RWTH AACHEN UNIVERSITY

Advanced Machine Learning Lecture 13

Hierarchical Dirichlet Processes

10.12.2012

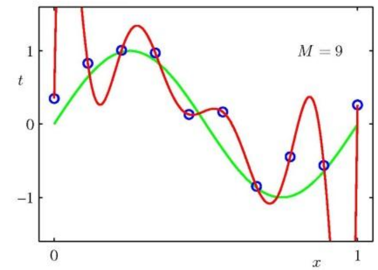
Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de/

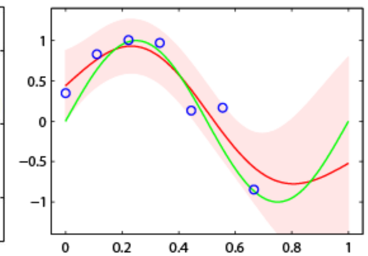
leibe@vision.rwth-aachen.de

This Lecture: *Advanced Machine Learning*

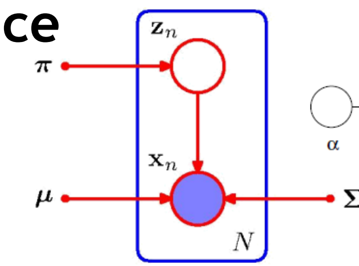
- Regression Approaches
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Kernels (Kernel Ridge Regression)
  - Gaussian Processes

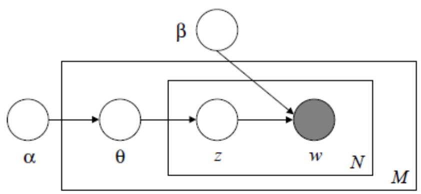
$$f : \mathcal{X} \rightarrow \mathbb{R}$$





- Bayesian Estimation & Bayesian Non-Parametrics
  - Prob. Distributions, Approx. Inference
  - Mixture Models & EM
  - Dirichlet Processes
  - Latent Factor Models
  - Beta Processes





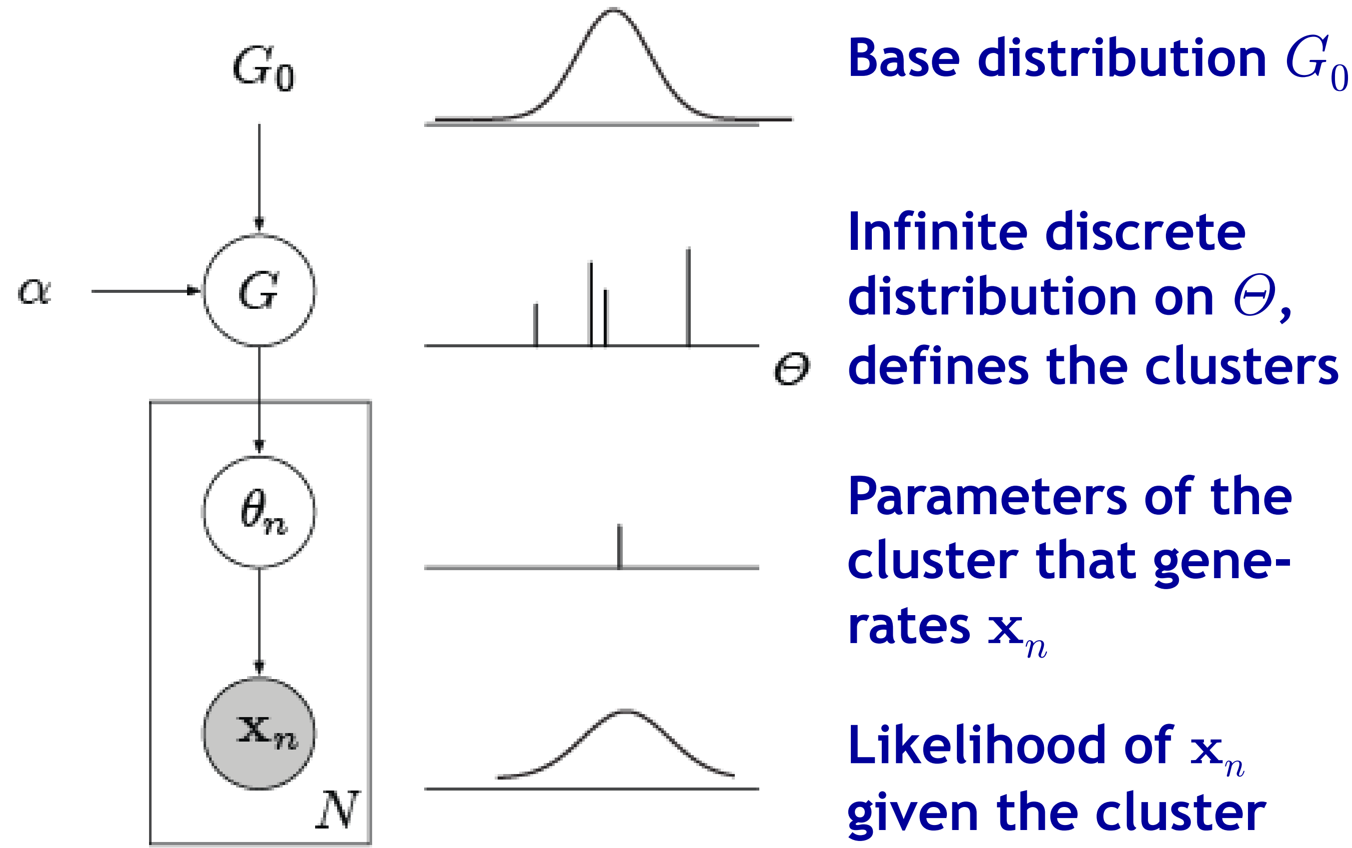
- SVMs and Structured Output Learning
  - SV Regression, SVDD
  - Large-margin Learning

$$f : \mathcal{X} \rightarrow \mathcal{Y}$$

Topics of This Lecture

- Applying DPs
  - Recap: DPs
  - Efficient Gibbs sampling
- Hierarchical Dirichlet Processes
  - Definition
  - Properties
  - Chinese Restaurant Franchise
  - Gibbs sampling for HDPs
- Applications
  - Topic modeling

Recap: Dirichlet Process Mixture Models



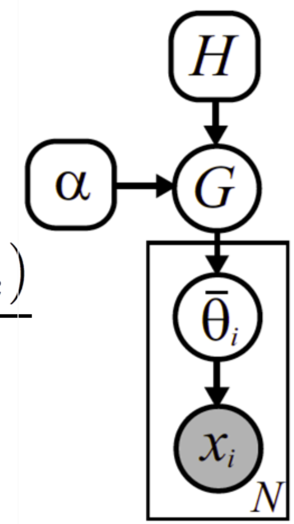
- **Distributional form**
  - Explicit representation of the DP through the node G.
  - Useful when we want to use the DPMM's predictive distribution.

Image sources: Yee Whye The

Recap: Pólya Urn Scheme

- Pólya Urn scheme
  - Simple generative process for the predictive distribution of a DP
  - Consider a set of N observations $\bar{\theta}_n \sim G$ taking K distinct values $\{\theta_k\}_{k=1}^{K}$. The predictive distribution of the next observation is then

$$p(\bar{\theta}_N = \theta | \bar{\theta}_{1:N-1}, \alpha, H) = \frac{\alpha H(\theta) + \sum_{k=1}^{K} N_k \delta(\theta, \theta_k)}{N - 1 + \alpha}$$



- Remarks
  - This procedure can be used to sample observations from a DP without explicitly constructing the underlying mixture.

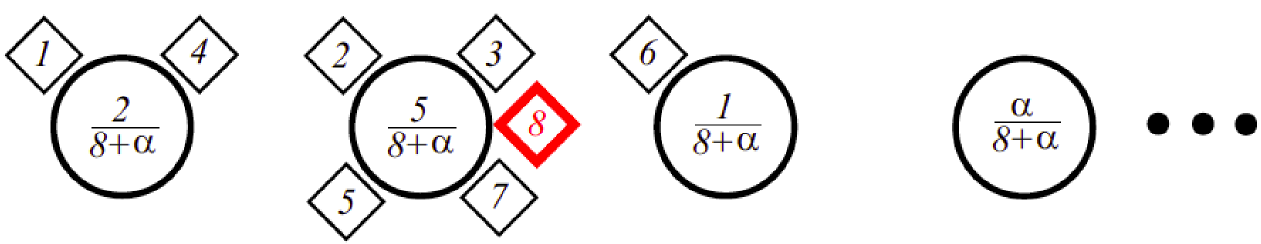
$\Rightarrow$ DPs lead to simple predictive distributions that can be evaluated by caching the number of previous observations taking each distinct value.

Recap: Chinese Restaurant Process (CRP)

- **Procedure**
  - Imagine a Chinese restaurant with an infinite number of tables, each of which can seat an infinite number of customers.
  - The 1st customer enters and sits at the first table.
  - The N^{th} customer enters and sits at table

$$\begin{cases} k & \text{with prob } \dfrac{N_k}{N-1+\alpha} \quad \text{for } k = 1,\ldots,K \\ K+1 & \text{with prob } \dfrac{\alpha}{N-1+\alpha} \quad \text{(new table)} \end{cases}$$

where N_k is the number of customers already sitting at table k.



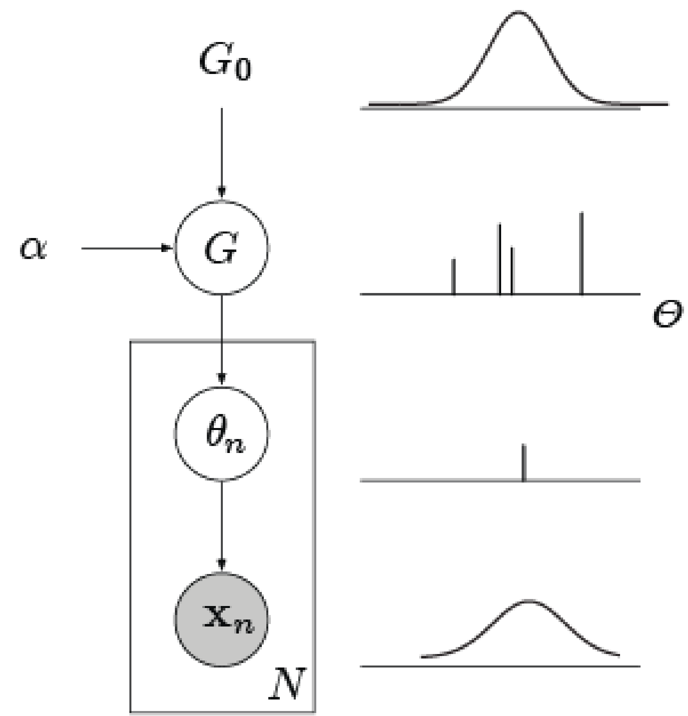
Recap: CRPs & De Finetti's Theorem

- **Putting all of this together…**
  - De Finetti's theorem tells us that the CRP has an underlying mixture distribution with a prior distribution over measures.
  - The Dirichlet Process is the De Finetti mixing distribution for the CRP.

