Dirichlet Processes: Tutorial and Practical Course
(updated)

Yee Whye Teh

Gatsby Computational Neuroscience Unit
University College London

August 2007 / MLSS

Dirichlet Processes

- Dirichlet processes (DPs) are a class of Bayesian nonparametric models.
- Dirichlet processes are used for:
  - Density estimation.
  - Semiparametric modelling.
  - Sidestepping model selection/averaging.
- I will give a tutorial on DPs, followed by a practical course on implementing DP mixture models in MATLAB.
- Prerequisites: understanding of the Bayesian paradigm (graphical models, mixture models, exponential families, Gaussian processes)—you should know these from previous courses.
- Other tutorials on DPs:
  - Zoubin Gharamani, UAI 2005.
  - Michael Jordan, NIPS 2005.
  - Volker Tresp, ICML nonparametric Bayes workshop 2006.
  - Workshop on Bayesian Nonparametric Regression, Cambridge, July 2007.

Dirichlet Processes

- Dirichlet processes (DPs) are a class of Bayesian nonparametric models.
- Dirichlet processes are used for:
  - Density estimation.
  - Semiparametric modelling.
  - Sidestepping model selection/averaging.
- I will give a tutorial on DPs, followed by a practical course on implementing DP mixture models in MATLAB.
- Prerequisites: understanding of the Bayesian paradigm (graphical models, mixture models, exponential families, Gaussian processes)—you should know these from previous courses.
- Other tutorials on DPs:
  - Zoubin Gharamani, UAI 2005.
  - Michael Jordan, NIPS 2005.
  - Volker Tresp, ICML nonparametric Bayes workshop 2006.
  - Workshop on Bayesian Nonparametric Regression, Cambridge, July 2007.

Dirichlet Processes

- Dirichlet processes (DPs) are a class of Bayesian nonparametric models.
- Dirichlet processes are used for:
  - Density estimation.
  - Semiparametric modelling.
  - Sidestepping model selection/averaging.
- I will give a tutorial on DPs, followed by a practical course on implementing DP mixture models in MATLAB.
- Prerequisites: understanding of the Bayesian paradigm (graphical models, mixture models, exponential families, Gaussian processes)—you should know these from previous courses.
- Other tutorials on DPs:
  - Zoubin Gharamani, UAI 2005.
  - Michael Jordan, NIPS 2005.
  - Volker Tresp, ICML nonparametric Bayes workshop 2006.
  - Workshop on Bayesian Nonparametric Regression, Cambridge, July 2007.

Dirichlet Processes

- Dirichlet processes (DPs) are a class of Bayesian nonparametric models.
- Dirichlet processes are used for:
  - Density estimation.
  - Semiparametric modelling.
  - Sidestepping model selection/averaging.
- I will give a tutorial on DPs, followed by a practical course on implementing DP mixture models in MATLAB.
- Prerequisites: understanding of the Bayesian paradigm (graphical models, mixture models, exponential families, Gaussian processes)—you should know these from previous courses.
- Other tutorials on DPs:
  - Zoubin Gharamani, UAI 2005.
  - Michael Jordan, NIPS 2005.
  - Volker Tresp, ICML nonparametric Bayes workshop 2006.
  - Workshop on Bayesian Nonparametric Regression, Cambridge, July 2007.

Outline

Outline

Function Estimation

- Parametric function estimation (e.g. regression, classification)

  Data: $\mathbf{x} = \{x_1, x_2, \ldots\}, \mathbf{y} = \{y_1, y_2, \ldots\}$

  Model: $y_i = f(x_i|w) + \mathcal{N}(0, \sigma^2)$
- Prior over parameters

$$p(w)$$

- Posterior over parameters

$$p(w|\mathbf{x}, \mathbf{y}) = \frac{p(w)p(\mathbf{y}|\mathbf{x}, w)}{p(\mathbf{y}|\mathbf{x})}$$

- Prediction with posteriors

$$p(y_\star|x_\star, \mathbf{x}, \mathbf{y}) = \int p(y_\star|x_\star, w)p(w|\mathbf{x}, \mathbf{y})\, dw$$

Function Estimation

- Bayesian nonparametric function estimation with Gaussian processes
  Data: $\mathbf{x} = \{x_1, x_2, \ldots\}, \mathbf{y} = \{y_1, y_2, \ldots\}$
  Model: $y_i = f(x_i) + \mathcal{N}(0, \sigma^2)$
- Prior over functions

$$f \sim \mathrm{GP}(\mu, \Sigma)$$

- Posterior over functions

$$p(f|\mathbf{x}, \mathbf{y}) = \frac{p(f)p(\mathbf{y}|\mathbf{x}, f)}{p(\mathbf{y}|\mathbf{x})}$$

- Prediction with posteriors

$$p(y_\star|x_\star, \mathbf{x}, \mathbf{y}) = \int p(y_\star|x_\star, f)p(f|\mathbf{x}, \mathbf{y})\, df$$

Function Estimation

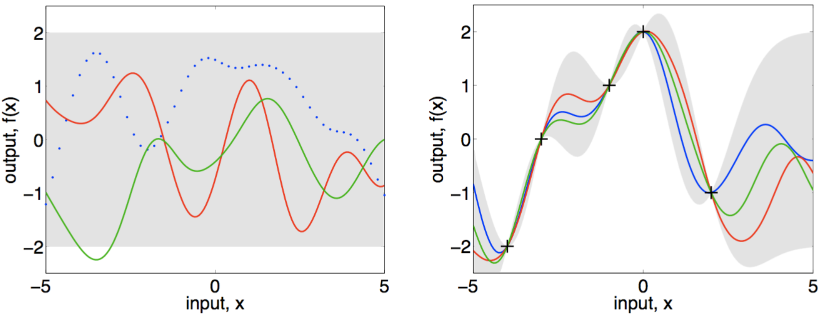


Figure from Carl's lecture.

Density Estimation

- Parametric density estimation (e.g. Gaussian, mixture models)

  Data: $\mathbf{x} = \{x_1, x_2, \ldots\}$

  Model: $x_i | w \sim F(\cdot | w)$
- Prior over parameters

$$p(w)$$

- Posterior over parameters

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

- Prediction with posteriors

$$p(x_\star|\mathbf{x}) = \int p(x_\star|w)p(w|\mathbf{x})\,dw$$

Density Estimation

- Bayesian nonparametric density estimation with Dirichlet processes
  Data: $\mathbf{x} = \{x_1, x_2, \ldots\}$
  Model: $x_i \sim F$
- Prior over distributions

$$F \sim \mathrm{DP}(\alpha, H)$$

- Posterior over distributions

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

- Prediction with posteriors

$$p(x_\star|\mathbf{x}) = \int p(x_\star|F)p(F|\mathbf{x})\,dF = \int F'(x_\star)p(F|\mathbf{x})\,dF$$

- *Not quite correct; see later.*

Density Estimation

- Bayesian nonparametric density estimation with Dirichlet processes
  Data: $\mathbf{x} = \{x_1, x_2, \ldots\}$
  Model: $x_i \sim F$
- Prior over distributions

$$F \sim \mathrm{DP}(\alpha, H)$$

- Posterior over distributions

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

- Prediction with posteriors

$$p(x_\star|\mathbf{x}) = \int p(x_\star|F)p(F|\mathbf{x})\, dF = \int F'(x_\star)p(F|\mathbf{x})\, dF$$

- *Not quite correct; see later.*

Density Estimation

- Bayesian nonparametric density estimation with Dirichlet processes
  Data: $\mathbf{x} = \{x_1, x_2, \ldots\}$
  Model: $x_i \sim F$
- Prior over distributions

$$F \sim \mathrm{DP}(\alpha, H)$$

- Posterior over distributions

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

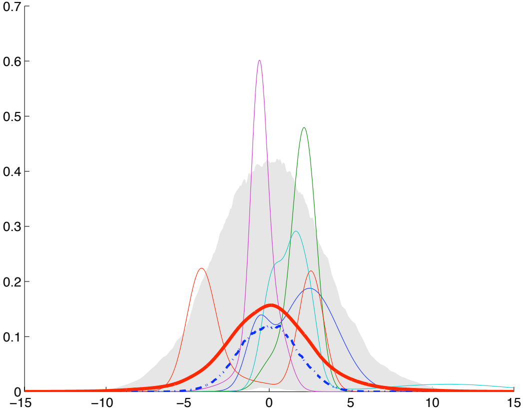
- Prediction with posteriors

$$p(x_\star|\mathbf{x}) = \int p(x_\star|F)p(F|\mathbf{x})\, dF = \int F'(x_\star)p(F|\mathbf{x})\, dF$$

- *Not quite correct; see later.*

Density Estimation

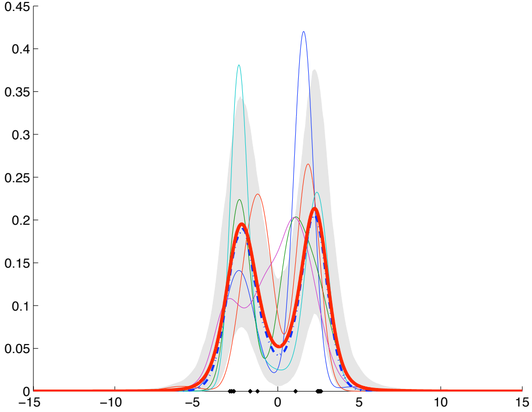
Prior:



Red: mean density. Blue: median density. Grey: 5-95 quantile.
Others: draws.

Density Estimation

Posterior:



Red: mean density. Blue: median density. Grey: 5-95 quantile.
Black: data. Others: draws.

Semiparametric Modelling

- Linear regression model for inferring effectiveness of new medical treatments.

$$y_{ij} = \beta^\top x_{ij} + b_i^\top z_{ij} + \epsilon_{ij}$$

  y_{ij} is outcome of jth trial on ith subject.

  x_{ij}, z_{ij} are predictors (treatment, dosage, age, health...).

  β are fixed-effects coefficients.

  b_i are random-effects subject-specific coefficients.

  ϵ_{ij} are noise terms.

- Care about inferring β. If x_{ij} is treatment, we want to determine $p(\beta > 0|\mathbf{x}, \mathbf{y})$.

