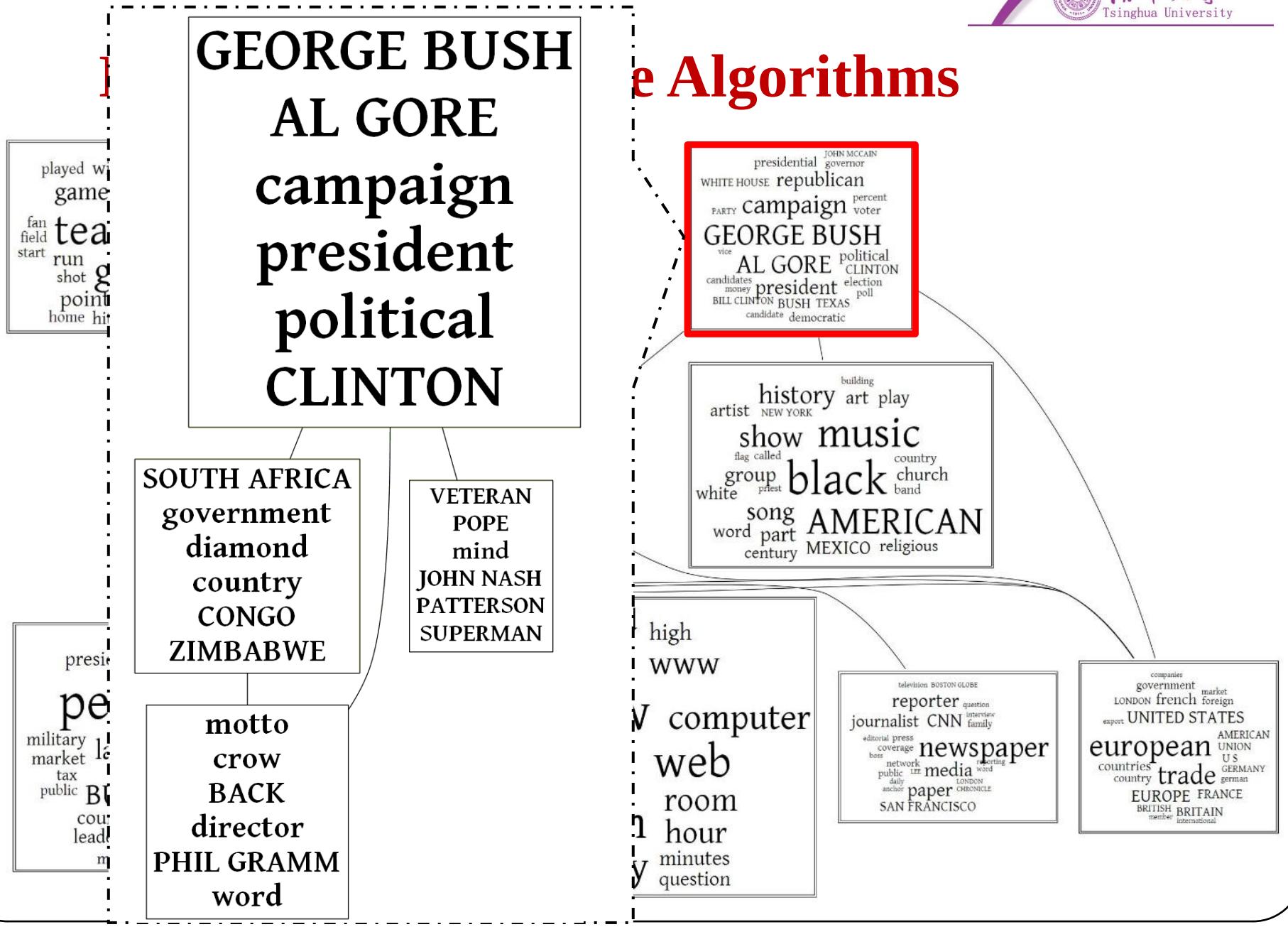
[Chen, Zhu, Wang, Zheng, & Zhang, NIPS 2013]