- **Graphical model visualization**
  - This means, when we integrate out G, we get the CRP:

$$p(\theta_1,\ldots,\theta_N) = \int \prod_{n=1}^{N} p(\theta_n|G)\mathrm{d}P(G)$$

$\Rightarrow$ ***If the DP is the prior on G, then the CRP defines how points are assigned to clusters when we integrate out G.***



Slide adapted from Kurt Miller, Mike Jordan

Image source: Kurt Miller

Recap: CRPs and Efficient Inference

- **Taking advantage of exchangeability…**
  - In clustering applications, we are ultimately interested in the cluster assignments $\mathbf{z}_1,\ldots,\mathbf{z}_N$.
  - Equivalent question in the CRP: Where should customer n sit, conditioned on the seating choices of all the other customers?
    - This is easy when customer n is the last customer to arrive:

$$p(\mathbf{z}_N = \mathbf{z} | \mathbf{z}_1, \ldots, \mathbf{z}_{N-1}, \alpha) = \frac{1}{N-1+\alpha}\left(\sum_{k=1}^{K} N_k \delta(\mathbf{z}, k) + \alpha \delta(\mathbf{z}, \bar{k})\right)$$

    - (Seemingly) hard otherwise…

⇒ ***Because of exchangeability, we can always swap customer n with the final customer and use the above formula!***

⇒ **We'll use this for efficient Gibbs sampling later on…**

Recap: Stick-Breaking Construction

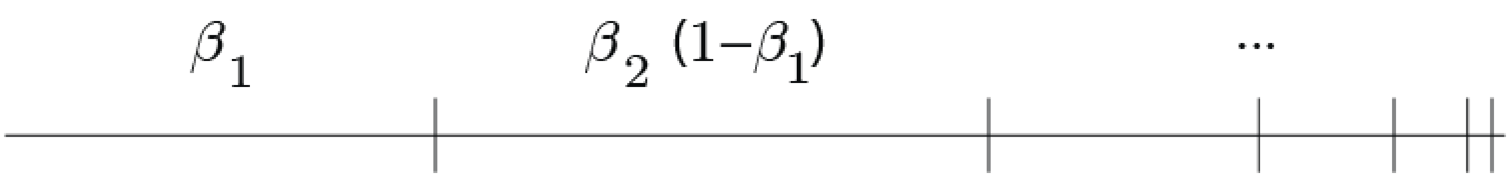
- **Explicit construction for the weights in DP realizations**
  - **Define an infinite sequence of random variables**

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

  - **Then define an infinite sequence of mixing proportions as**

$$\begin{aligned} \pi_1 &= \beta_1 \\ \pi_k &= \beta_k \prod_{l=1}^{k-1} (1 - \beta_l) \qquad k = 2, 3, \ldots \end{aligned}$$

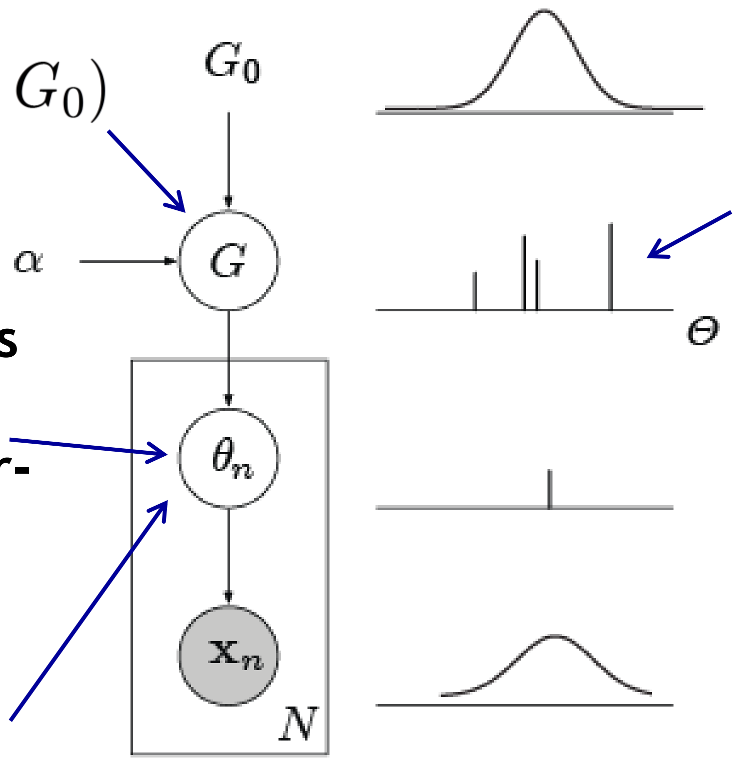
  - **This can be viewed as breaking off portions of a stick**



  - **When the π_k are drawn this way, we can write $\pi \sim \mathrm{GEM}(\alpha)$. (where GEM stands for Griffiths, Engen, McCloskey)**

Summary: Pólya Urns, CRPs, and Stick-Breaking

$G \sim \mathrm{DP}(\alpha, G_0)$



The **Pólya urn** describes the **predictive distribution** of θ when G is marginalized out

The **CRP** describes the **partitions** of θ when G is marginalized out

The **Stick-Breaking Process** describes the **partition weights**

Summary: Pólya Urns, CRPs, and Stick-Breaking

- Better understanding of the properties of DPs
  - All three schemes lead to proofs that DPs exist.
  - Using the Polya urn scheme, we showed that we can sample from DPs without constructing the underlying mixture explicitly.
  - Using the Chinese Restaurant Process, we showed that the expected number of clusters grows with $\mathcal{O}(\alpha \log N)$.
  - Using the Stick-Breaking Construction, we showed that Dirichlet measures are discrete with probability one.

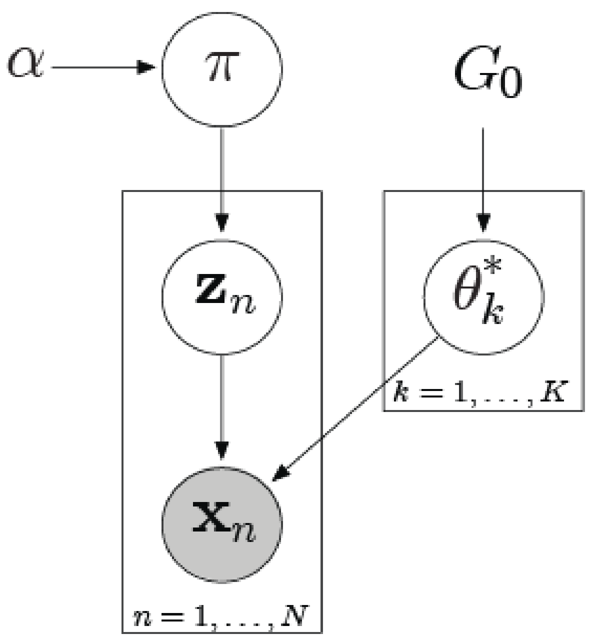
- Uses for inference
  - All three schemes can be used to construct efficient inference methods.
  - We will mostly look at Gibbs samplers that are derived from the CRP.

Topics of This Lecture

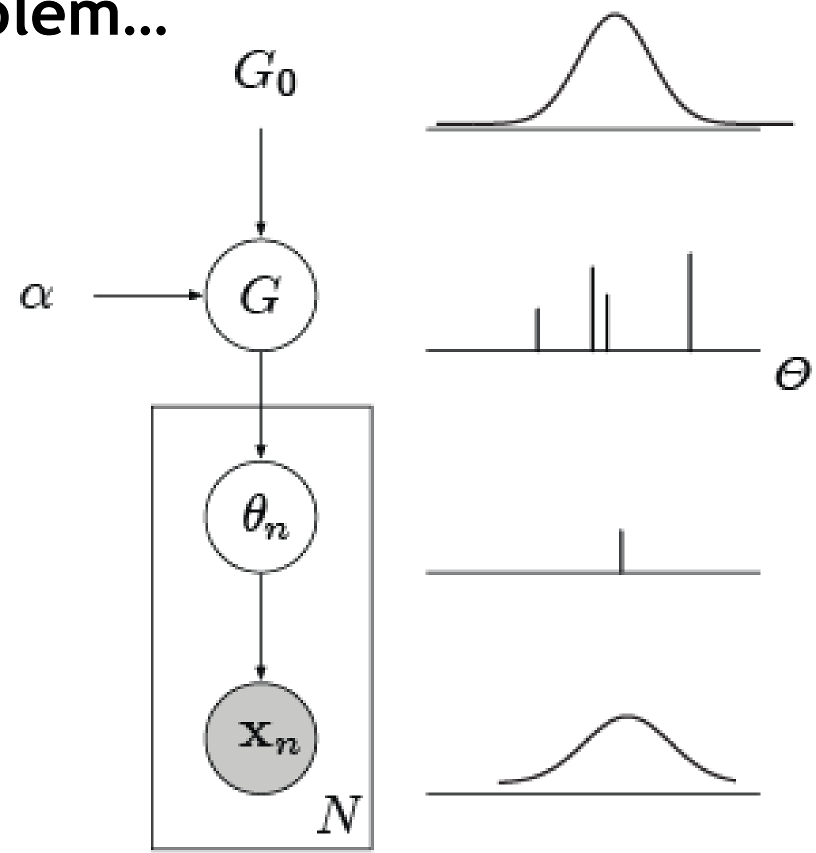
- **Applying DPs**
  - Recap: DPs
  - Efficient Gibbs sampling
- **Hierarchical Dirichlet Processes**
  - Definition
  - Properties
  - Chinese Restaurant Franchise
  - Gibbs sampling for HDPs
- **Applications**
  - Topic modeling

Dirichlet Process Mixture Models

- Back to the clustering problem…



Indicator variable representation



Distributional form

Image sources: Yee Whye The, Kurt Miller

Collapsed DP Mixture Sampler

- Efficient algorithm
  - Generalize the collapsed (Rao-Blackwellized) Gibbs sampler we derived for finite mixtures
  - As before, sample the indicator variables $\mathbf{z}_n$ assigning observations to latent clusters, marginalizing mixture weights π_k and parameters θ_k.
  - Assume the cluster priors $H(\lambda)$ are conjugate.
- Derivation
  - The model implies the factorization

$$p(\mathbf{z}_n|\mathbf{z}_{-n},\mathbf{x},\alpha,\lambda) \propto \underbrace{p(\mathbf{z}_n|\mathbf{z}_{-n},\alpha)}_{\text{Prior on partitions expressed by the CRP!}} p(\mathbf{x}_n|\mathbf{z},\mathbf{x}_{-n},\lambda)$$