Semiparametric Modelling

$$y_{ij} = \beta^\top x_{ij} + b_i^\top z_{ij} + \epsilon_{ij}$$

- Usually we assume Gaussian noise $\epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$. Is this a sensible prior? Over-dispersion, skewness,...
- May be better to model noise nonparametrically,

$$\begin{aligned} \epsilon_{ij} &\sim F \\ F &\sim \mathrm{DP} \end{aligned}$$

- Also possible to model subject-specific random effects nonparametrically,

$$\begin{aligned} b_i &\sim G \\ G &\sim \mathrm{DP} \end{aligned}$$

Model Selection/Averaging

- Data: $\mathbf{x} = \{x_1, x_2, \dots\}$
  Models: $p(\theta_k|M_k)$, $p(\mathbf{x}|\theta_k, M_k)$
- Marginal likelihood

$$p(\mathbf{x}|M_k) = \int p(\mathbf{x}|\theta_k, M_k) p(\theta_k|M_k)\, d\theta_k$$

- Model selection

$$M = \operatorname*{argmax}_{M_k} p(\mathbf{x}|M_k)$$

- Model averaging

$$p(x_\star|\mathbf{x}) = \sum_{M_k} p(x_\star|M_k) p(M_k|\mathbf{x}) = \sum_{M_k} p(x_\star|M_k) \frac{p(\mathbf{x}|M_k)p(M_k)}{p(\mathbf{x})}$$

- *But: is this computationally feasible?*

Model Selection/Averaging

- Data: $\mathbf{x} = \{x_1, x_2, \ldots\}$
  Models: $p(\theta_k|M_k)$, $p(\mathbf{x}|\theta_k, M_k)$
- Marginal likelihood

$$p(\mathbf{x}|M_k) = \int p(\mathbf{x}|\theta_k, M_k)p(\theta_k|M_k)\,d\theta_k$$

- Model selection

$$M = \operatorname*{argmax}_{M_k} p(\mathbf{x}|M_k)$$

- Model averaging

$$p(x_\star|\mathbf{x}) = \sum_{M_k} p(x_\star|M_k)p(M_k|\mathbf{x}) = \sum_{M_k} p(x_\star|M_k)\frac{p(\mathbf{x}|M_k)p(M_k)}{p(\mathbf{x})}$$

- *But: is this computationally feasible?*

Model Selection/Averaging

- Data: $\mathbf{x} = \{x_1, x_2, \ldots\}$
  Models: $p(\theta_k|M_k)$, $p(\mathbf{x}|\theta_k, M_k)$
- Marginal likelihood

$$p(\mathbf{x}|M_k) = \int p(\mathbf{x}|\theta_k, M_k)p(\theta_k|M_k)\, d\theta_k$$

- Model selection

$$M = \operatorname*{argmax}_{M_k} p(\mathbf{x}|M_k)$$

- Model averaging

$$p(x_\star|\mathbf{x}) = \sum_{M_k} p(x_\star|M_k)p(M_k|\mathbf{x}) = \sum_{M_k} p(x_\star|M_k)\frac{p(\mathbf{x}|M_k)p(M_k)}{p(\mathbf{x})}$$

- *But: is this computationally feasible?*

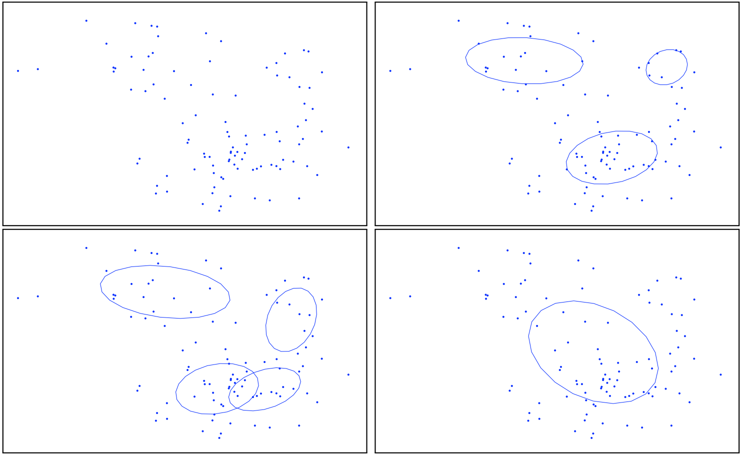
Model Selection/Averaging

- Marginal likelihood is usually extremely hard to compute.

$$p(\mathbf{x}|M_k) = \int p(\mathbf{x}|\theta_k, M_k)p(\theta_k|M_k)\, d\theta_k$$

- Model selection/averaging is to prevent underfitting and overfitting.
- But reasonable and proper Bayesian methods should not overfit [Rasmussen and Ghahramani 2001].
- Use a really large model M_∞ instead, and let the data speak for themselves.

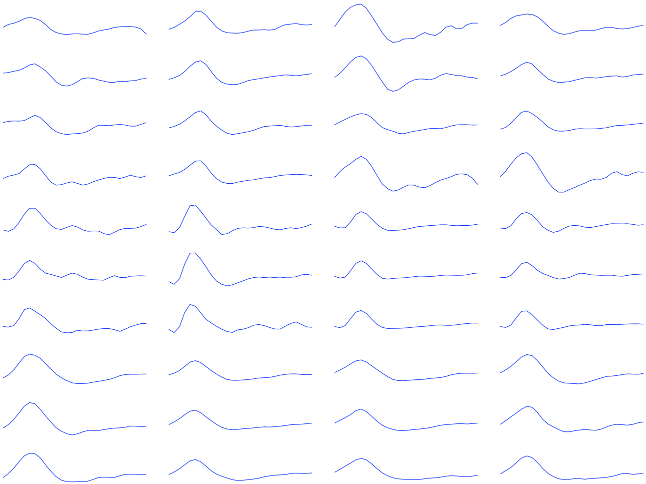
How many clusters are there?



Model Selection/Averaging

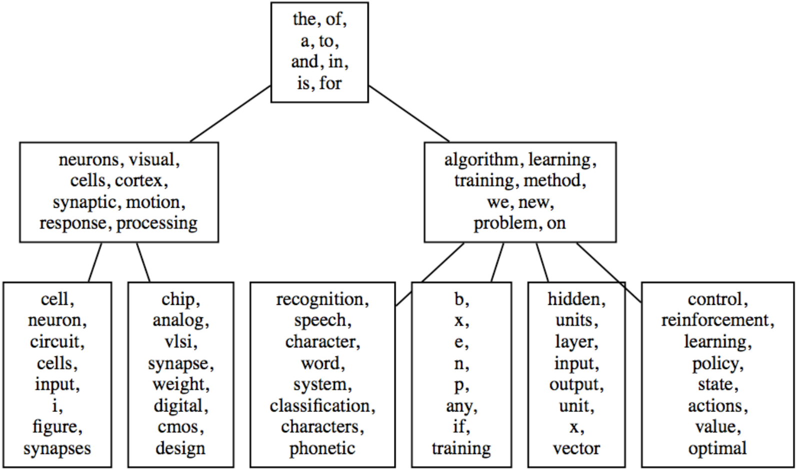
Spike Sorting

How many neurons are there?



[Görür 2007, Wood et al. 2006a]

How many topics are there?



[Blei et al. 2004, Teh et al. 2006]

How many grammar symbols are there?

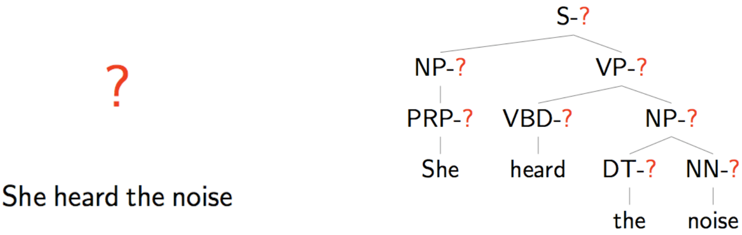


Figure from Liang. [Liang et al. 2007b, Finkel et al. 2007]

Model Selection/Averaging
Visual Scene Analysis

How many objects, parts, features?

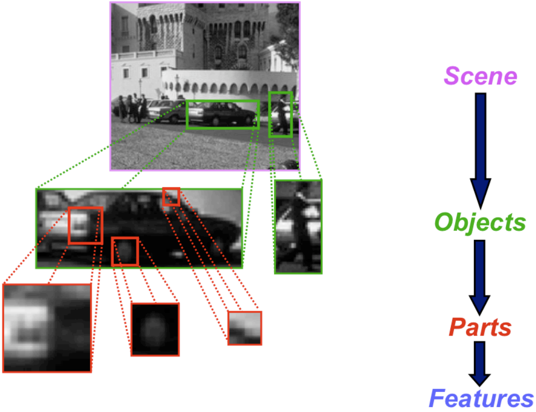


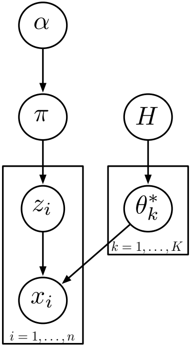
Figure from Sudderth. [Sudderth et al. 2007]

Outline

Finite Mixture Models

- A finite mixture model is defined as follows:

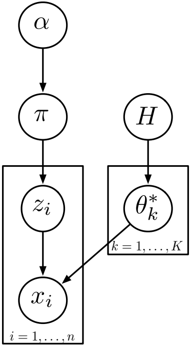
$$\begin{aligned} \theta_k^* &\sim H \\ \pi &\sim \text{Dirichlet}(\alpha/K, \dots, \alpha/K) \\ z_i|\pi &\sim \text{Discrete}(\pi) \\ x_i|\theta_{z_i}^* &\sim F(\cdot|\theta_{z_i}^*) \end{aligned}$$



- Model selection/averaging over:
  - Hyperparameters in H.
  - Dirichlet parameter α.
  - Number of components K.
- Determining K hardest.

Infinite Mixture Models

- Imagine that $K \gg 0$ is really large.
- If parameters θ_k^* and mixing proportions π integrated out, the number of latent variables left does not grow with K—no overfitting.
- At most n components will be associated with data, aka "active".
- Usually, the number of active components is much less than n.
- This gives an infinite mixture model.
- Demo: dpm_demo2d
- *Issue 1: can we take this limit $K \to \infty$?*
- *Issue 2: what is the corresponding limiting model?*



[Rasmussen 2000]

Gaussian Processes

What are they?

- A Gaussian process (GP) is a distribution over functions

$$f : \mathbb{X} \mapsto \mathbb{R}$$

- Denote $f \sim \text{GP}$ if f is a GP-distributed random function.
- For any finite set of input points $x_1, \dots, x_n$, we require $(f(x_1), \dots, f(x_n))$ to be a multivariate Gaussian.

Gaussian Processes

What are they?

- The GP is parametrized by its mean $m(x)$ and covariance $c(x, y)$ functions:

$$\begin{bmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_n) \end{bmatrix}, \begin{bmatrix} c(x_1, x_1) & \dots & c(x_1, x_n) \\ \vdots & \ddots & \vdots \\ c(x_n, x_1) & \dots & c(x_n, x_n) \end{bmatrix} \right)$$

- The above are finite dimensional marginal distributions of the GP.
- A salient property of these marginal distributions is that they are consistent: integrating out variables preserves the parametric form of the marginal distributions above.