Distributed Inference Algorithms

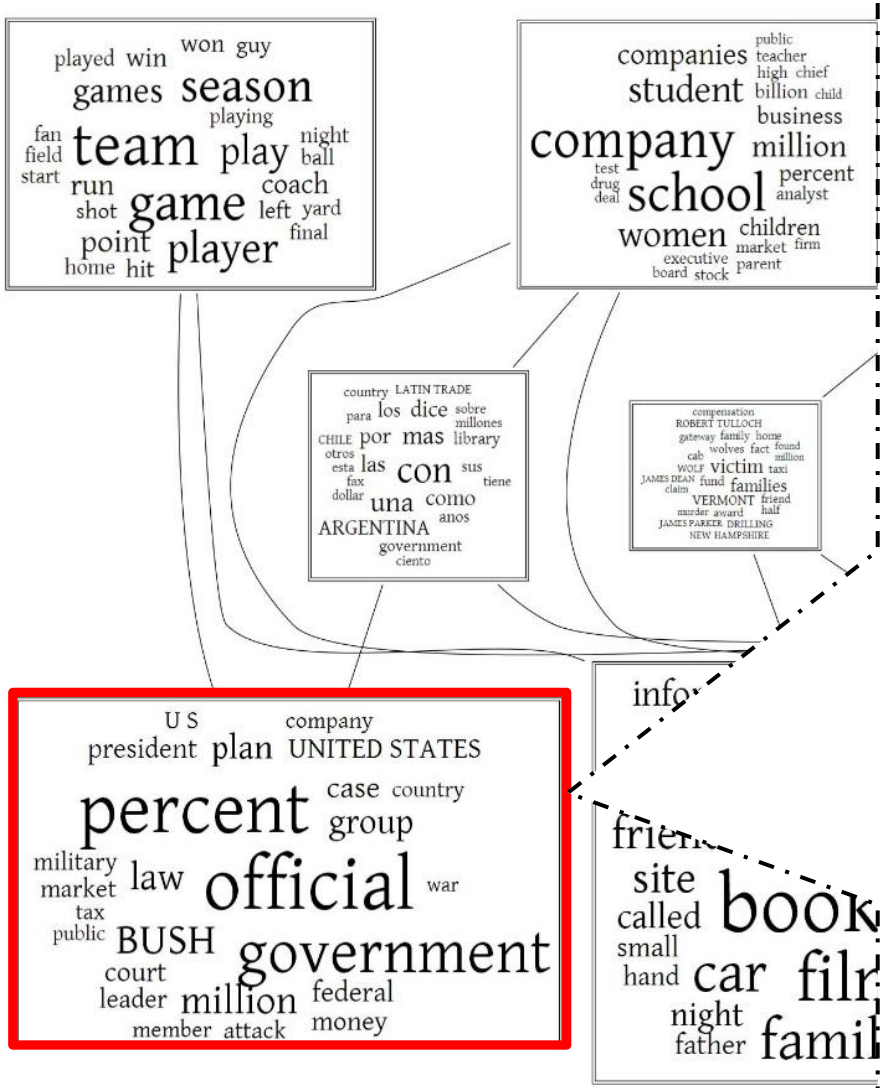
- Demo with 1,000 topics: **http://ml-thu.net/~scalable-ctm**