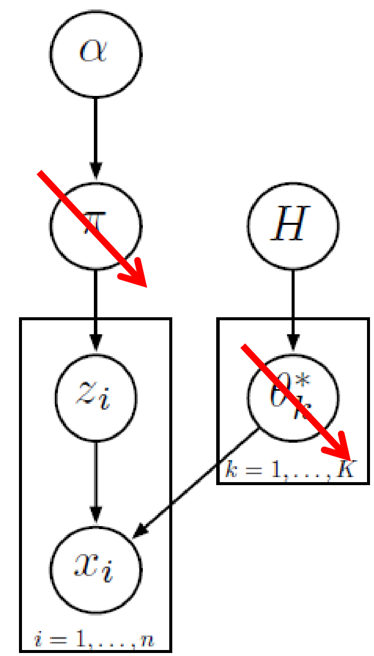


Image source: Yee Whye Teh

Collapsed DP Mixture Sampler

- Derivation (cont'd)
  - Exchangeability: Think of $\mathbf{z}_n$ as the last observation in sequence

$$p(\mathbf{z}_n|\mathbf{z}_{-n},\alpha) = \frac{1}{N-1+\alpha}\left(\sum_{k=1}^{K} N_{-n,k}\delta(\mathbf{z}_n,k) + \alpha\delta(\mathbf{z}_n,\bar{k})\right)$$

  - The predictive likelihood of $\mathbf{x}_n$ is computed as for finite mixtures:

$$p(\mathbf{x}_n|\mathbf{z}_n = k, \mathbf{z}_{-n}, \mathbf{x}_{-n}, \lambda) = p(\mathbf{x}_n|\{\mathbf{x}_m|z_{mk}=1, m\neq n\},\lambda)$$

  - New clusters $\bar{k}$ are based on the predictive likelihood implied by the hyperparameters λ

$$p(\mathbf{x}_n|\mathbf{z}_n = \bar{k}, \mathbf{z}_{-n}, \mathbf{x}_{-n}, \lambda) = p(\mathbf{x}_n|\lambda) = \int_{\Theta} p(\mathbf{x}_n|\theta)h(\theta|\lambda)\mathrm{d}\theta$$

Collapsed DP Mixture Sampler

- **Algorithm**
  1. Sample a random permutation $\tau(\cdot)$ of the integers $\{1,\ldots,N\}$.
  2. Set $\alpha = \alpha^{(t\text{-}1)}$ and $\mathbf{z} = \mathbf{z}^{(t\text{-}1)}$. For each $n \in \{\tau(1),\ldots,\tau(N)\}$, sequentially resample $\mathbf{z}_n$ as follows
     a) For each of the K existing clusters, determine the predictive likelihood

     $$p_k(\mathbf{x}_n|\mathbf{z}_{-n},\lambda) = p(\mathbf{x}_n|\{\mathbf{x}_m|z_{mk}=1, m \neq n\},\lambda)$$

     Also determine the likelihood $p_{\bar{k}}(\mathbf{x}_n)$ of a potential new cluster $\bar{k}$

     $$p_{\bar{k}}(\mathbf{x}_n|\mathbf{z}_{-n},\lambda) = p(\mathbf{x}_n|\lambda) = \int_{\Theta} p(\mathbf{x}_n|\theta)h(\theta|\lambda)\mathrm{d}\theta$$

     b) Sample a new assignment $\mathbf{z}_n$ from the multinomial distribution

     $$\mathbf{z}_n \sim \frac{z_{n\bar{k}}\alpha p_{\bar{k}}(\mathbf{x}_n|\mathbf{z}_{-n},\lambda) + \sum_{k=1}^{K} z_{nk}N_{-n,k}p_k(\mathbf{x}_n|\mathbf{z}_{-n},\lambda)}{\alpha p_{\bar{k}}(\mathbf{x}_n|\mathbf{z}_{-n},\lambda) + \sum_{j=1}^{K}(N_{-n,j}p_j(\mathbf{x}_n|\mathbf{z}_{-n},\lambda)}$$

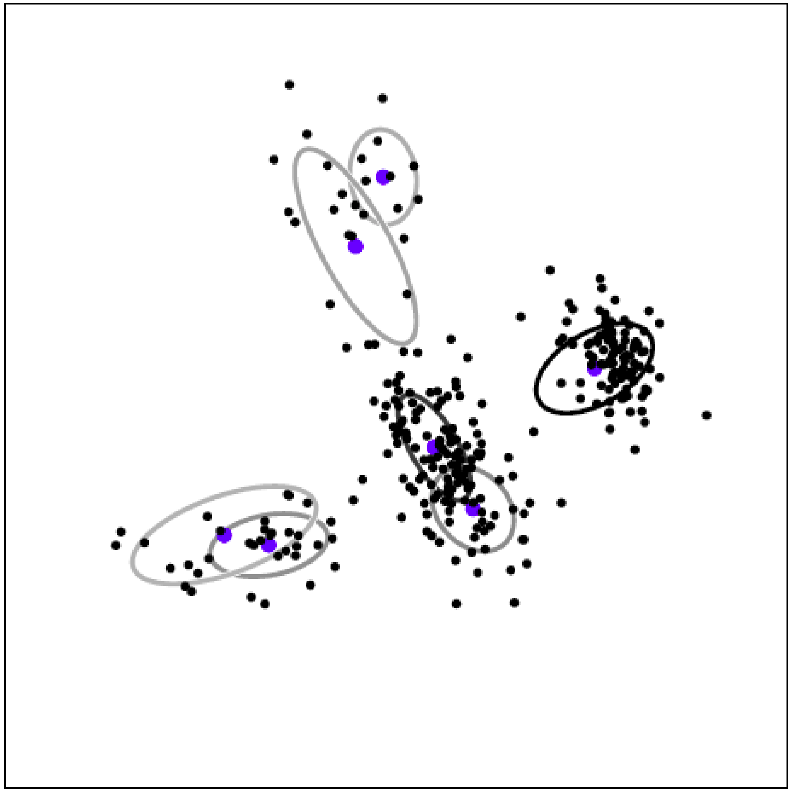
     c) Update cached sufficient statistics to reflect assignment z_{nk}. If $\mathbf{z}_n = \bar{k}$, create a new cluster and increment K.

Collapsed DP Mixture Sampler (cont'd)

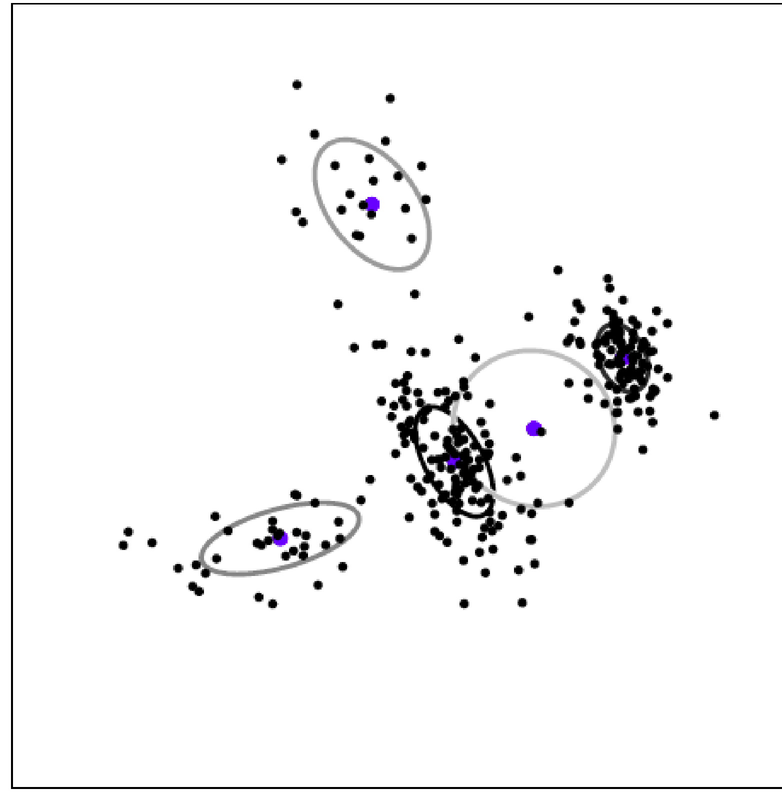
- Algorithm (cont'd)
  3. Set $\mathbf{z}^{(t)} = \mathbf{z}$. Optionally, mixture parameters for the K currently instantiated clusters may be sampled as in step 3 of the standard finite mixture sampler.
  4. If any current clusters are empty ($N_k = 0$), remove them and decrement K accordingly.

- Remarks
  - Algorithm is valid if the cluster priors $H(\lambda)$ are conjugate.
  - Cluster assignments $\mathbf{z}^{(t)}$ produced by Gibbs sampler provide estimates $K^{(t)}$ of the number of clusters underlying the observations $\mathbf{X}$, as well as their associated parameters.
  - Predictions based on samples average over mixtures of varying size, avoiding difficulties in selecting a single model.