- A sequence of input points $x_1, x_2, x_3, \ldots$ dense in $\mathbb{X}$.
- Draw

$$
\begin{aligned}
&f(x_1)\\
&f(x_2) \mid f(x_1)\\
&f(x_3) \mid f(x_1), f(x_2)\\
&\qquad\vdots
\end{aligned}
$$

- Each conditional distribution is Gaussian since $(f(x_1), \ldots, f(x_n))$ is Gaussian.
- Demo: GPgenerate

- A Dirichlet distribution is a distribution over the K-dimensional probability simplex:

$$\Delta_K = \left\{(\pi_1, \dots, \pi_K) \,:\, \pi_k \geq 0, \textstyle\sum_k \pi_k = 1\right\}$$

- We say $(\pi_1, \dots, \pi_K)$ is Dirichlet distributed,

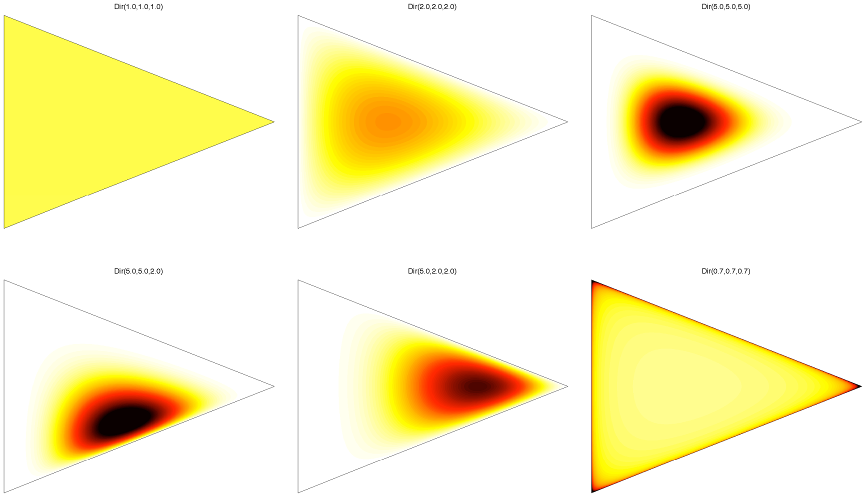
$$(\pi_1, \dots, \pi_K) \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_K)$$

  with parameters $(\alpha_1, \dots, \alpha_K)$, if

$$p(\pi_1, \dots, \pi_K) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{n} \pi_k^{\alpha_k - 1}$$

Dirichlet Processes

Examples of Dirichlet distributions.



- Combining entries of probability vectors preserves Dirichlet property, for example:

$$\begin{aligned} & (\pi_1, \dots, \pi_K) \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_K) \\ \Rightarrow \qquad & (\pi_1 + \pi_2, \pi_3, \dots, \pi_K) \sim \text{Dirichlet}(\alpha_1 + \alpha_2, \alpha_3, \dots, \alpha_K) \end{aligned}$$

- Generally, if $(I_1, \dots, I_j)$ is a partition of $(1, \dots, n)$:

$$\left(\sum_{i \in I_1} \pi_i, \dots, \sum_{i \in I_j} \pi_i \right) \sim \text{Dirichlet} \left(\sum_{i \in I_1} \alpha_i, \dots, \sum_{i \in I_j} \alpha_i \right)$$

- The converse of the agglomerative property is also true, for example if:

$$(\pi_1, \ldots, \pi_K) \sim \text{Dirichlet}(\alpha_1, \ldots, \alpha_K)$$
$$(\tau_1, \tau_2) \sim \text{Dirichlet}(\alpha_1\beta_1, \alpha_1\beta_2)$$

with $\beta_1 + \beta_2 = 1$,

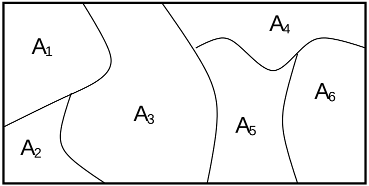
$$\Rightarrow \quad (\pi_1\tau_1, \pi_1\tau_2, \pi_2, \ldots, \pi_K) \sim \text{Dirichlet}(\alpha_1\beta_1, \alpha_2\beta_2, \alpha_2, \ldots, \alpha_K)$$

- A Dirichlet process (DP) is an “infinitely decimated” Dirichlet distribution:

$$
\begin{aligned}
1 &\sim \text{Dirichlet}(\alpha) \\
(\pi_1, \pi_2) &\sim \text{Dirichlet}(\alpha/2, \alpha/2) & \pi_1 + \pi_2 = 1 \\
(\pi_{11}, \pi_{12}, \pi_{21}, \pi_{22}) &\sim \text{Dirichlet}(\alpha/4, \alpha/4, \alpha/4, \alpha/4) & \pi_{i1} + \pi_{i2} = \pi_i \\
&\vdots
\end{aligned}
$$

- Each decimation step involves drawing from a Beta distribution (a Dirichlet with 2 components) and multiplying into the relevant entry.
- Demo: DPgenerate

- A probability measure is a function from subsets of a space $\mathbb{X}$ to $[0, 1]$ satisfying certain properties.
- A Dirichlet Process (DP) is a distribution over probability measures.
- Denote $G \sim \mathrm{DP}$ if G is a DP-distributed random probability measure.
- For any finite set of partitions $A_1 \dot\cup \ldots \dot\cup A_K = \mathbb{X}$, we require $(G(A_1), \ldots, G(A_K))$ to be Dirichlet distributed.



- A DP has two parameters:
  - Base distribution H, which is like the *mean* of the DP.
  - Strength parameter α, which is like an *inverse-variance* of the DP.
- We write:

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

if for any partition $(A_1, \ldots, A_K)$ of $\mathbb{X}$:

$$(G(A_1), \ldots, G(A_K)) \sim \mathrm{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K))$$

- A DP has two parameters:
  - Base distribution H, which is like the *mean* of the DP.
  - Strength parameter α, which is like an *inverse-variance* of the DP.
- We write:

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

- The first two cumulants of the DP:

$$\text{Expectation:} \qquad \mathbb{E}[G(A)] = H(A)$$

$$\text{Variance:} \qquad \mathbb{V}[G(A)] = \frac{H(A)(1 - H(A))}{\alpha + 1}$$

where A is any measurable subset of $\mathbb{X}$.

- A probability measure is a function from subsets of a space $\mathbb{X}$ to $[0, 1]$ satisfying certain properties.
- A DP is a distribution over probability measures such that marginals on finite partitions are Dirichlet distributed.
- How do we know that such an object exists?!?
- Kolmogorov Consistency Theorem: [Ferguson 1973].
- de Finetti's Theorem: Blackwell-MacQueen urn scheme, Chinese restaurant process, [Blackwell and MacQueen 1973, Aldous 1985].
- Stick-breaking Construction: [Sethuraman 1994].
- Gamma Process: [Ferguson 1973].

- A probability measure is a function from subsets of a space $\mathbb{X}$ to $[0, 1]$ satisfying certain properties.
- A DP is a distribution over probability measures such that marginals on finite partitions are Dirichlet distributed.
- How do we know that such an object exists?!?
- Kolmogorov Consistency Theorem: [Ferguson 1973].
- de Finetti's Theorem: Blackwell-MacQueen urn scheme, Chinese restaurant process, [Blackwell and MacQueen 1973, Aldous 1985].
- Stick-breaking Construction: [Sethuraman 1994].
- Gamma Process: [Ferguson 1973].

- A probability measure is a function from subsets of a space $\mathbb{X}$ to $[0, 1]$ satisfying certain properties.
- A DP is a distribution over probability measures such that marginals on finite partitions are Dirichlet distributed.
- How do we know that such an object exists?!?
- Kolmogorov Consistency Theorem: [Ferguson 1973].
- de Finetti's Theorem: Blackwell-MacQueen urn scheme, Chinese restaurant process, [Blackwell and MacQueen 1973, Aldous 1985].
- Stick-breaking Construction: [Sethuraman 1994].
- Gamma Process: [Ferguson 1973].

- A probability measure is a function from subsets of a space $\mathbb{X}$ to $[0, 1]$ satisfying certain properties.
- A DP is a distribution over probability measures such that marginals on finite partitions are Dirichlet distributed.
- How do we know that such an object exists?!?
- Kolmogorov Consistency Theorem: [Ferguson 1973].
- de Finetti's Theorem: Blackwell-MacQueen urn scheme, Chinese restaurant process, [Blackwell and MacQueen 1973, Aldous 1985].
- Stick-breaking Construction: [Sethuraman 1994].
- Gamma Process: [Ferguson 1973].

- A probability measure is a function from subsets of a space $\mathbb{X}$ to $[0, 1]$ satisfying certain properties.
- A DP is a distribution over probability measures such that marginals on finite partitions are Dirichlet distributed.
- How do we know that such an object exists?!?
- Kolmogorov Consistency Theorem: [Ferguson 1973].
- de Finetti's Theorem: Blackwell-MacQueen urn scheme, Chinese restaurant process, [Blackwell and MacQueen 1973, Aldous 1985].
- Stick-breaking Construction: [Sethuraman 1994].
- Gamma Process: [Ferguson 1973].

Outline

- Suppose $G \sim \mathrm{DP}(\alpha, H)$. G is a (random) probability measure over $\mathbb{X}$. We can treat it as a distribution over $\mathbb{X}$. Let

$$\theta_1, \ldots, \theta_n \sim G$$

be a random variable with distribution G.

- We saw in the demo that draws from a Dirichlet process seem to be discrete distributions. If so, then:

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

and there is positive probability that θ_i's can take on the same value θ_k^* for some k, i.e. the θ_i's cluster together.

- In this section we are concerned with representations of Dirichlet processes based upon both the clustering property and the sum of point masses.

- Suppose $G \sim \mathrm{DP}(\alpha, H)$. G is a (random) probability measure over $\mathbb{X}$. We can treat it as a distribution over $\mathbb{X}$. Let

$$\theta_1, \dots, \theta_n \sim G$$

be a random variable with distribution G.

- We saw in the demo that draws from a Dirichlet process seem to be discrete distributions. If so, then:

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

and there is positive probability that θ_i's can take on the same value θ_k^* for some k, i.e. the θ_i's cluster together.

- In this section we are concerned with representations of Dirichlet processes based upon both the clustering property and the sum of point masses.

- Suppose $G \sim \mathrm{DP}(\alpha, H)$. G is a (random) probability measure over $\mathbb{X}$. We can treat it as a distribution over $\mathbb{X}$. Let

$$\theta_1, \ldots, \theta_n \sim G$$

be a random variable with distribution G.

- We saw in the demo that draws from a Dirichlet process seem to be discrete distributions. If so, then:

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

and there is positive probability that θ_i's can take on the same value θ_k^* for some k, i.e. the θ_i's cluster together.

- In this section we are concerned with representations of Dirichlet processes based upon both the clustering property and the sum of point masses.

Posterior Dirichlet Processes

Sampling from a Dirichlet Process

- Suppose G is Dirichlet process distributed:

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

- G is a (random) probability measure over $\mathbb{X}$. We can treat it as a distribution over $\mathbb{X}$. Let

$$\theta \sim G$$

  be a random variable with distribution G.