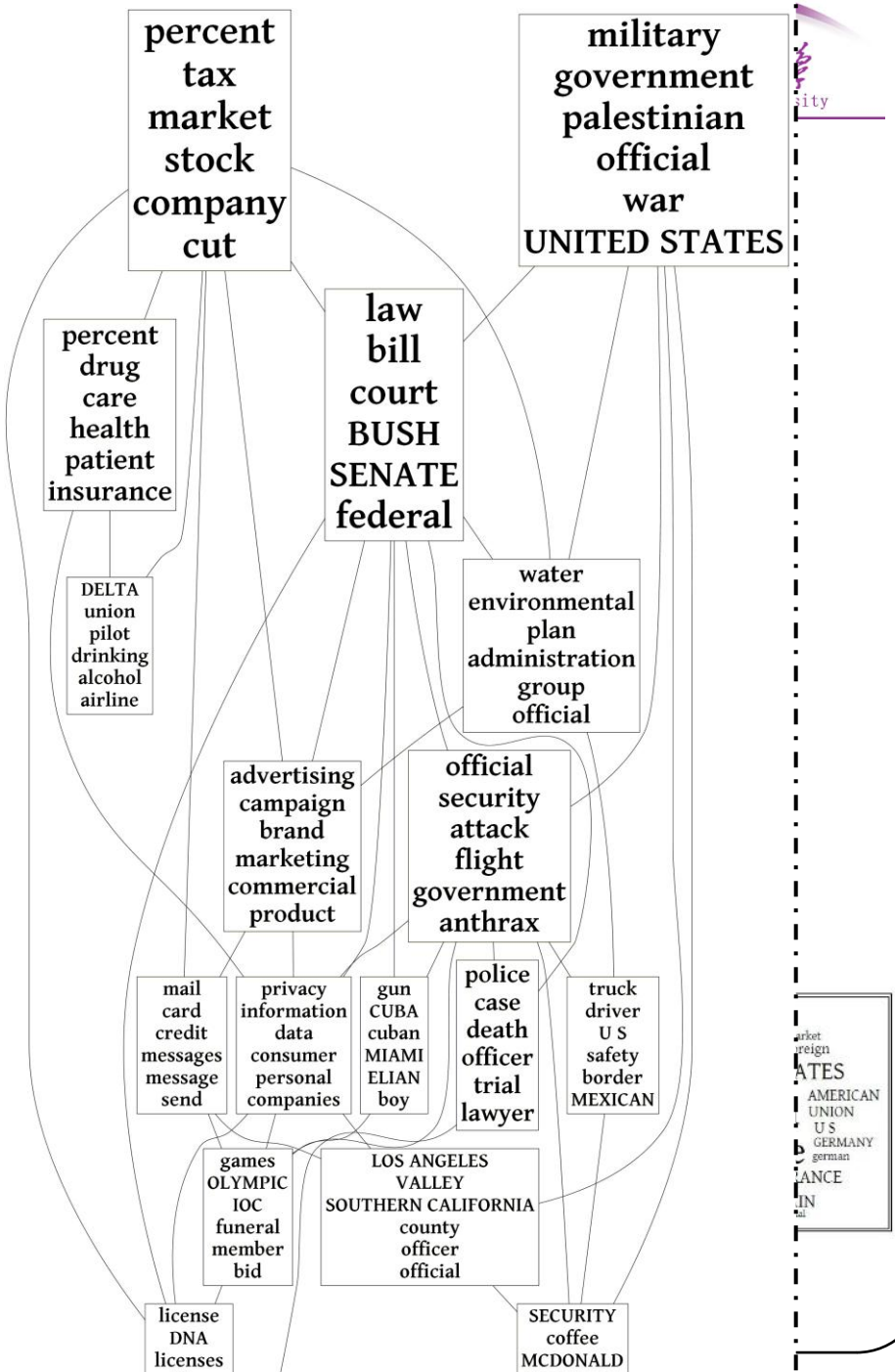


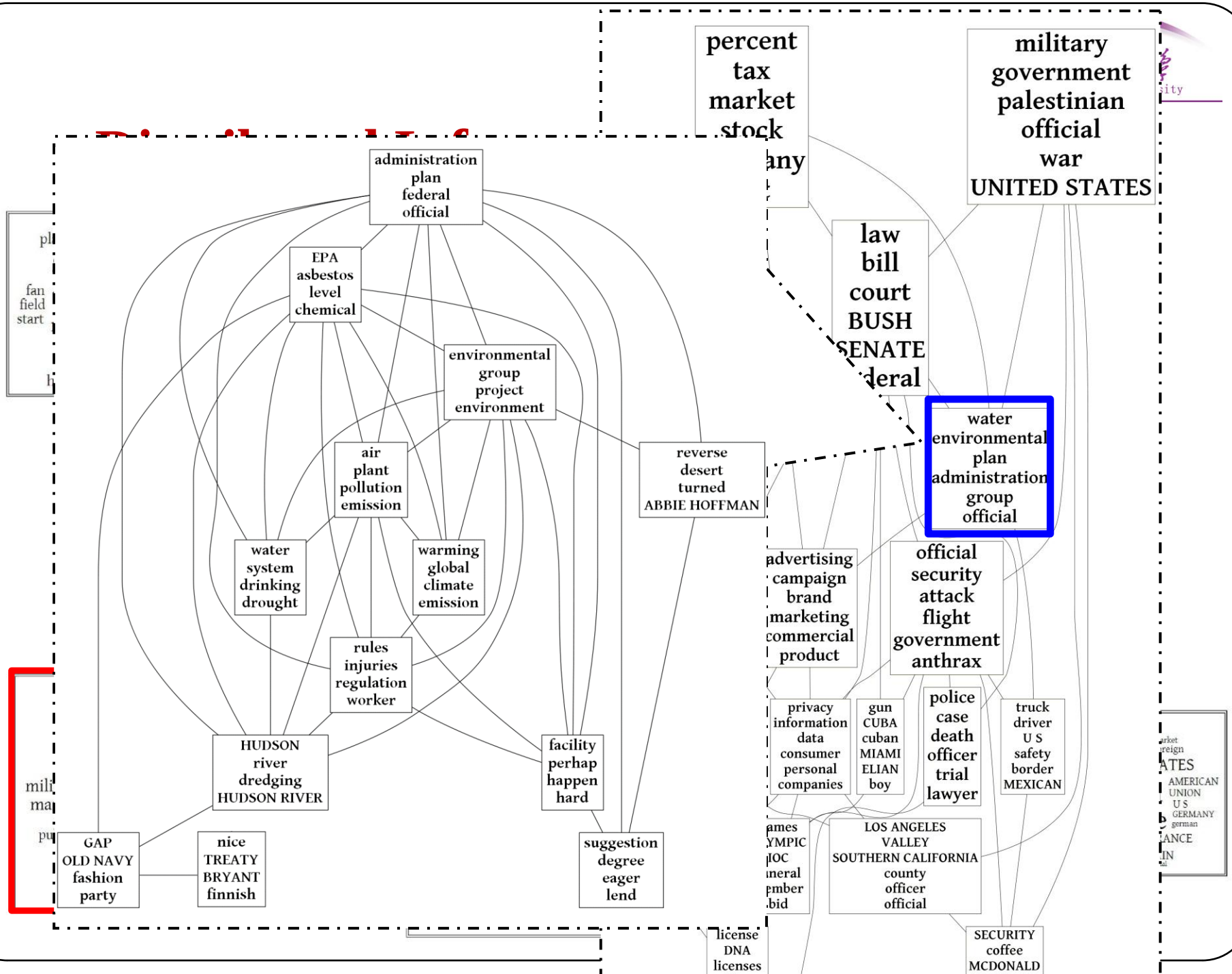
e Algorithms



Distributed Inferenc






percent
tax
market
stock
military
government
palestinian
official
war
UNITED STATES
administration
plan
federal
official
EPA
asbestos
level
chemical
law
bill
court
BUSH
SENATE
environmental
group
project
environment
water
environmental
plan
administration
group
official
air
plant
pollution
emission
reverse
desert
turned
ABBIE HOFFMAN
water
system
drinking
drought
warming
global
climate
emission
advertising
campaign
brand
marketing
commercial
product
official
security
attack
flight
government
anthrax
rules
injuries
regulation
worker
privacy
information
data
consumer
personal
companies
gun
CUBA
cuban
MIAMI
ELIAN
boy
police
case
death
officer
trial
lawyer
truck
driver
U S
safety
border
MEXICAN
HUDSON
river
dredging
HUDSON RIVER
facility
perhap
happen
hard
GAP
OLD NAVY
fashion
party
nice
TREATY
BRYANT
finnish
suggestion
degree
eager
lend
LOS ANGELES
VALLEY
SOUTHERN CALIFORNIA
county
officer
official
SECURITY
coffee
MCDONALD
license