Collapsed DP Sampler: 2 Iterations

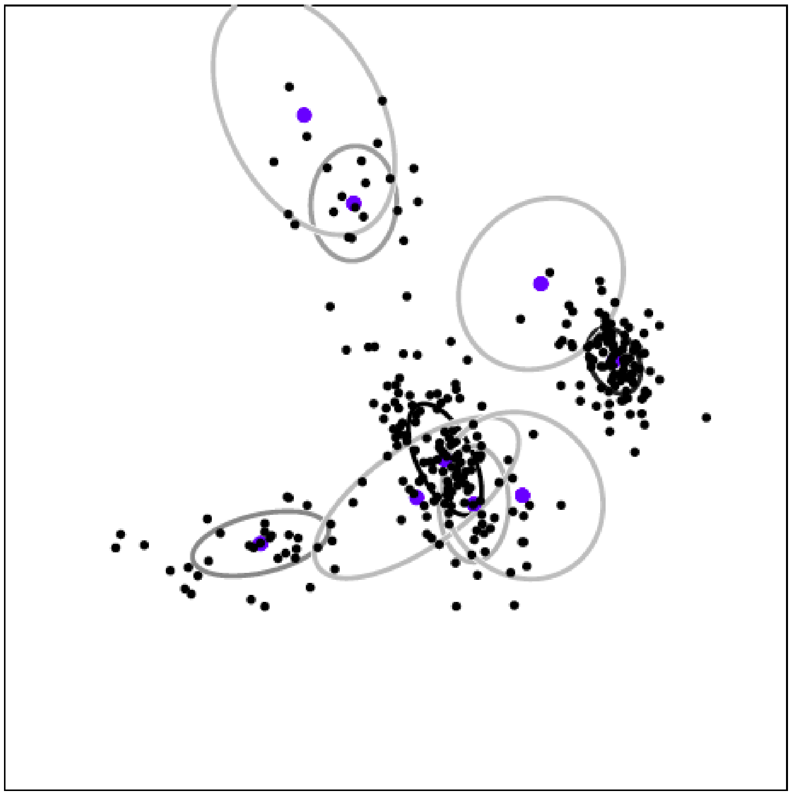


log p(x I π, θ) = –462.25

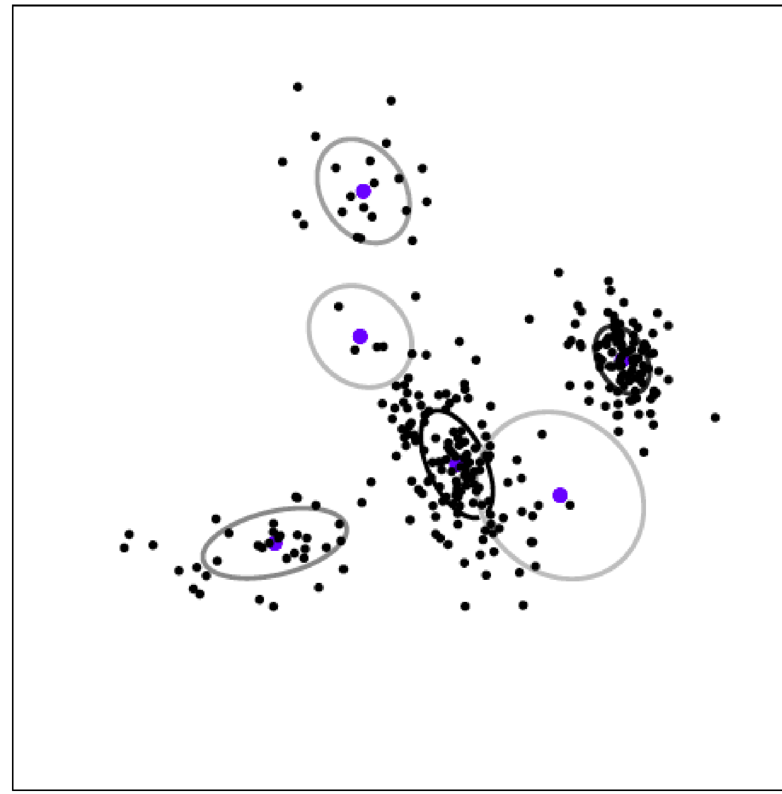


log p(x I π, θ) = –399.82

Collapsed DP Sampler: 10 Iterations

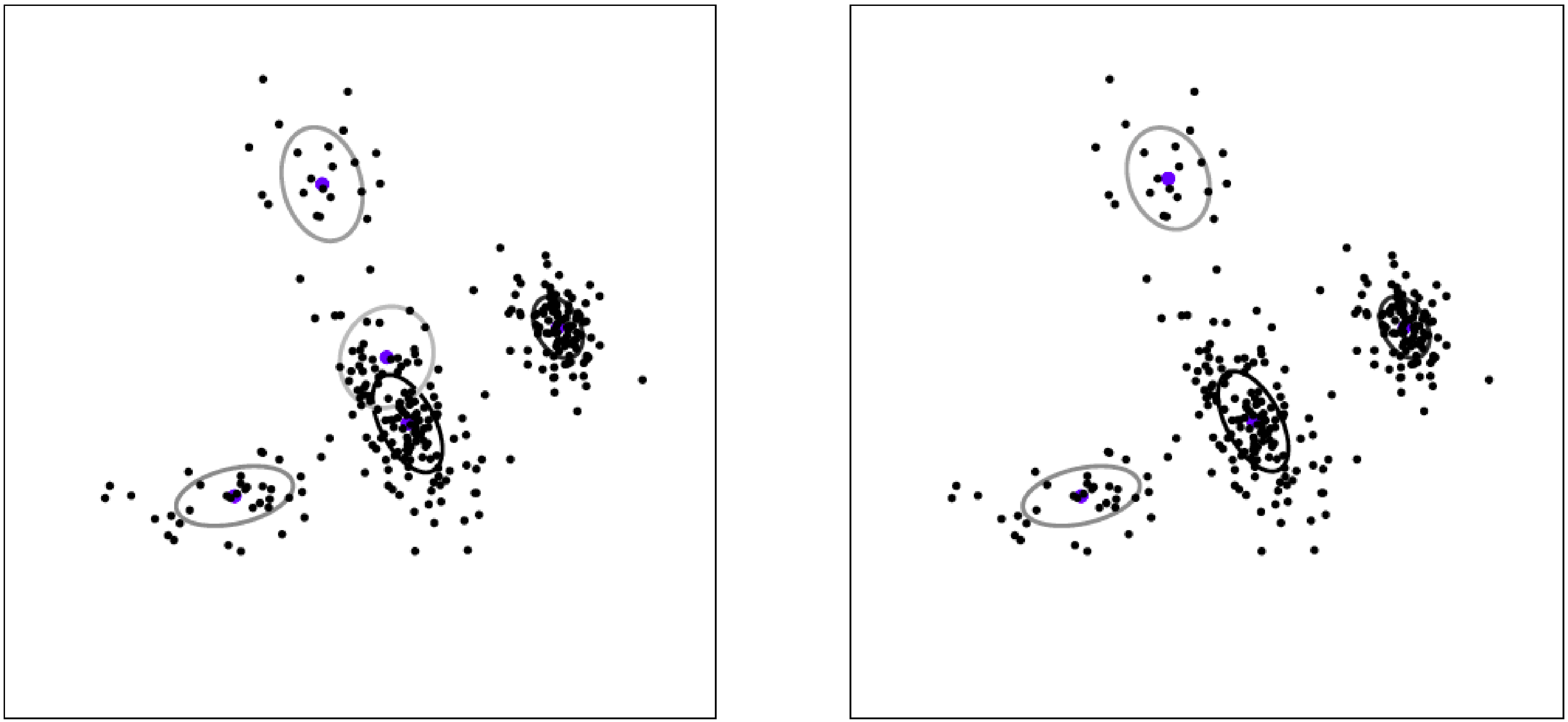


log p(x | π, θ) = –398.32



log p(x | π, θ) = –399.08

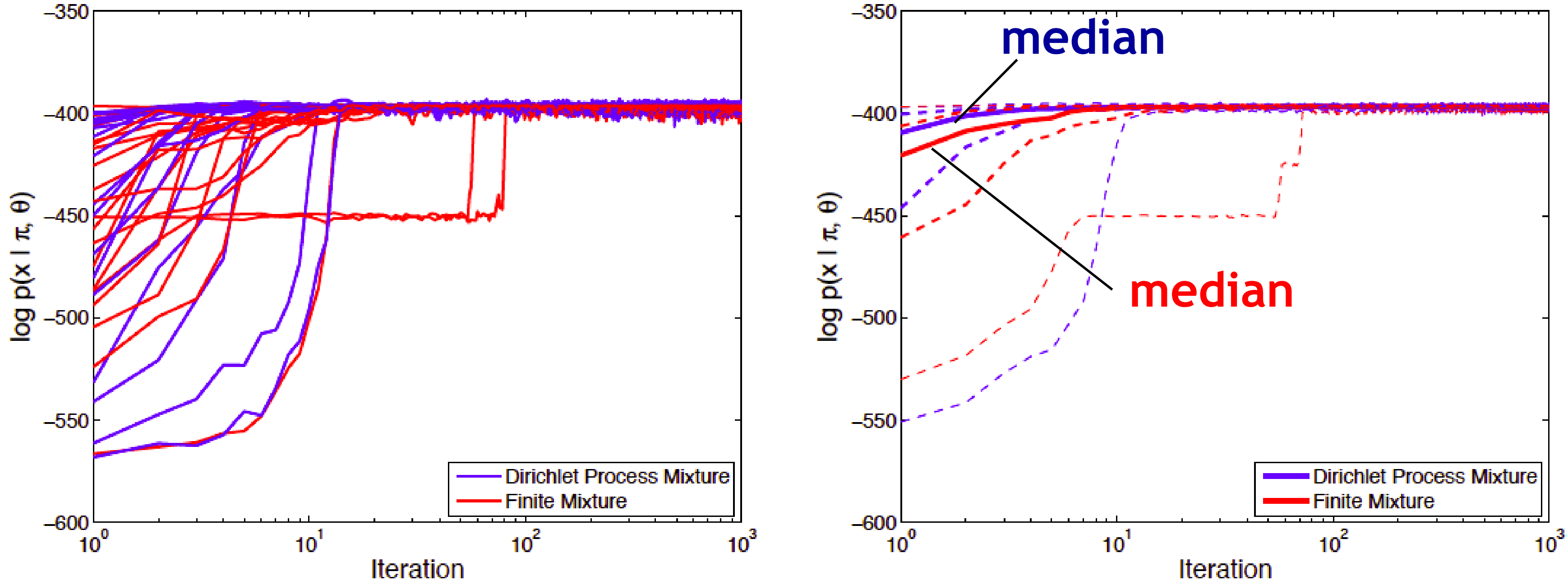
Collapsed DP Sampler: 50 Iterations



log p(x | π, θ) = –397.67

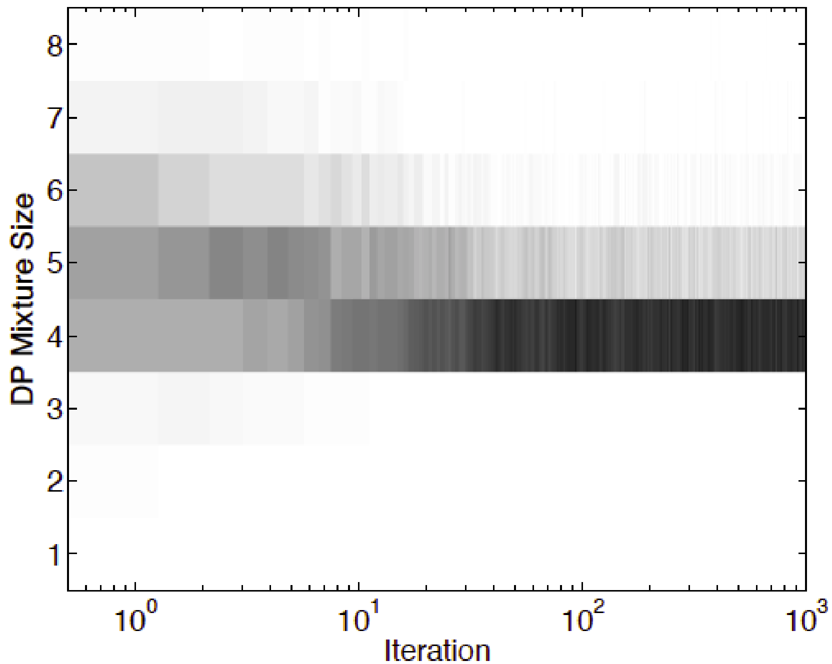
log p(x | π, θ) = –396.71

DPMMs vs. Finite Mixture Samplers

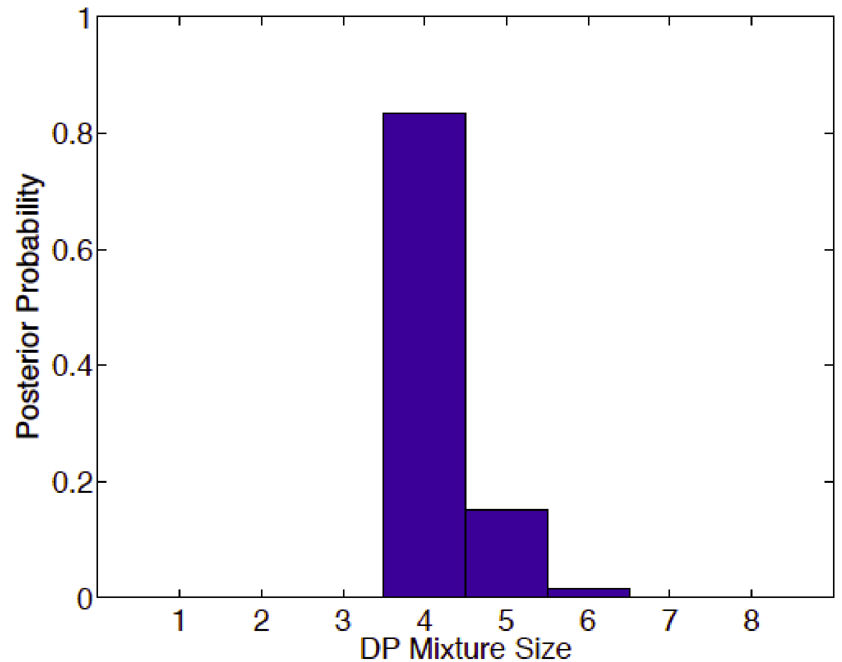


- **Observations**
  - **Despite having to search over mixtures of varying order, the DP sampler typically converges faster.**
  - **Avoids local optima by creating redundant clusters at beginning.**