- We are interested in:

$$p(\theta) = \int p(\theta|G)p(G)\,dG$$

$$p(G|\theta) = \frac{p(\theta|G)p(G)}{p(\theta)}$$

- Consider:

$$(\pi_1, \dots, \pi_K) \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_K)$$
$$z|(\pi_1, \dots, \pi_K) \sim \text{Discrete}(\pi_1, \dots, \pi_K)$$

  z is a multinomial variate, taking on value $i \in \{1, \dots, n\}$ with probability π_i.

- Then:

$$z \sim \text{Discrete}\left(\frac{\alpha_1}{\sum_i \alpha_i}, \dots, \frac{\alpha_K}{\sum_i \alpha_i}\right)$$
$$(\pi_1, \dots, \pi_K)|z \sim \text{Dirichlet}(\alpha_1 + \delta_1(z), \dots, \alpha_K + \delta_K(z))$$

  where $\delta_i(z) = 1$ if z takes on value i, 0 otherwise.

- Converse also true.

Posterior Dirichlet Processes

- Fix a partition $(A_1, \dots, A_K)$ of $\mathbb{X}$. Then

$$(G(A_1), \dots, G(A_K)) \sim \text{Dirichlet}(\alpha H(A_1), \dots, \alpha H(A_K))$$
$$P(\theta \in A_i | G) = G(A_i)$$

- Using Dirichlet-multinomial conjugacy,

$$P(\theta \in A_i) = H(A_i)$$
$$(G(A_1), \dots, G(A_K)) | \theta \sim \text{Dirichlet}(\alpha H(A_1) + \delta_\theta(A_1), \dots, \alpha H(A_K) + \delta_\theta(A_K))$$

- The above is true for every finite partition of $\mathbb{X}$. In particular, taking a really fine partition,

$$p(\theta) d\theta = H(d\theta)$$

- Also, the posterior $G|\theta$ is also a Dirichlet process:

$$G|\theta \sim \text{DP}\left(\alpha + 1, \frac{\alpha H + \delta_\theta}{\alpha + 1}\right)$$

Posterior Dirichlet Processes

- Fix a partition $(A_1, \ldots, A_K)$ of $\mathbb{X}$. Then

$$(G(A_1), \ldots, G(A_K)) \sim \text{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K))$$
$$P(\theta \in A_i | G) = G(A_i)$$

- Using Dirichlet-multinomial conjugacy,

$$P(\theta \in A_i) = H(A_i)$$
$$(G(A_1), \ldots, G(A_K))|\theta \sim \text{Dirichlet}(\alpha H(A_1) + \delta_\theta(A_1), \ldots, \alpha H(A_K) + \delta_\theta(A_K))$$

- The above is true for every finite partition of $\mathbb{X}$. In particular, taking a really fine partition,

$$p(\theta)d\theta = H(d\theta)$$

- Also, the posterior $G|\theta$ is also a Dirichlet process:

$$G|\theta \sim \text{DP}\left(\alpha + 1, \frac{\alpha H + \delta_\theta}{\alpha + 1}\right)$$

Posterior Dirichlet Processes

- Fix a partition $(A_1, \ldots, A_K)$ of $\mathbb{X}$. Then

$$\begin{aligned}(G(A_1), \ldots, G(A_K)) &\sim \text{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K)) \\ P(\theta \in A_i | G) &= G(A_i)\end{aligned}$$

- Using Dirichlet-multinomial conjugacy,

$$\begin{aligned}P(\theta \in A_i) &= H(A_i) \\ (G(A_1), \ldots, G(A_K))|\theta &\sim \text{Dirichlet}(\alpha H(A_1)+\delta_\theta(A_1), \ldots, \alpha H(A_K)+\delta_\theta(A_K))\end{aligned}$$

- The above is true for every finite partition of $\mathbb{X}$. In particular, taking a really fine partition,

$$p(\theta)d\theta = H(d\theta)$$

- Also, the posterior $G|\theta$ is also a Dirichlet process:

$$G|\theta \sim \text{DP}\left(\alpha + 1, \frac{\alpha H + \delta_\theta}{\alpha + 1}\right)$$

Posterior Dirichlet Processes

- Fix a partition $(A_1, \ldots, A_K)$ of $\mathbb{X}$. Then

$$\begin{aligned}(G(A_1), \ldots, G(A_K)) &\sim \text{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K)) \\ P(\theta \in A_i | G) &= G(A_i)\end{aligned}$$

- Using Dirichlet-multinomial conjugacy,

$$\begin{aligned}P(\theta \in A_i) &= H(A_i) \\ (G(A_1), \ldots, G(A_K))|\theta &\sim \text{Dirichlet}(\alpha H(A_1) + \delta_\theta(A_1), \ldots, \alpha H(A_K) + \delta_\theta(A_K))\end{aligned}$$

- The above is true for every finite partition of $\mathbb{X}$. In particular, taking a really fine partition,

$$p(\theta)d\theta = H(d\theta)$$

- Also, the posterior $G|\theta$ is also a Dirichlet process:

$$G|\theta \sim \text{DP}\left(\alpha + 1, \frac{\alpha H + \delta_\theta}{\alpha + 1}\right)$$

Posterior Dirichlet Processes

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta | G &\sim G \end{aligned} \qquad \Longleftrightarrow \qquad \begin{aligned} \theta &\sim H \\ G | \theta &\sim \mathrm{DP}\left(\alpha + 1, \frac{\alpha H + \delta_\theta}{\alpha + 1}\right) \end{aligned}$$

Blackwell-MacQueen Urn Scheme

- First sample:

$$\theta_1 | G \sim G \qquad G \sim \mathrm{DP}(\alpha, H)$$
$$\iff \quad \theta_1 \sim H \qquad G|\theta_1 \sim \mathrm{DP}(\alpha + 1, \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1})$$

- Second sample:

$$\theta_2 | \theta_1, G \sim G \qquad G|\theta_1 \sim \mathrm{DP}(\alpha + 1, \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1})$$
$$\iff \quad \theta_2 | \theta_1 \sim \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1} \qquad G|\theta_1, \theta_2 \sim \mathrm{DP}(\alpha + 2, \tfrac{\alpha H + \delta_{\theta_1} + \delta_{\theta_2}}{\alpha + 2})$$

- n^{th} sample

$$\theta_n | \theta_{1:n-1}, G \sim G \qquad G|\theta_{1:n-1} \sim \mathrm{DP}(\alpha + n - 1, \tfrac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1})$$
$$\iff \quad \theta_n | \theta_{1:n-1} \sim \tfrac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1} \qquad G|\theta_{1:n} \sim \mathrm{DP}(\alpha + n, \tfrac{\alpha H + \sum_{i=1}^{n} \delta_{\theta_i}}{\alpha + n})$$

Blackwell-MacQueen Urn Scheme

- First sample:

$$\theta_1 | G \sim G \qquad G \sim \mathrm{DP}(\alpha, H)$$

$$\iff \theta_1 \sim H \qquad G|\theta_1 \sim \mathrm{DP}(\alpha + 1, \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1})$$

- Second sample:

$$\theta_2 | \theta_1, G \sim G \qquad G|\theta_1 \sim \mathrm{DP}(\alpha + 1, \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1})$$

$$\iff \theta_2 | \theta_1 \sim \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1} \qquad G|\theta_1, \theta_2 \sim \mathrm{DP}(\alpha + 2, \tfrac{\alpha H + \delta_{\theta_1} + \delta_{\theta_2}}{\alpha + 2})$$

- n^{th} sample

$$\theta_n | \theta_{1:n-1}, G \sim G \qquad G|\theta_{1:n-1} \sim \mathrm{DP}(\alpha + n - 1, \tfrac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1})$$

$$\iff \theta_n | \theta_{1:n-1} \sim \tfrac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1} \qquad G|\theta_{1:n} \sim \mathrm{DP}(\alpha + n, \tfrac{\alpha H + \sum_{i=1}^{n} \delta_{\theta_i}}{\alpha + n})$$

Blackwell-MacQueen Urn Scheme

- First sample:

$$\begin{aligned} && \theta_1 \mid G \sim G && G \sim \mathrm{DP}(\alpha, H) \\ \iff && \theta_1 \sim H && G \mid \theta_1 \sim \mathrm{DP}\left(\alpha + 1, \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1}\right) \end{aligned}$$

- Second sample:

$$\begin{aligned} && \theta_2 \mid \theta_1, G \sim G && G \mid \theta_1 \sim \mathrm{DP}\left(\alpha + 1, \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1}\right) \\ \iff && \theta_2 \mid \theta_1 \sim \tfrac{\alpha H + \delta_{\theta_1}}{\alpha + 1} && G \mid \theta_1, \theta_2 \sim \mathrm{DP}\left(\alpha + 2, \tfrac{\alpha H + \delta_{\theta_1} + \delta_{\theta_2}}{\alpha + 2}\right) \end{aligned}$$

- n^{th} sample

$$\begin{aligned} && \theta_n \mid \theta_{1:n-1}, G \sim G && G \mid \theta_{1:n-1} \sim \mathrm{DP}\left(\alpha + n - 1, \tfrac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1}\right) \\ \iff && \theta_n \mid \theta_{1:n-1} \sim \tfrac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1} && G \mid \theta_{1:n} \sim \mathrm{DP}\left(\alpha + n, \tfrac{\alpha H + \sum_{i=1}^{n} \delta_{\theta_i}}{\alpha + n}\right) \end{aligned}$$

Blackwell-MacQueen Urn Scheme

- Blackwell-MacQueen urn scheme produces a sequence $\theta_1, \theta_2, \ldots$ with the following conditionals:

$$\theta_n | \theta_{1:n-1} \sim \frac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1}$$

- Picking balls of different colors from an urn:
  - Start with no balls in the urn.
  - with probability $\propto \alpha$, draw $\theta_n \sim H$, and add a ball of that color into the urn.
  - With probability $\propto n - 1$, pick a ball at random from the urn, record θ_n to be its color, return the ball into the urn and place a second ball of same color into urn.
- Blackwell-MacQueen urn scheme is like a "representer" for the DP—a finite projection of an infinite object.

Exchangeability and de Finetti's Theorem

- Starting with a DP, we constructed Blackwell-MacQueen urn scheme.
- The reverse is possible using de Finetti's Theorem.
- Since θ_i are iid $\sim G$, their joint distribution is invariant to permutations, thus $\theta_1, \theta_2, \ldots$ are exchangeable.
- Thus a distribution over measures must exist making them iid.
- This is the DP.

Chinese Restaurant Process

- Draw $\theta_1, \ldots, \theta_n$ from a Blackwell-MacQueen urn scheme.
- They take on $K < n$ distinct values, say $\theta_1^*, \ldots, \theta_K^*$.
- This defines a partition of $1, \ldots, n$ into K clusters, such that if i is in cluster k, then $\theta_i = \theta_k^*$.
- Random draws $\theta_1, \ldots, \theta_n$ from a Blackwell-MacQueen urn scheme induces a random partition of $1, \ldots, n$.
- The induced distribution over partitions is a Chinese restaurant process (CRP).