DNA
licenses

Summary

- RegBayes: bridging the gap between Bayesian methods, learning and optimization
- RegBayes with Max-margin Posterior Regularization
  - supervised topic models
  - classification and multi-task learning
  - link prediction
  - collaborative prediction
- Averaging classifiers suitable for variational inference
- Gibbs classifiers suitable for MCMC sampling with DA
- Large-scale inference is a challenge

Acknowledgements

- Collaborators:
  - Prof. Bo Zhang (Tsinghua)、Prof. Eric P. Xing (CMU)、Prof. Li Fei-Fei (Stanford)
  - Amr Ahmed (CMU), Ning Chen (Tsinghua), Ni Lao (CMU), Seunghak Lee (CMU), Li-jia Li (Stanford), Xiaojiang Liu (USTC), Xiaolin Shi (Stanford), Hao Su (Stanford), Yuandong Tian (CMU).
- Students at Tsinghua:
  - Aonan Zhang Minjie Xu Hugh Perkins

  Jianfei Chen, Bei Chen, Shike Mei, Xun Zheng, Fei Xia, Zi Wang, Tianlin Shi, Yining Wang, Li Zhou, etc.
- Funding:

973 CHINA

NSFC

TSINGHUA UNIVERSITY 1911

Microsoft Research 微软亚洲研究院

清華大學 Tsinghua University

Thanks!

Some code available at:

http://www.ml-thu.net/~jun