DP Posterior Number of Clusters



Number of mixture components with at least 2% of the probability mass at each iteration



Average across the final 900 iterations

Image source: Erik Sudderth

Summary: Nonparametric Bayesian Clustering

- **DPMMs for Clustering**
  - First specify the likelihood. This is application dependent.
  - Next, specify a prior on all parameters - the Dirichlet Process!
  - Exact posterior inference is intractable. But we can use a Gibbs sampler for approximate inference. This is based on the CRP representation.

DPMM Software Packages

- **Matlab packages for CRP mixture models**

Algorithm	Author	Link
MCMC	J. Eisenstein	http://people.csail.mit.edu/jacobe/software.html
Variational	K. Kurihara	http://sites.google.com/site/kenichikurihara/academic-software

Topics of This Lecture

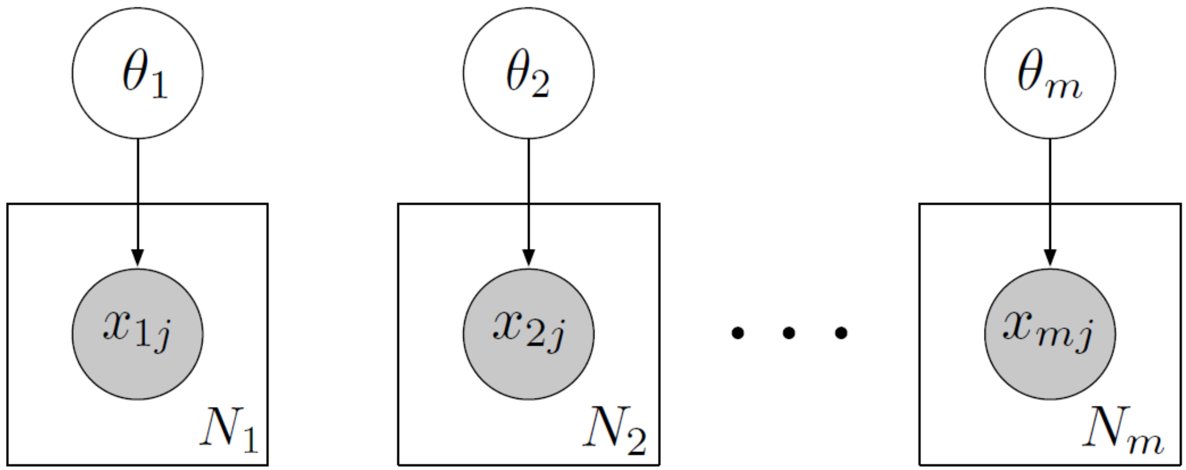
- Applying DPs
  - Recap: DPs
  - Efficient Gibbs sampling
- Hierarchical Dirichlet Processes
  - Definition
  - Properties
  - Chinese Restaurant Franchise
  - Gibbs sampling for HDPs
- Applications
  - Topic modeling

Hierarchical Bayesian Models

- Original Bayesian idea
  - View parameters as random variables - place a prior on them.
- Problem
  - Often the priors themselves need parameters (hyperparameters)
- Solution
  - Place a prior on these parameters!

Multiple Learning Problems

- We often face multiple, related learning problems
  - E.g., multiple related Gaussian means: $x_{ij} \sim \mathcal{N}(\theta_i, \sigma_i^2)$



  - Maximum likelihood: $\hat{\theta}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} x_{ij}$
  - ML often does not work very well…
  - Want to "share statistical strength" (i.e., smooth)

Hierarchical Bayesian Approach

- **Bayesian solution**
  - Treat the parameters θ_i as random variables sampled from an underlying prior θ_0.

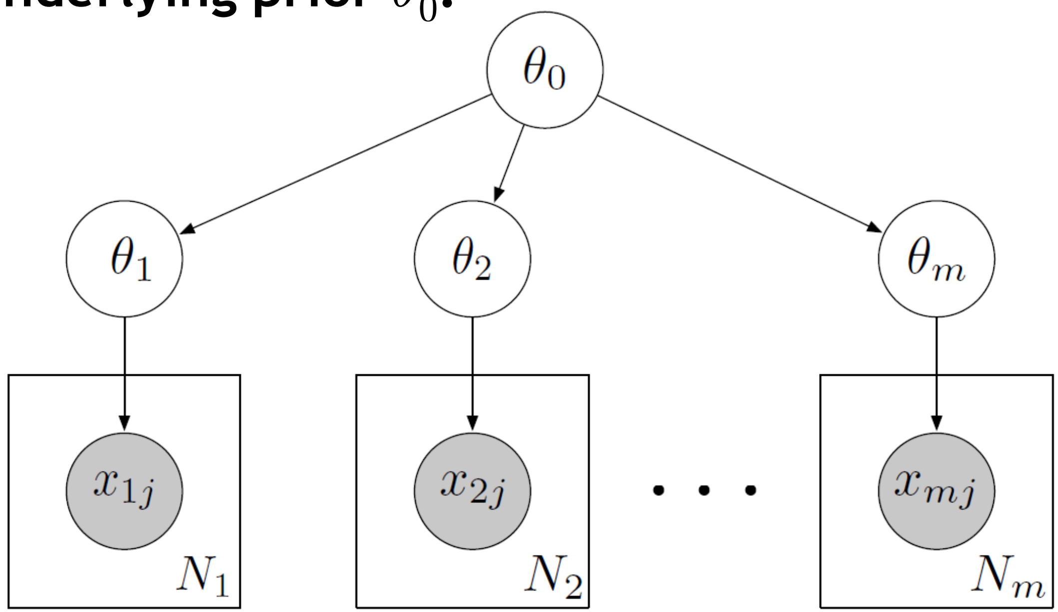
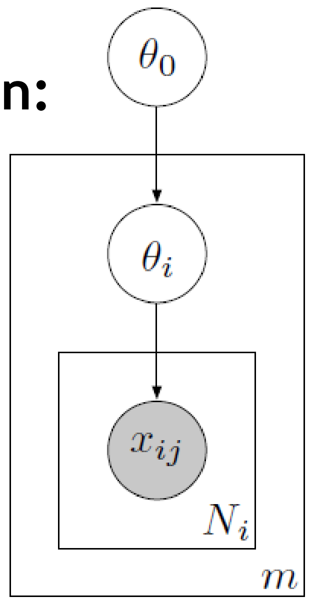


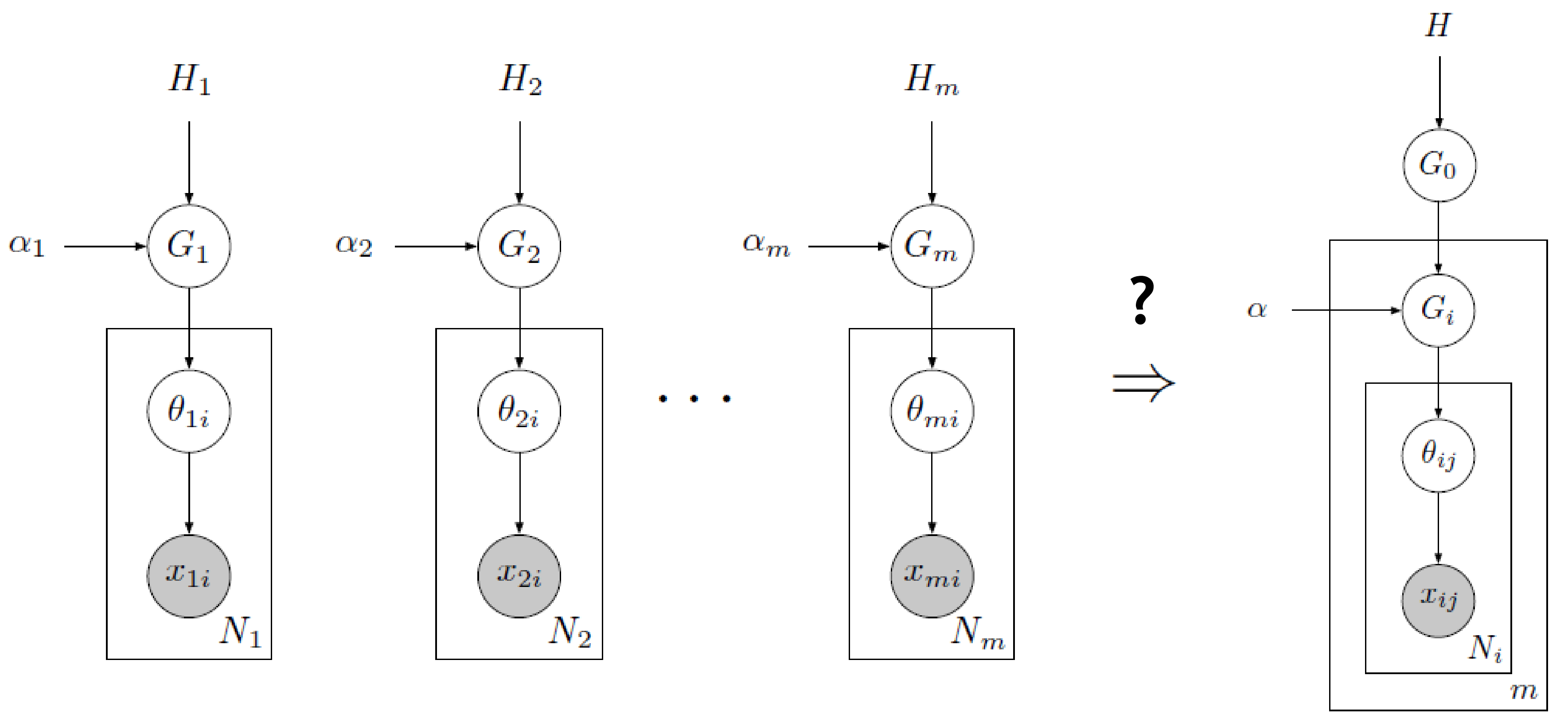
Plate notation:



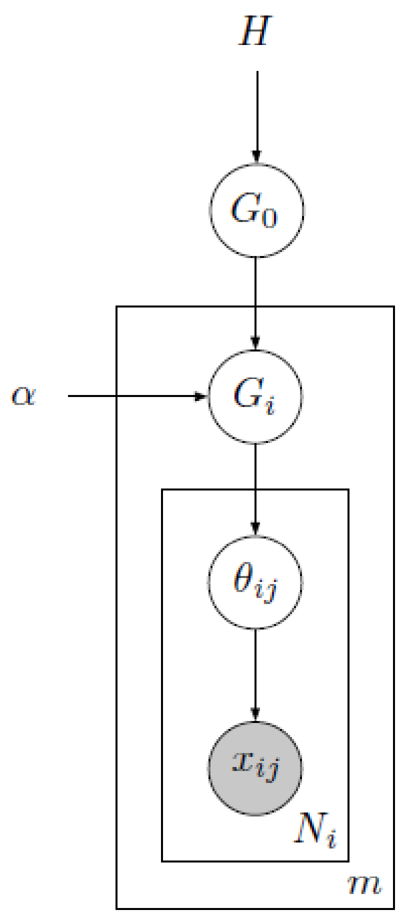
- **Bayesian inference yields shrinkage**
  - Posterior mean for each θ_k combines data from all of the groups, without simply lumping the data into one group.