Chinese Restaurant Process

- Generating from the CRP:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k}{\alpha+n-1}$ where n_k is the number of customers at table k.
    - A new table $K+1$ with probability $\frac{\alpha}{\alpha+n-1}$.
  - Customers $\Leftrightarrow$ integers, tables $\Leftrightarrow$ clusters.
- The CRP exhibits the clustering property of the DP.

Chinese Restaurant Process

- Generating from the CRP:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k}{\alpha+n-1}$ where n_k is the number of customers at table k.
    - A new table $K+1$ with probability $\frac{\alpha}{\alpha+n-1}$.
  - Customers $\Leftrightarrow$ integers, tables $\Leftrightarrow$ clusters.
- The CRP exhibits the clustering property of the DP.

Chinese Restaurant Process

- To get back from the CRP to Blackwell-MacQueen urn scheme, simply draw

$$\theta_k^* \sim H$$

for $k = 1, \ldots, K$, then for $i = 1, \ldots, n$ set

$$\theta_i = \theta_{z_i}^*$$

where z_i is the table that customer i sat at.

- The CRP teases apart the clustering property of the DP, from the base distribution.

Stick-breaking Construction

- Returning to the posterior process:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta|G &\sim G \end{aligned} \qquad \Leftrightarrow \qquad \begin{aligned} \theta &\sim H \\ G|\theta &\sim \mathrm{DP}(\alpha+1, \tfrac{\alpha H+\delta_\theta}{\alpha+1}) \end{aligned}$$

- Consider a partition $(\theta, \mathbb{X}\backslash\theta)$ of $\mathbb{X}$. We have:

$$\begin{aligned} (G(\theta), G(\mathbb{X}\backslash\theta)) &\sim \mathrm{Dirichlet}((\alpha+1)\tfrac{\alpha H+\delta_\theta}{\alpha+1}(\theta), (\alpha+1)\tfrac{\alpha H+\delta_\theta}{\alpha+1}(\mathbb{X}\backslash\theta)) \\ &= \mathrm{Dirichlet}(1, \alpha) \end{aligned}$$

- G has a point mass located at θ:

$$G = \beta\delta_\theta + (1-\beta)G' \qquad \text{with} \qquad \beta \sim \mathrm{Beta}(1, \alpha)$$

  and G' is the (renormalized) probability measure with the point mass removed.

- What is G'?

Stick-breaking Construction

- Returning to the posterior process:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta \mid G &\sim G \end{aligned} \qquad \Leftrightarrow \qquad \begin{aligned} \theta &\sim H \\ G \mid \theta &\sim \mathrm{DP}\left(\alpha + 1, \tfrac{\alpha H + \delta_\theta}{\alpha + 1}\right) \end{aligned}$$

- Consider a partition $(\theta, \mathbb{X}\backslash\theta)$ of $\mathbb{X}$. We have:

$$\begin{aligned} (G(\theta), G(\mathbb{X}\backslash\theta)) &\sim \mathrm{Dirichlet}\left((\alpha+1)\tfrac{\alpha H + \delta_\theta}{\alpha+1}(\theta), (\alpha+1)\tfrac{\alpha H + \delta_\theta}{\alpha+1}(\mathbb{X}\backslash\theta)\right) \\ &= \mathrm{Dirichlet}(1, \alpha) \end{aligned}$$

- G has a point mass located at θ:

$$G = \beta\delta_\theta + (1-\beta)G' \qquad \text{with} \qquad \beta \sim \mathrm{Beta}(1, \alpha)$$

and G' is the (renormalized) probability measure with the point mass removed.

- What is G'?

Stick-breaking Construction

- Returning to the posterior process:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta|G &\sim G \end{aligned} \qquad \Leftrightarrow \qquad \begin{aligned} \theta &\sim H \\ G|\theta &\sim \mathrm{DP}(\alpha+1, \tfrac{\alpha H+\delta_\theta}{\alpha+1}) \end{aligned}$$

- Consider a partition $(\theta, \mathbb{X}\backslash\theta)$ of $\mathbb{X}$. We have:

$$\begin{aligned} (G(\theta), G(\mathbb{X}\backslash\theta)) &\sim \mathrm{Dirichlet}((\alpha+1)\tfrac{\alpha H+\delta_\theta}{\alpha+1}(\theta), (\alpha+1)\tfrac{\alpha H+\delta_\theta}{\alpha+1}(\mathbb{X}\backslash\theta)) \\ &= \mathrm{Dirichlet}(1, \alpha) \end{aligned}$$

- G has a point mass located at θ:

$$G = \beta\delta_\theta + (1-\beta)G' \qquad \text{with} \qquad \beta \sim \mathrm{Beta}(1, \alpha)$$

  and G' is the (renormalized) probability measure with the point mass removed.

- What is G'?

Stick-breaking Construction

- Returning to the posterior process:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta | G &\sim G \end{aligned} \qquad \Leftrightarrow \qquad \begin{aligned} \theta &\sim H \\ G|\theta &\sim \mathrm{DP}(\alpha+1, \tfrac{\alpha H + \delta_\theta}{\alpha+1}) \end{aligned}$$

- Consider a partition $(\theta, \mathbb{X}\backslash\theta)$ of $\mathbb{X}$. We have:

$$\begin{aligned} (G(\theta), G(\mathbb{X}\backslash\theta)) &\sim \mathrm{Dirichlet}((\alpha+1)\tfrac{\alpha H + \delta_\theta}{\alpha+1}(\theta), (\alpha+1)\tfrac{\alpha H + \delta_\theta}{\alpha+1}(\mathbb{X}\backslash\theta)) \\ &= \mathrm{Dirichlet}(1, \alpha) \end{aligned}$$

- G has a point mass located at θ:

$$G = \beta\delta_\theta + (1-\beta)G' \qquad \text{with} \qquad \beta \sim \mathrm{Beta}(1, \alpha)$$

  and G' is the (renormalized) probability measure with the point mass removed.

- What is G'?

Stick-breaking Construction

- Currently, we have:

$$
\begin{aligned}
G &\sim \mathrm{DP}(\alpha, H) \\
\theta &\sim G
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
G &\sim \mathrm{DP}\left(\alpha + 1, \tfrac{\alpha H + \delta_\theta}{\alpha+1}\right) \\
G &= \beta\delta_\theta + (1-\beta)G' \\
\theta &\sim H \\
\beta &\sim \mathrm{Beta}(1, \alpha)
\end{aligned}
$$

- Consider a further partition $(\theta, A_1, \ldots, A_K)$ of $\mathbb{X}$:

$$
\begin{aligned}
&(G(\theta), G(A_1), \ldots, G(A_K)) \\
=&(\beta, (1-\beta)G'(A_1), \ldots, (1-\beta)G'(A_K))
\end{aligned}
$$

- The agglomerative/decimative property of Dirichlet implies:

$$
\begin{aligned}
(G'(A_1), \ldots, G'(A_K)) &\sim \mathrm{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K)) \\
G' &\sim \mathrm{DP}(\alpha, H)
\end{aligned}
$$

Stick-breaking Construction

- Currently, we have:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta &\sim G \end{aligned} \qquad \Rightarrow \qquad \begin{aligned} G &\sim \mathrm{DP}\left(\alpha+1, \tfrac{\alpha H + \delta_\theta}{\alpha+1}\right) \\ G &= \beta\delta_\theta + (1-\beta)G' \\ \theta &\sim H \\ \beta &\sim \mathrm{Beta}(1, \alpha) \end{aligned}$$

- Consider a further partition $(\theta, A_1, \ldots, A_K)$ of $\mathbb{X}$:

$$\begin{aligned} &(G(\theta), G(A_1), \ldots, G(A_K)) \\ =&(\beta, (1-\beta)G'(A_1), \ldots, (1-\beta)G'(A_K)) \\ \sim&\,\mathrm{Dirichlet}(1, \alpha H(A_1), \ldots, \alpha H(A_K)) \end{aligned}$$

- The agglomerative/decimative property of Dirichlet implies:

$$\begin{aligned} (G'(A_1), \ldots, G'(A_K)) &\sim \mathrm{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K)) \\ G' &\sim \mathrm{DP}(\alpha, H) \end{aligned}$$

Stick-breaking Construction

- Currently, we have:

$$
\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta &\sim G \end{aligned}
\qquad \Rightarrow \qquad
\begin{aligned} G &\sim \mathrm{DP}\left(\alpha+1, \tfrac{\alpha H + \delta_\theta}{\alpha+1}\right) \\ G &= \beta\delta_\theta + (1-\beta)G' \\ \theta &\sim H \\ \beta &\sim \mathrm{Beta}(1, \alpha) \end{aligned}
$$

- Consider a further partition $(\theta, A_1, \ldots, A_K)$ of $\mathbb{X}$:

$$
\begin{aligned} &(G(\theta), G(A_1), \ldots, G(A_K)) \\ =&(\beta, (1-\beta)G'(A_1), \ldots, (1-\beta)G'(A_K)) \\ \sim&\, \mathrm{Dirichlet}(1, \alpha H(A_1), \ldots, \alpha H(A_K)) \end{aligned}
$$

- The agglomerative/decimative property of Dirichlet implies:

$$
\begin{aligned} (G'(A_1), \ldots, G'(A_K)) &\sim \mathrm{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_K)) \\ G' &\sim \mathrm{DP}(\alpha, H) \end{aligned}
$$

Stick-breaking Construction

- We have:

$$
\begin{aligned}
G &\sim \mathrm{DP}(\alpha, H) \\
G &= \beta_1 \delta_{\theta_1^*} + (1 - \beta_1) G_1 \\
G &= \beta_1 \delta_{\theta_1^*} + (1 - \beta_1)(\beta_2 \delta_{\theta_2^*} + (1 - \beta_2) G_2) \\
&\vdots \\
G &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*}
\end{aligned}
$$

  where

$$
\pi_k = \beta_k \textstyle\prod_{i=1}^{k-1} (1 - \beta_i) \qquad \beta_k \sim \mathrm{Beta}(1, \alpha) \qquad \theta_k^* \sim H
$$

- This is the stick-breaking construction.
- Demo: SBgenerate

Stick-breaking Construction

- Starting with a DP, we showed that draws from the DP looks like a sum of point masses, with masses drawn from a stick-breaking construction.
- The steps are limited by assumptions of regularity on $\mathbb{X}$ and smoothness on H.
- [Sethuraman 1994] started with the stick-breaking construction, and showed that draws are indeed DP distributed, under very general conditions.