Multiple Clustering Problems

- What to do if we have DPs for multiple related datasets?



Attempt 1



- **What kind of distribution do we use for G_0? What for H?**
- **Suppose θ_{ij} are mean parameters for a Gaussian where**

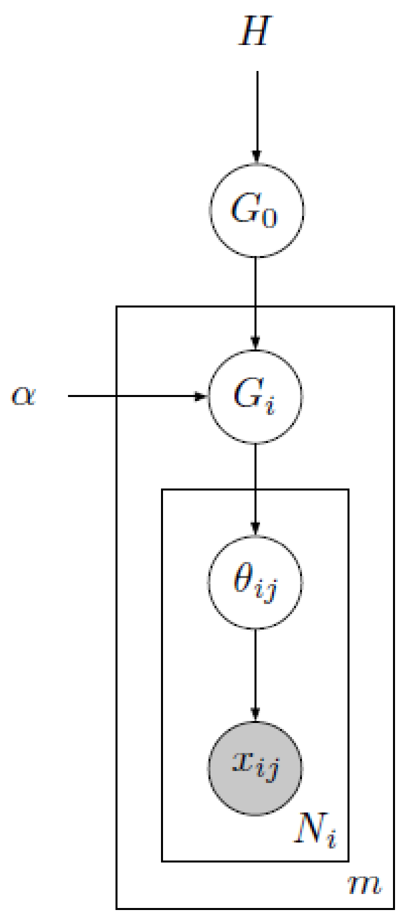
$$G_i \sim \mathrm{DP}(\alpha, G_0)$$

and G_0 is a Gaussian with unknown mean?

$$G_0 = \mathcal{N}(\theta_0, \sigma_0^2)$$

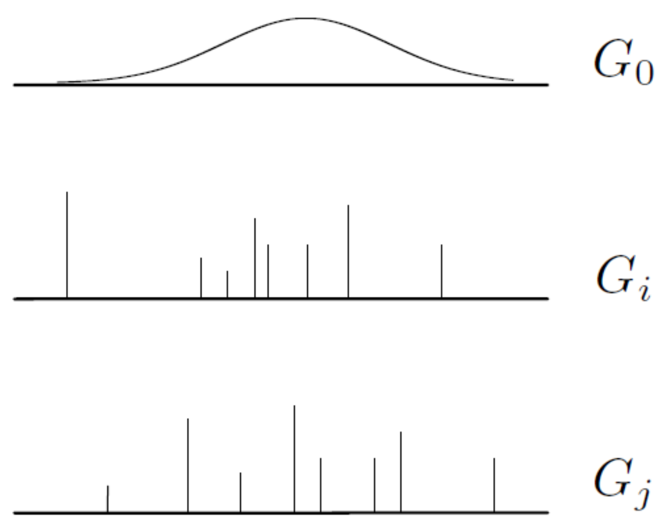
- **This does NOT work! *Why?***

Attempt 1



- Problem: if G_0 is continuous, then with probability ONE, G_i and G_j will share ZERO atoms.

$\Rightarrow$ This means NO clustering!



Hierarchical Dirichlet Processes

- We need to have the base measure G_0 be discrete
  - But also need it to be flexible and random.

- Solution:
  - Let G_0 itself be distributed according to a DP:

$$G_0|\gamma, H \sim \mathrm{DP}(\gamma, H)$$

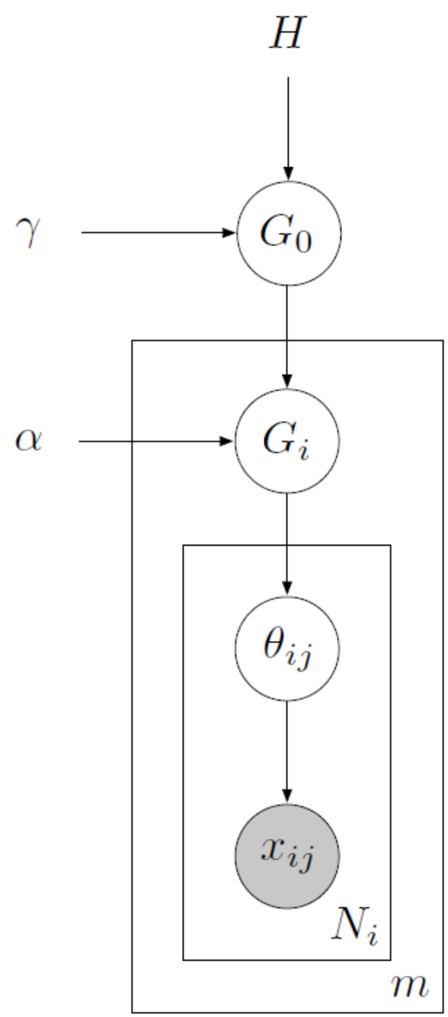
  - Then

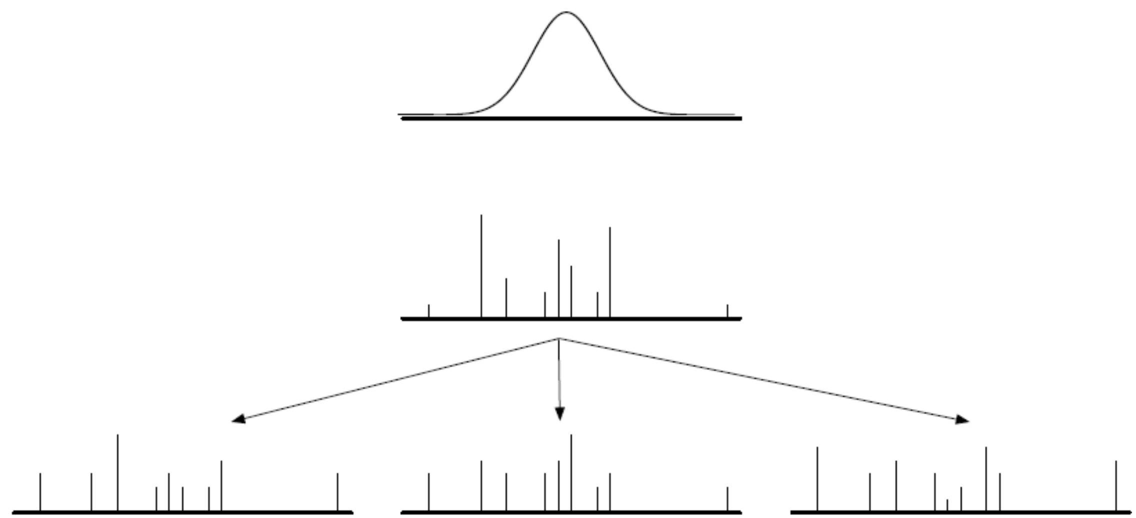
$$G_j|\alpha, G_0 \sim \mathrm{DP}(\alpha_0, G_0)$$

has at its base measure a (random) atomic distribution.

$\Rightarrow$ Samples of G_j will resample from those atoms.

Hierarchical Dirichlet Processes [Teh *et al.*, 2006]





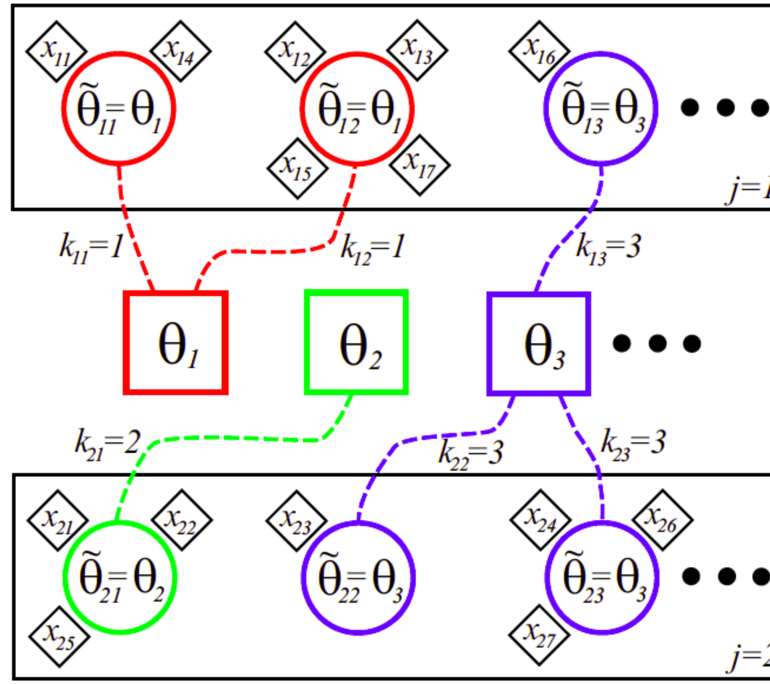
$$
\begin{aligned}
G_0 &\sim \mathrm{DP}(\gamma, H) \\
G_i &\sim \mathrm{DP}(\alpha, G_0) \\
\theta_{ij} | G_i &\sim G_i \\
x_{ij} | \theta_{ij} &\sim p(x_{ij} | \theta_{ij})
\end{aligned}
$$

Comparison

- Dirichlet Process
  - Useful in models for which a component of the model is a discrete random variable of unknown cardinality.
- Hierarchical Dirichlet Processes [Teh et al., 2006]
  - Useful in problems in which there are multiple groups of data, where the model for each group of data incorporates a discrete variable of unknown cardinality, and where we wish to tie these variables across groups.
- Similar representations for HDP to derive its properties
  - Stick-Breaking construction
  - Chinese Restaurant Franchise

Chinese Restaurant Franchise (CRF)