- Posterior Dirichlet process:

$$\begin{array}{c} G \sim \mathrm{DP}(\alpha, H) \\ \theta | G \sim G \end{array} \quad \Longleftrightarrow \quad \begin{array}{c} \theta \sim H \\ G|\theta \sim \mathrm{DP}\left(\alpha + 1, \frac{\alpha H + \delta_\theta}{\alpha+1}\right) \end{array}$$

- Blackwell-MacQueen urn scheme:

$$\theta_n | \theta_{1:n-1} \sim \frac{\alpha H + \sum_{i=1}^{n-1} \delta_{\theta_i}}{\alpha + n - 1}$$

- Chinese restaurant process:

$$p(\text{customer } n \text{ sat at table } k|\text{past}) = \begin{cases} \frac{n_k}{n-1+\alpha} & \text{if occupied table} \\ \frac{\alpha}{n-1+\alpha} & \text{if new table} \end{cases}$$

- Stick-breaking construction:

$$\pi_k = \beta_k \prod_{i=1}^{k-1} (1 - \beta_i) \qquad \beta_k \sim \mathrm{Beta}(1, \alpha) \qquad \theta_k^* \sim H \qquad G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*}$$

Outline

Density Estimation

- Recall our approach to density estimation with Dirichlet processes:

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

- The above does not work. Why?
- Problem: G is a discrete distribution; in particular it has no density!
- Solution: Convolve the DP with a smooth distribution:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ F_x(\cdot) &= \int F(\cdot|\theta)dG(\theta) \\ x_i &\sim F_x \end{aligned} \quad \Rightarrow \quad \begin{aligned} G &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*} \\ F_x(\cdot) &= \sum_{k=1}^{\infty} \pi_k F(\cdot|\theta_k^*) \\ x_i &\sim F_x \end{aligned}$$

Density Estimation

- Recall our approach to density estimation with Dirichlet processes:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ x_i &\sim G \end{aligned}$$

- The above does not work. Why?
- Problem: G is a discrete distribution; in particular it has no density!
- Solution: Convolve the DP with a smooth distribution:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ F_x(\cdot) &= \int F(\cdot|\theta) dG(\theta) \\ x_i &\sim F_x \end{aligned} \quad \Rightarrow \quad \begin{aligned} G &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*} \\ F_x(\cdot) &= \sum_{k=1}^{\infty} \pi_k F(\cdot|\theta_k^*) \\ x_i &\sim F_x \end{aligned}$$

Density Estimation

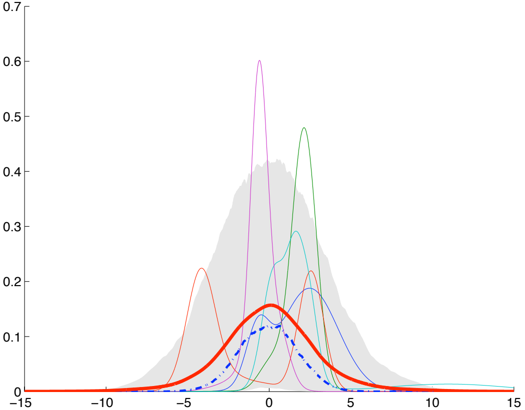
- Recall our approach to density estimation with Dirichlet processes:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ x_i &\sim G \end{aligned}$$

- The above does not work. Why?
- Problem: G is a discrete distribution; in particular it has no density!
- Solution: Convolve the DP with a smooth distribution:

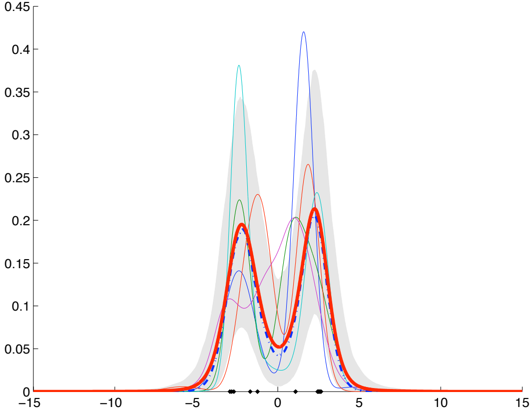
$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ F_x(\cdot) &= \int F(\cdot|\theta) dG(\theta) \\ x_i &\sim F_x \end{aligned} \qquad \Rightarrow \qquad \begin{aligned} G &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*} \\ F_x(\cdot) &= \sum_{k=1}^{\infty} \pi_k F(\cdot|\theta_k^*) \\ x_i &\sim F_x \end{aligned}$$

Density Estimation



$F(\cdot|\mu,\Sigma)$ is Gaussian with mean μ, covariance Σ.
$H(\mu,\Sigma)$ is Gaussian-inverse-Wishart conjugate prior.

Density Estimation



$F(\cdot|\mu, \Sigma)$ is Gaussian with mean μ, covariance Σ.
$H(\mu, \Sigma)$ is Gaussian-inverse-Wishart conjugate prior.

Clustering

- Recall our approach to density estimation:

$$
\begin{aligned}
G &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*} \sim \mathrm{DP}(\alpha, H) \\
F_x(\cdot) &= \sum_{k=1}^{\infty} \pi_k F(\cdot|\theta_k^*) \\
x_i &\sim F_x
\end{aligned}
$$

- Above model equivalent to:

$$
\begin{aligned}
z_i &\sim \mathrm{Discrete}(\boldsymbol{\pi}) \\
\theta_i &= \theta_{z_i}^* \\
x_i|z_i &\sim F(\cdot|\theta_i) = F(\cdot|\theta_{z_i}^*)
\end{aligned}
$$

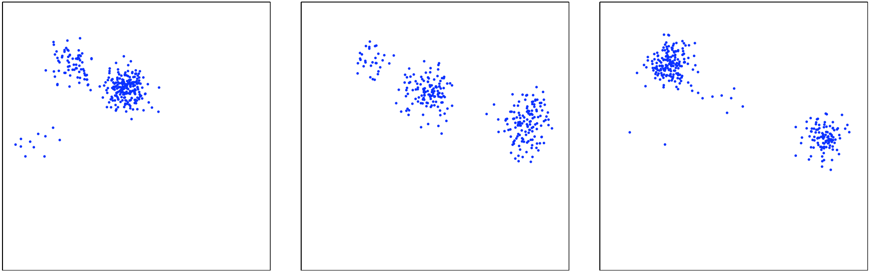
- This is simply a mixture model with an infinite number of components. This is called a DP mixture model.

Clustering

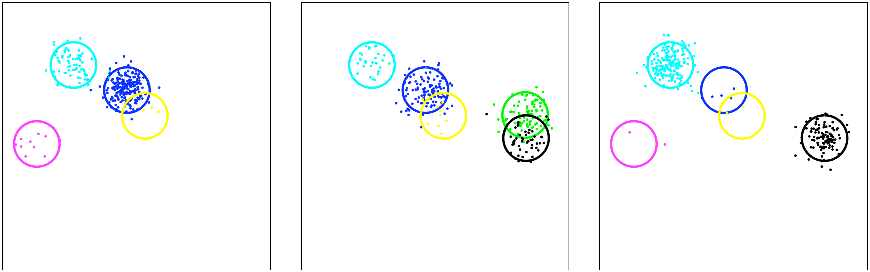
- DP mixture models are used in a variety of clustering applications, where the number of clusters is not known a priori.
- They are also used in applications in which we believe the number of clusters grows without bound as the amount of data grows.
- DPs have also found uses in applications beyond clustering, where the number of latent objects is not known or unbounded.
  - Nonparametric probabilistic context free grammars.
  - Visual scene analysis.
  - Infinite hidden Markov models/trees.
  - Haplotype inference.
  - ...
- In many such applications it is important to be able to model the same set of objects in different contexts.
- This corresponds to the problem of grouped clustering and can be tackled using hierarchical Dirichlet processes.

[Teh et al. 2006]

Grouped Clustering



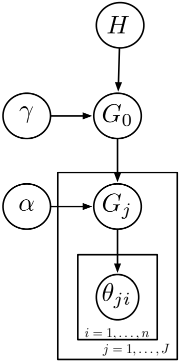
Grouped Clustering



Hierarchical Dirichlet Processes

- Hierarchical Dirichlet process:

$$G_0|\gamma, H \sim \mathrm{DP}(\gamma, H)$$
$$G_j|\alpha, G_0 \sim \mathrm{DP}(\alpha, G_0)$$
$$\theta_{ji}|G_j \sim G_j$$

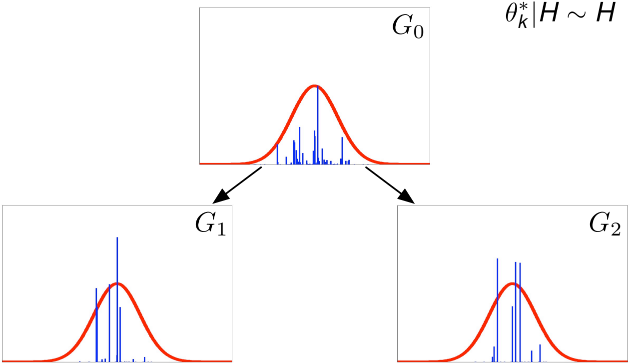


Hierarchical Dirichlet Processes

$$G_0|\gamma, H \sim \mathrm{DP}(\gamma, H) \qquad G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k^*} \qquad \boldsymbol{\beta}|\gamma \sim \mathrm{Stick}(\gamma)$$

$$G_j|\alpha, G_0 \sim \mathrm{DP}(\alpha, G_0) \qquad G_j = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k^*} \qquad \boldsymbol{\pi}_j|\alpha, \boldsymbol{\beta} \sim \mathrm{DP}(\alpha, \boldsymbol{\beta})$$

$$\theta_k^*|H \sim H$$



Summary

- Dirichlet process is “just” a glorified Dirichlet distribution.
- Draws from a DP are probability measures consisting of a weighted sum of point masses.
- Many representations: Blackwell-MacQueen urn scheme, Chinese restaurant process, stick-breaking construction.
- DP mixture models are mixture models with countably infinite number of components.

- I have not delved into:
  - Applications.
  - Generalizations, extensions, other nonparametric processes.
  - Inference: MCMC sampling, variational approximation.
- Also see the tutorial material from Ghaharamani, Jordan and Tresp.

Summary

- Dirichlet process is “just” a glorified Dirichlet distribution.
- Draws from a DP are probability measures consisting of a weighted sum of point masses.
- Many representations: Blackwell-MacQueen urn scheme, Chinese restaurant process, stick-breaking construction.
- DP mixture models are mixture models with countably infinite number of components.

- I have not delved into:
  - Applications.
  - Generalizations, extensions, other nonparametric processes.
  - Inference: MCMC sampling, variational approximation.
- Also see the tutorial material from Ghaharamani, Jordan and Tresp.