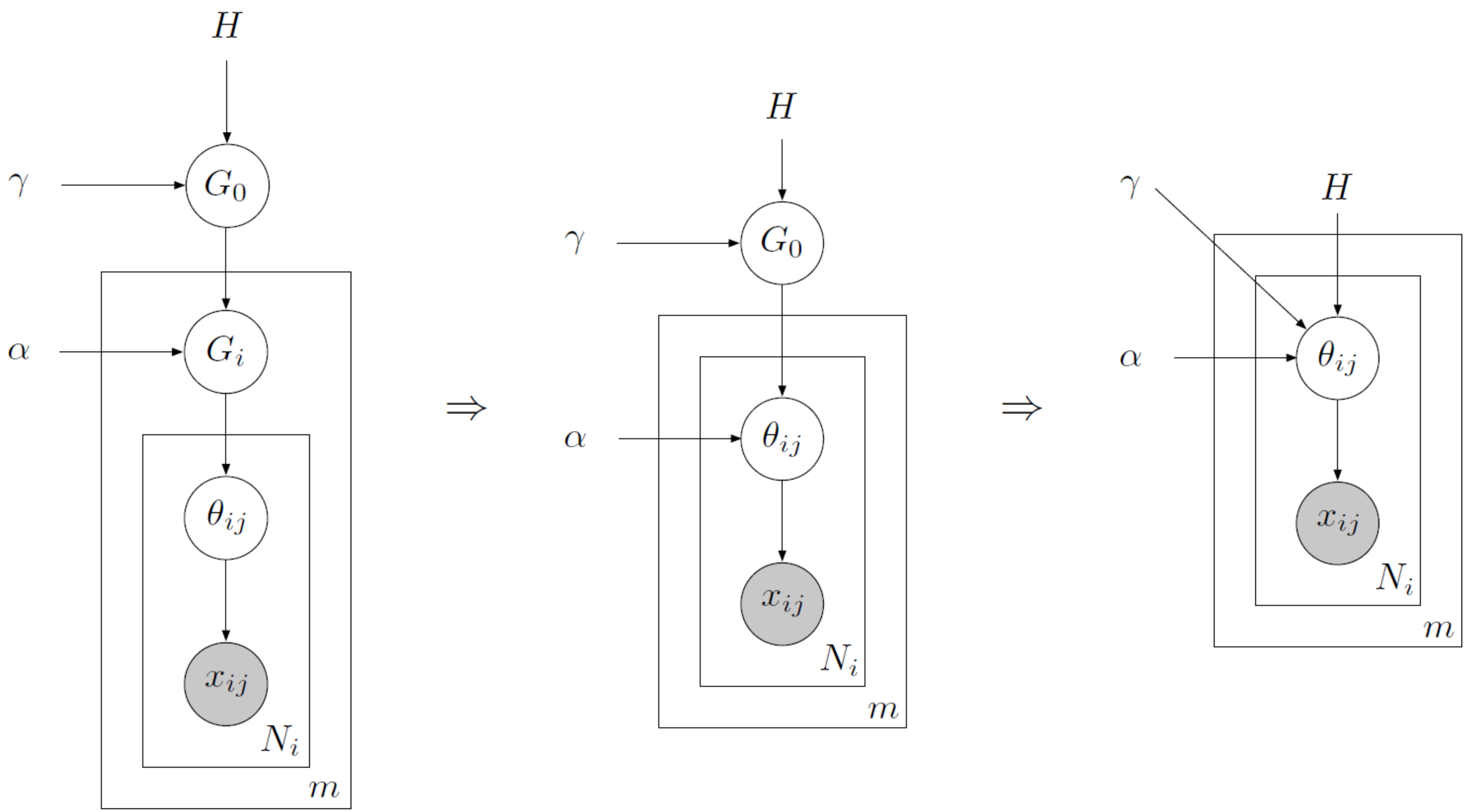
- Chain of Chinese restaurants
  - Each restaurant has an unbounded number of tables.
  - There is a global menu with an unbounded number of dishes.
  - The first customer at a table selects the dish for that table from the global menu.



- Reinforcement effects
  - Customers prefer to sit at tables with many other customers, and prefer dishes that are chosen by many other customers.
  - Dishes are chosen with probability proportional to the number of tables (franchise-wide) that have previously served that dish.

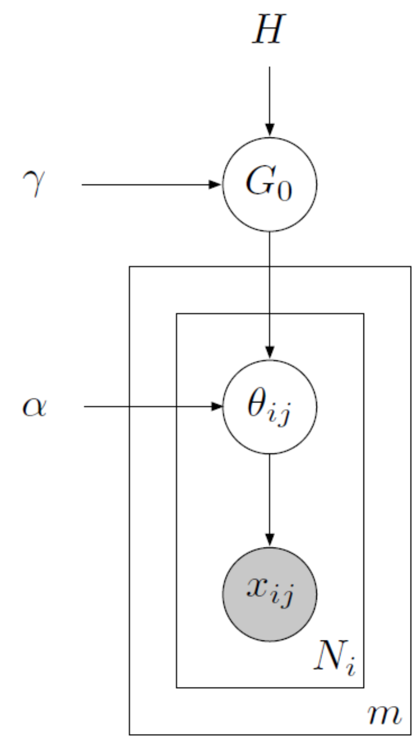
Chinese Restaurant Franchise (CRF)

- **Examine marginal properties of HDP**
  - **First integrate out G_i, then G_0.**



Chinese Restaurant Franchise (CRF)

- **Step 1: Integrate out G_i:**
  - **Variable definitions**
    - θ_{ij} **: RV for customer i in restaurant j.**
    - $\theta_{jt}{}^{*}$ **: RV for table t in restaurant j.**
    - $\theta_{k}{}^{**}$ **: RV for dish k.**
    - m_{jk}**: number of tables in rest. j serving dish k.**
    - n_{jtk}**: number of customers in rest. j sitting at table t and being served dish k.**
    - **We denote marginal counts by dots, e.g.** $m_{j\cdot} = \sum_{k=1}^{K} m_{jk}$



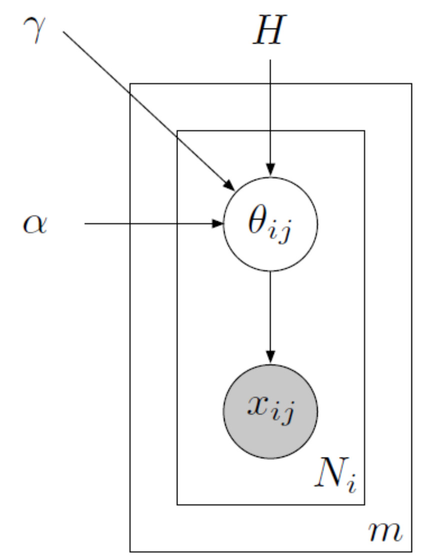
  - **Integration yields a set of conditional distributions described by a Polya urn scheme**

$$\theta_{ij} | \theta_{1j}, \ldots, \theta_{i-1,j}, \alpha, G_0 \;\sim\; \sum_{t=1}^{m_{j\cdot}} \frac{n_{jt\cdot}}{\alpha + n_{j\cdot\cdot}} \delta_{\theta^*_{jt}} + \frac{\alpha}{\alpha + n_{j\cdot\cdot}} G_0$$

Image source: Kurt Miller

Chinese Restaurant Franchise (CRF)

- **Step 2: Integrate out G_0:**
  - **Variable definitions**
    - θ_{ij} : RV for customer i in restaurant j.
    - θ_{jt}^* : RV for table t in restaurant j.
    - θ_k^{**} : RV for dish k.
    - m_{jk}: number of tables in rest. j serving dish k.
    - n_{jtk}: number of customers in rest. j sitting at table t and being served dish k.
    - We denote marginal counts by dots, e.g. $m_{j\cdot} = \sum_{k=1}^{K} m_{jk}$



  - **Again, we get a Polya urn scheme**

$$\theta_{jt}^* | \theta_{11}^*, \ldots, \theta_{1,m_{1,\cdot}}^*, \ldots, \theta_{j,t-1}^*, \gamma, H \sim \sum_{k=1}^{K} \frac{m_{\cdot k}}{\gamma + m_{\cdot\cdot}} \delta_{\theta_k^{**}} + \frac{\gamma}{\gamma + m_{\cdot\cdot}} H$$

Image source: Kurt Miller

Inference for HDP: CRF Sampler

- Using the CRF representation of the HDP
  - Customer i in restaurant j is associated with i.i.d draw from G_i and sits at table t_{ij}.
  - Table t in restaurant j is associated with i.i.d draw from G_0 and serves dish k_{jt}.
  - Dish k is associated with i.i.d draw from H.
- Gibbs sampling approach
  - Iteratively sample the table and dish assignment variables, conditioned on the state of all other variables.
  - The parameters θ_{ij} are integrated out analytically (assuming conjugacy).
  - To resample, make use of exchangeability.

  $\Rightarrow$ Imagine each customer i being the last to enter restaurant j.

Inference for HDP: CRF Sampler

- **Procedure**
  1. **Resample t_{ij} according to the following distribution**

$$
\begin{cases}
t_{ij} = t & \text{with prob.} \quad \propto \frac{n_{jt\cdot}^{\neg ij}}{n_{j\cdot\cdot}^{\neg ij} + \alpha} f_{k_{jt}}(\{x_{ij}\}) \\
t_{ij} = t^{\text{new}}, k_{jt^{\text{new}}} = k & \text{with prob.} \quad \propto \frac{\alpha}{n_{j\cdot\cdot}^{\neg ij} + \alpha} \propto \frac{m_{\cdot k}^{\neg ij}}{m_{\cdot\cdot}^{\neg ij} + \gamma} f_k(\{x_{ij}\}) \\
t_{ij} = t^{\text{new}}, k_{jt^{\text{new}}} = k^{\text{new}} & \text{with prob.} \quad \propto \frac{\alpha}{n_{j\cdot\cdot}^{\neg ij} + \alpha} \propto \frac{\gamma}{m_{\cdot\cdot}^{\neg ij} + \gamma} f_{k^{\text{new}}}(\{x_{ij}\})
\end{cases}
$$

where $\neg ij$ denotes counts in which customer i in restaurant j is removed from the CRF. (If this empties a table, we also remove the table from the CRF, along with the dish on it.)

  - **The terms $f_k(\{x_{ij}\})$ are defined as follows**

$$
f_k(\{x_{ij}\}_{ij \in D}) = \frac{\int h(\theta) \prod_{i'j' \in D_k \cup D} f_\theta(x_{i'j'}) \mathrm{d}\theta}{\int h(\theta) \prod_{i'j' \in D_k \setminus D} f_\theta(x_{i'j'}) \mathrm{d}\theta}
$$

where D_K denotes the set of indices associated with dish k.