Bayesian Nonparametrics

- Parametric models can only capture a bounded amount of information from data, since they have bounded complexity.
- Real data is often complex and the parametric assumption is often wrong.
- Nonparametric models allow relaxation of the parametric assumption, bringing significant flexibility to our models of the world.
- Nonparametric models can also often lead to model selection/averaging behaviours without the cost of actually doing model selection/averaging.
- Nonparametric models are gaining popularity, spurred by growth in computational resources and inference algorithms.
- In addition to DPs, HDPs and their generalizations, other nonparametric models include Indian buffet processes, beta processes, tree processes...

[Tutorials at Workshop on Bayesian Nonparametric Regression, Isaac Newton Institute, Cambridge, July 2007]

Bayesian Nonparametrics

- Parametric models can only capture a bounded amount of information from data, since they have bounded complexity.
- Real data is often complex and the parametric assumption is often wrong.
- Nonparametric models allow relaxation of the parametric assumption, bringing significant flexibility to our models of the world.
- Nonparametric models can also often lead to model selection/averaging behaviours without the cost of actually doing model selection/averaging.
- Nonparametric models are gaining popularity, spurred by growth in computational resources and inference algorithms.
- In addition to DPs, HDPs and their generalizations, other nonparametric models include Indian buffet processes, beta processes, tree processes...

[Tutorials at Workshop on Bayesian Nonparametric Regression, Isaac Newton Institute, Cambridge, July 2007]

Bayesian Nonparametrics

- Parametric models can only capture a bounded amount of information from data, since they have bounded complexity.
- Real data is often complex and the parametric assumption is often wrong.
- Nonparametric models allow relaxation of the parametric assumption, bringing significant flexibility to our models of the world.
- Nonparametric models can also often lead to model selection/averaging behaviours without the cost of actually doing model selection/averaging.
- Nonparametric models are gaining popularity, spurred by growth in computational resources and inference algorithms.
- In addition to DPs, HDPs and their generalizations, other nonparametric models include Indian buffet processes, beta processes, tree processes...

[Tutorials at Workshop on Bayesian Nonparametric Regression, Isaac Newton Institute, Cambridge, July 2007]

Bayesian Nonparametrics

- Parametric models can only capture a bounded amount of information from data, since they have bounded complexity.
- Real data is often complex and the parametric assumption is often wrong.
- Nonparametric models allow relaxation of the parametric assumption, bringing significant flexibility to our models of the world.
- Nonparametric models can also often lead to model selection/averaging behaviours without the cost of actually doing model selection/averaging.
- Nonparametric models are gaining popularity, spurred by growth in computational resources and inference algorithms.
- In addition to DPs, HDPs and their generalizations, other nonparametric models include Indian buffet processes, beta processes, tree processes...

[Tutorials at Workshop on Bayesian Nonparametric Regression, Isaac Newton Institute, Cambridge, July 2007]

Bayesian Nonparametrics

- Parametric models can only capture a bounded amount of information from data, since they have bounded complexity.
- Real data is often complex and the parametric assumption is often wrong.
- Nonparametric models allow relaxation of the parametric assumption, bringing significant flexibility to our models of the world.
- Nonparametric models can also often lead to model selection/averaging behaviours without the cost of actually doing model selection/averaging.
- Nonparametric models are gaining popularity, spurred by growth in computational resources and inference algorithms.
- In addition to DPs, HDPs and their generalizations, other nonparametric models include Indian buffet processes, beta processes, tree processes...

[Tutorials at Workshop on Bayesian Nonparametric Regression, Isaac Newton Institute, Cambridge, July 2007]

Outline

Exploring the Dirichlet Process

- Before using DPs, it is important to understand its properties, so that we understand what prior assumptions we are imposing on our models.
- In this practical course we shall work towards implementing a DP mixture model to cluster NIPS papers, thus the relevant properties are the clustering properties of the DP.
- Consider the Chinese restaurant process representation of DPs:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k}{\alpha+n-1}$ where n_k is the number of customers at table k.
    - A new table $K+1$ with probability $\frac{\alpha}{\alpha+n-1}$.
- How does number of clusters K scale as a function of α and of n (on average)?
- How does the number n_k of customers sitting around table k depend on k and n (on average)?

Exploring the Dirichlet Process

- Before using DPs, it is important to understand its properties, so that we understand what prior assumptions we are imposing on our models.
- In this practical course we shall work towards implementing a DP mixture model to cluster NIPS papers, thus the relevant properties are the clustering properties of the DP.
- Consider the Chinese restaurant process representation of DPs:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k}{\alpha+n-1}$ where n_k is the number of customers at table k.
    - A new table $K+1$ with probability $\frac{\alpha}{\alpha+n-1}$.
- How does number of clusters K scale as a function of α and of n (on average)?
- How does the number n_k of customers sitting around table k depend on k and n (on average)?

Exploring the Dirichlet Process

- Before using DPs, it is important to understand its properties, so that we understand what prior assumptions we are imposing on our models.
- In this practical course we shall work towards implementing a DP mixture model to cluster NIPS papers, thus the relevant properties are the clustering properties of the DP.
- Consider the Chinese restaurant process representation of DPs:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k}{\alpha+n-1}$ where n_k is the number of customers at table k.
    - A new table $K+1$ with probability $\frac{\alpha}{\alpha+n-1}$.
- How does number of clusters K scale as a function of α and of n (on average)?
- How does the number n_k of customers sitting around table k depend on k and n (on average)?

Exploring the Dirichlet Process

- Before using DPs, it is important to understand its properties, so that we understand what prior assumptions we are imposing on our models.
- In this practical course we shall work towards implementing a DP mixture model to cluster NIPS papers, thus the relevant properties are the clustering properties of the DP.
- Consider the Chinese restaurant process representation of DPs:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k}{\alpha+n-1}$ where n_k is the number of customers at table k.
    - A new table $K+1$ with probability $\frac{\alpha}{\alpha+n-1}$.
- How does number of clusters K scale as a function of α and of n (on average)?
- How does the number n_k of customers sitting around table k depend on k and n (on average)?

Exploring the Pitman-Yor Process

- Sometimes the assumptions embedded in using DPs to model data are inappropriate.
- The Pitman-Yor process is a generalization of the DP that often has more appropriate properties.
- It has two parameters: d and α with $0 \leq d < 1$ and $\alpha > -d$.

  When $d = 0$ the Pitman-Yor process reduces to a DP.
- It also has a Chinese restaurant process representation:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k - d}{\alpha + n - 1}$ where n_k is the number of customers at table k.
    - A new table $K + 1$ with probability $\frac{\alpha + Kd}{\alpha + n - 1}$.
- How does K scale as a function of n, α and d (on average)?

Exploring the Pitman-Yor Process

- Sometimes the assumptions embedded in using DPs to model data are inappropriate.
- The Pitman-Yor process is a generalization of the DP that often has more appropriate properties.
- It has two parameters: d and α with $0 \leq d < 1$ and $\alpha > -d$.
  When $d = 0$ the Pitman-Yor process reduces to a DP.
- It also has a Chinese restaurant process representation:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k - d}{\alpha + n - 1}$ where n_k is the number of customers at table k.
    - A new table $K + 1$ with probability $\frac{\alpha + Kd}{\alpha + n - 1}$.
- How does K scale as a function of n, α and d (on average)?

Exploring the Pitman-Yor Process

- Sometimes the assumptions embedded in using DPs to model data are inappropriate.
- The Pitman-Yor process is a generalization of the DP that often has more appropriate properties.
- It has two parameters: d and α with $0 \leq d < 1$ and $\alpha > -d$.
  When $d = 0$ the Pitman-Yor process reduces to a DP.
- It also has a Chinese restaurant process representation:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k - d}{\alpha + n - 1}$ where n_k is the number of customers at table k.
    - A new table $K + 1$ with probability $\frac{\alpha + Kd}{\alpha + n - 1}$.
- How does K scale as a function of n, α and d (on average)?

Exploring the Pitman-Yor Process

- Sometimes the assumptions embedded in using DPs to model data are inappropriate.
- The Pitman-Yor process is a generalization of the DP that often has more appropriate properties.
- It has two parameters: d and α with $0 \leq d < 1$ and $\alpha > -d$.
  When $d = 0$ the Pitman-Yor process reduces to a DP.
- It also has a Chinese restaurant process representation:
  - First customer sits at the first table.
  - Customer n sits at:
    - Table k with probability $\frac{n_k - d}{\alpha + n - 1}$ where n_k is the number of customers at table k.
    - A new table $K + 1$ with probability $\frac{\alpha + Kd}{\alpha + n - 1}$.
- How does K scale as a function of n, α and d (on average)?

Dirichlet Process Mixture Models

- We model a data set $x_1, \ldots, x_n$ using the following model:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta_i | G &\sim G \\ x_i | \theta_i &\sim F(\cdot|\theta_i) & \text{for } i = 1, \ldots, n \end{aligned}$$

- Each θ_i is a latent parameter modelling x_i, while G is the unknown distribution over parameters modelled using a DP.
- This is the basic DP mixture model.
- Implement a DP mixture model.

Dirichlet Process Mixture Models

- We model a data set $x_1, \ldots, x_n$ using the following model:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \theta_i | G &\sim G \\ x_i | \theta_i &\sim F(\cdot|\theta_i) \qquad\qquad \text{for } i = 1, \ldots, n \end{aligned}$$

- Each θ_i is a latent parameter modelling x_i, while G is the unknown distribution over parameters modelled using a DP.
- This is the basic DP mixture model.
- Implement a DP mixture model.