Inference for HDP: CRF Sampler

- Procedure (cont'd)
  2. Resample k_{jt} (Gibbs update for the dish)

$$k_{jt} = \begin{cases} k & \text{with prob.} \quad \propto \frac{m_{\cdot k}^{\neg jt}}{m_{\cdot\cdot}^{\neg jt} + \gamma} f_k(\{x_{ij} : t_{ij} = t\}) \\ k^{\text{new}} & \text{with prob.} \quad \propto \frac{\gamma}{m_{\cdot\cdot}^{\neg jt} + \gamma} f_{k^{\text{new}}}(\{x_{ij} : t_{ij} = t\}) \end{cases}$$

- Remarks
  - Computational cost of Gibbs updates is dominated by computation of the marginal conditional probabilities $f_k(\cdot)$.
  - Still, the number of possible events that can occur at one Gibbs step is one plus the total number of tables and dishes in all restaurants that are ancestors of j.
  - This number can get quite large in deep or wide hierarchies…

Topics of This Lecture

- Applying DPs
  - Recap: DPs
  - Efficient Gibbs sampling
- Hierarchical Dirichlet Processes
  - Definition
  - Properties
  - Chinese Restaurant Franchise
  - Gibbs sampling for HDPs
- Applications
  - Topic modeling

Applications

- **Example: Document topic modelling**
  - Topic: probability distribution over a set of words
  - Model each document as a probability distribution over topics.

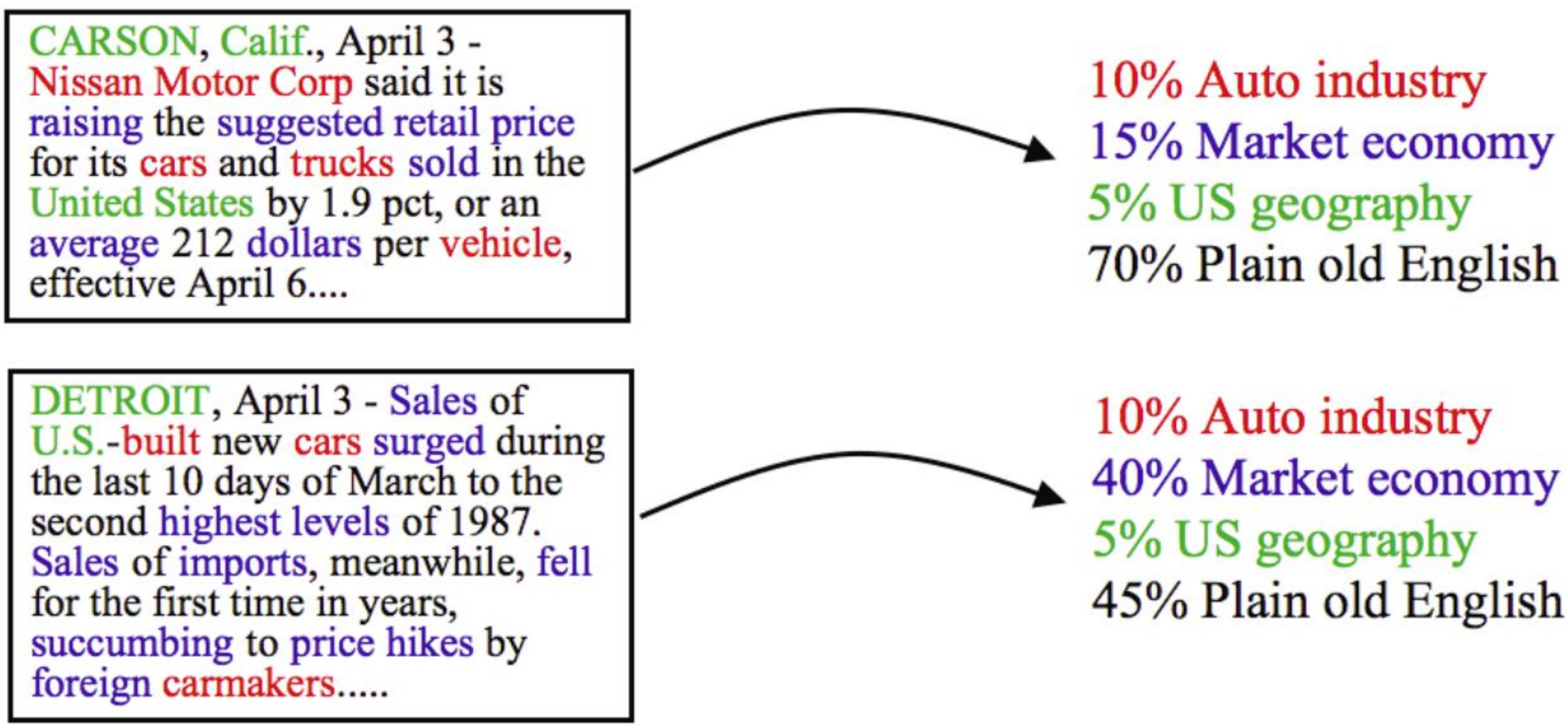
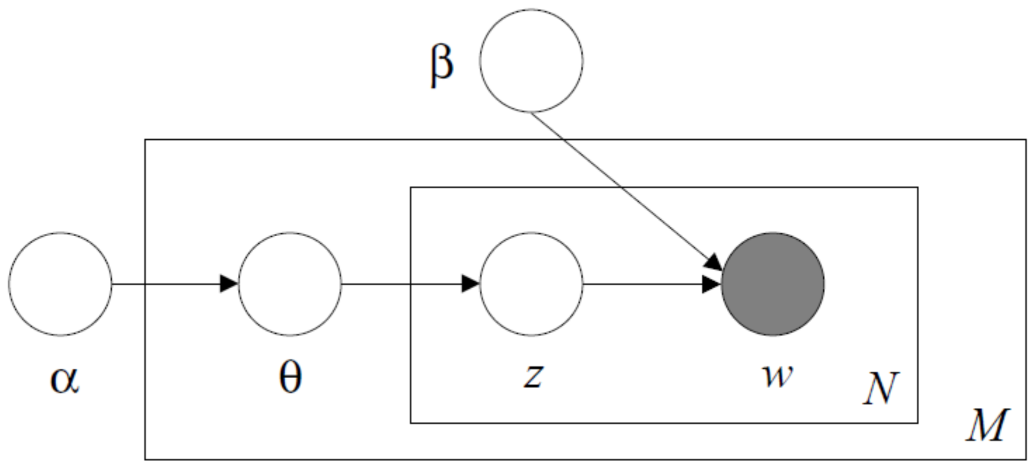


Image source: Yee Whye Teh

Applications

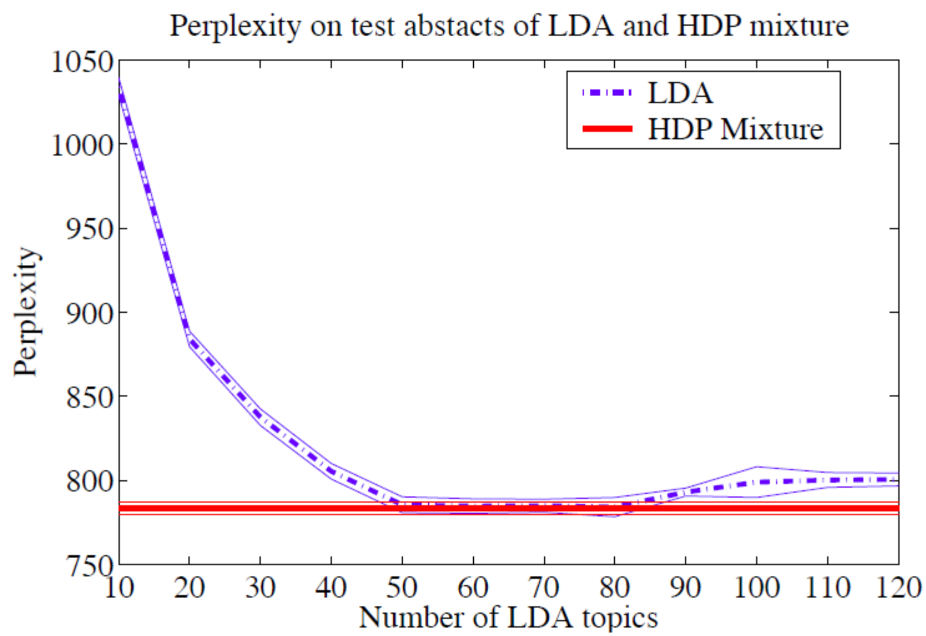
- **Latent Dirichlet Allocation** [Blei *et al.*, 2003]
  - Popular topic modelling approach with fixed number of topics k

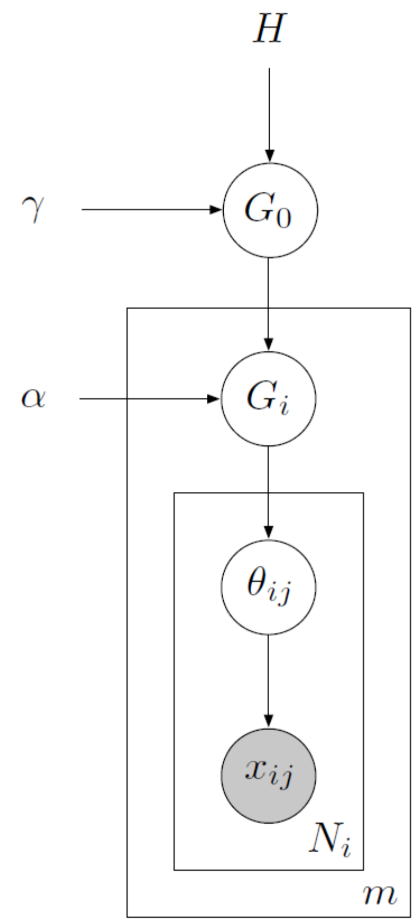


  - Random variables
    - A word is represented as a multinomial random variable w
    - A topic is represented as a multinomial random variable z
    - A document is represented as a Dirichlet random variable θ

Applications

- **HDPs can be used to define a BNP version of LDA**
  - **Number of topics is open-ended**
  - **Multiple infinite mixture models, linked via shared topic distribution.**





$\Rightarrow$ **HDP-LDA avoids the need for model selection.**

Applications

- There are many other generalizations I didn’t talk about
  - Dependent DPs
  - Nested DPs
  - Pitman-Yor processes
  - Infinite HMMs
  - ...

- And some that I will talk about in Lectures 15/16...
  - Infinite Latent Factor Models
  - Beta Processes
  - Indian Buffet Process
  - Hierarchical Beta Process

References and Further Reading

- Unfortunately, there are currently no good introductory textbooks on the Dirichlet Process. We will therefore post a number of tutorial papers on their different aspects.

  - One of the best available general introductions
    - E.B. Sudderth, "Graphical Models for Visual Object Recognition and Tracking", PhD thesis, Chapter 2, Section 2.5, 2006.

  - A tutorial on Hierarchical DPs
    - Y.W. Teh, M.I. Jordan, Hierarchical Bayesian Nonparametric Models with Applications. Bayesian Nonparametrics, Cambridge Univ. Press, 2010.

  - Good overview of MCMC methods for DPMMs
    - R. Neal, Markov Chain Sampling Methods for Dirichlet Process Mixture Models. Journal of Computational and Graphical Statistics, Vol. 9(2), p. 249-265, 2000.