- Different representations lead to different inference algorithms for DP mixture models.
- The most common are based on the Chinese restaurant process and on the stick-breaking construction.
- Here we shall work with the Chinese restaurant process representation, which, incidentally, can also be derived as the infinite limit of finite mixture models.
- A finite mixture model is defined as follows:

$$
\begin{aligned}
\theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\
\pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) && \\
z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\
x_i|\theta_{z_i}^* &\sim F(\cdot|\theta_{z_i}^*) &&
\end{aligned}
$$

- Different representations lead to different inference algorithms for DP mixture models.
- The most common are based on the Chinese restaurant process and on the stick-breaking construction.
- Here we shall work with the Chinese restaurant process representation, which, incidentally, can also be derived as the infinite limit of finite mixture models.
- A finite mixture model is defined as follows:

$$\begin{aligned}
\theta_k^* &\sim H & \text{for } k = 1, \ldots, K \\
\pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \\
z_i|\pi &\sim \text{Discrete}(\pi) & \text{for } i = 1, \ldots, n \\
x_i|\theta_{z_i}^* &\sim F(\cdot|\theta_{z_i}^*)
\end{aligned}$$

- Different representations lead to different inference algorithms for DP mixture models.
- The most common are based on the Chinese restaurant process and on the stick-breaking construction.
- Here we shall work with the Chinese restaurant process representation, which, incidentally, can also be derived as the infinite limit of finite mixture models.
- A finite mixture model is defined as follows:

$$\begin{aligned} \theta_k^* &\sim H & \text{for } k = 1, \ldots, K \\ \pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) & \\ z_i|\pi &\sim \text{Discrete}(\pi) & \text{for } i = 1, \ldots, n \\ x_i|\theta_{z_i}^* &\sim F(\cdot|\theta_{z_i}^*) & \end{aligned}$$

- Different representations lead to different inference algorithms for DP mixture models.
- The most common are based on the Chinese restaurant process and on the stick-breaking construction.
- Here we shall work with the Chinese restaurant process representation, which, incidentally, can also be derived as the infinite limit of finite mixture models.
- A finite mixture model is defined as follows:

$$
\begin{aligned}
\theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\
\pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \\
z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\
x_i|\theta_{z_i}^* &\sim F(\cdot|\theta_{z_i}^*)
\end{aligned}
$$

- A finite mixture model is defined as follows:

$$\begin{aligned} \theta_k^* &\sim H & \text{for } k = 1, \ldots, K \\ \pi &\sim \mathrm{Dirichlet}(\alpha/K, \ldots, \alpha/K) & \\ z_i|\pi &\sim \mathrm{Discrete}(\pi) & \text{for } i = 1, \ldots, n \\ x_i|\theta_{z_i}^* &\sim F(\theta_{z_i}^*) & \end{aligned}$$

- Assuming H is conjugate to $F(\cdot|\theta)$, we can integrate out both π and θ_k^*'s, leaving us with z_i's only.
- The simplest MCMC algorithm is to Gibbs sample z_i's (collapsed Gibbs sampling):

$$p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}) \propto p(z_i = k|\mathbf{z}_{\neg i})p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i})$$

$$p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) = \int p(x_i|\theta_k^*)p(\theta_k^*|\{x_j : j \neq i, z_j = k\})\, d\theta_k^*$$

- A finite mixture model is defined as follows:

$$
\begin{aligned}
\theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\
\pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) && \\
z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\
x_i|\theta_{z_i}^* &\sim F(\theta_{z_i}^*) &&
\end{aligned}
$$

- Assuming H is conjugate to $F(\cdot|\theta)$, we can integrate out both π and θ_k^*'s, leaving us with z_i's only.
- The simplest MCMC algorithm is to Gibbs sample z_i's (collapsed Gibbs sampling):

$$
p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}) \propto p(z_i = k|\mathbf{z}_{\neg i})p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i})
$$

$$
p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) = \int p(x_i|\theta_k^*)p(\theta_k^*|\{x_j : j \neq i, z_j = k\})\, d\theta_k^*
$$

- A finite mixture model is defined as follows:

$$
\begin{aligned}
\theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\
\pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \\
z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\
x_i|\theta_{z_i}^* &\sim F(\theta_{z_i}^*)
\end{aligned}
$$

- Assuming H is conjugate to $F(\cdot|\theta)$, we can integrate out both π and θ_k^*'s, leaving us with z_i's only.
- The simplest MCMC algorithm is to Gibbs sample z_i's (collapsed Gibbs sampling):

$$
p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}) \propto p(z_i = k|\mathbf{z}_{\neg i}) p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i})
$$

$$
p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) = \int p(x_i|\theta_k^*) p(\theta_k^*|\{x_j : j \neq i, z_j = k\})\, d\theta_k^*
$$

Aside: Markov Chain Monte Carlo Sampling

- Markov chain Monte Carlo sampling is a dominant and diverse family of inference algorithms for probabilistic models. Here we are interested in obtaining samples from the posterior:

$$\mathbf{z}^{(s)} \sim p(\mathbf{z}|\mathbf{x}) = \int p(\mathbf{z}, \theta^*, \pi|\mathbf{x})\, d\theta^* d\pi$$

- The basic idea is to construct a sequence $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots$ so that for large enough t, $\mathbf{z}^{(t)}$ will be an (approximate) sample from the posterior $p(\mathbf{z}|\mathbf{x})$.
- Convergence to the posterior is guaranteed, but (most of the time) there is no convergence diagnostics, only heuristics. Won't worry about this.
- Given the previous state $\mathbf{z}^{(t-1)}$, we construct $\mathbf{z}^{(t)}$ by making a small (stochastic) alteration to $\mathbf{z}^{(t-1)}$ so that $\mathbf{z}^{(t)}$ is "closer" to the posterior.
- In Gibbs sampling, this alteration is achieved by taking an entry, say z_i, and sampling it from the conditional:

$$z_i^{(t)} \sim p(z_i|\mathbf{z}_{\neg i}^{(t-1)}, \mathbf{x}) \qquad \mathbf{z}_{\neg i}^{(t)} = \mathbf{z}_{\neg i}^{(t-1)}$$

[Neal 1993]

Aside: Markov Chain Monte Carlo Sampling

- Markov chain Monte Carlo sampling is a dominant and diverse family of inference algorithms for probabilistic models. Here we are interested in obtaining samples from the posterior:

$$\mathbf{z}^{(s)} \sim p(\mathbf{z}|\mathbf{x}) = \int p(\mathbf{z}, \theta^*, \pi|\mathbf{x})\, d\theta^* d\pi$$

- The basic idea is to construct a sequence $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots$ so that for large enough t, $\mathbf{z}^{(t)}$ will be an (approximate) sample from the posterior $p(\mathbf{z}|\mathbf{x})$.
- Convergence to the posterior is guaranteed, but (most of the time) there is no convergence diagnostics, only heuristics. Won't worry about this.
- Given the previous state $\mathbf{z}^{(t-1)}$, we construct $\mathbf{z}^{(t)}$ by making a small (stochastic) alteration to $\mathbf{z}^{(t-1)}$ so that $\mathbf{z}^{(t)}$ is "closer" to the posterior.
- In Gibbs sampling, this alteration is achieved by taking an entry, say z_i, and sampling it from the conditional:

$$z_i^{(t)} \sim p(z_i|\mathbf{z}_{\neg i}^{(t-1)}, \mathbf{x}) \qquad \mathbf{z}_{\neg i}^{(t)} = \mathbf{z}_{\neg i}^{(t-1)}$$

[Neal 1993]

Aside: Markov Chain Monte Carlo Sampling

- Markov chain Monte Carlo sampling is a dominant and diverse family of inference algorithms for probabilistic models. Here we are interested in obtaining samples from the posterior:

$$\mathbf{z}^{(s)} \sim p(\mathbf{z}|\mathbf{x}) = \int p(\mathbf{z}, \theta^*, \pi|\mathbf{x})\, d\theta^* d\pi$$

- The basic idea is to construct a sequence $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots$ so that for large enough t, $\mathbf{z}^{(t)}$ will be an (approximate) sample from the posterior $p(\mathbf{z}|\mathbf{x})$.
- Convergence to the posterior is guaranteed, but (most of the time) there is no convergence diagnostics, only heuristics. Won't worry about this.
- Given the previous state $\mathbf{z}^{(t-1)}$, we construct $\mathbf{z}^{(t)}$ by making a small (stochastic) alteration to $\mathbf{z}^{(t-1)}$ so that $\mathbf{z}^{(t)}$ is "closer" to the posterior.
- In Gibbs sampling, this alteration is achieved by taking an entry, say z_i, and sampling it from the conditional:

$$z_i^{(t)} \sim p(z_i|\mathbf{z}_{\neg i}^{(t-1)}, \mathbf{x}) \qquad \mathbf{z}_{\neg i}^{(t)} = \mathbf{z}_{\neg i}^{(t-1)}$$

[Neal 1993]

Aside: Markov Chain Monte Carlo Sampling

- Markov chain Monte Carlo sampling is a dominant and diverse family of inference algorithms for probabilistic models. Here we are interested in obtaining samples from the posterior:

$$\mathbf{z}^{(s)} \sim p(\mathbf{z}|\mathbf{x}) = \int p(\mathbf{z}, \theta^*, \pi|\mathbf{x})\, d\theta^*\, d\pi$$

- The basic idea is to construct a sequence $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots$ so that for large enough t, $\mathbf{z}^{(t)}$ will be an (approximate) sample from the posterior $p(\mathbf{z}|\mathbf{x})$.
- Convergence to the posterior is guaranteed, but (most of the time) there is no convergence diagnostics, only heuristics. Won't worry about this.
- Given the previous state $\mathbf{z}^{(t-1)}$, we construct $\mathbf{z}^{(t)}$ by making a small (stochastic) alteration to $\mathbf{z}^{(t-1)}$ so that $\mathbf{z}^{(t)}$ is "closer" to the posterior.
- In Gibbs sampling, this alteration is achieved by taking an entry, say z_i, and sampling it from the conditional:

$$z_i^{(t)} \sim p(z_i|\mathbf{z}_{\neg i}^{(t-1)}, \mathbf{x}) \qquad \mathbf{z}_{\neg i}^{(t)} = \mathbf{z}_{\neg i}^{(t-1)}$$

[Neal 1993]

Aside: Markov Chain Monte Carlo Sampling

- Markov chain Monte Carlo sampling is a dominant and diverse family of inference algorithms for probabilistic models. Here we are interested in obtaining samples from the posterior:

$$\mathbf{z}^{(s)} \sim p(\mathbf{z}|\mathbf{x}) = \int p(\mathbf{z}, \theta^*, \pi|\mathbf{x})\, d\theta^*\, d\pi$$

- The basic idea is to construct a sequence $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots$ so that for large enough t, $\mathbf{z}^{(t)}$ will be an (approximate) sample from the posterior $p(\mathbf{z}|\mathbf{x})$.
- Convergence to the posterior is guaranteed, but (most of the time) there is no convergence diagnostics, only heuristics. Won't worry about this.
- Given the previous state $\mathbf{z}^{(t-1)}$, we construct $\mathbf{z}^{(t)}$ by making a small (stochastic) alteration to $\mathbf{z}^{(t-1)}$ so that $\mathbf{z}^{(t)}$ is "closer" to the posterior.
- In Gibbs sampling, this alteration is achieved by taking an entry, say z_i, and sampling it from the conditional:

$$z_i^{(t)} \sim p(z_i|\mathbf{z}_{\neg i}^{(t-1)}, \mathbf{x}) \qquad\qquad \mathbf{z}_{\neg i}^{(t)} = \mathbf{z}_{\neg i}^{(t-1)}$$

[Neal 1993]

Aside: Exponential Families

- An exponential family of distributions is parametrized as:

$$\begin{aligned} p(x|\theta) &= \exp\left(t(\theta)^\top s(x) - \phi(x) - \psi(\theta)\right) \\ s(x) &= \text{sufficient statistics vector.} \\ t(\theta) &= \text{natural parameter vector.} \\ \psi(\theta) &= \log \sum_{x'} \exp\left(t(\theta)^\top s(x') - \phi(x')\right) \ \text{(log normalization)} \end{aligned}$$

- The conjugate prior is an exponential family distribution over θ:

$$p(\theta) = \exp\left(t(\theta)^\top \nu - \eta\psi(\theta) - \xi(\nu, \eta)\right)$$

- The posterior given observations $x_1, \ldots, x_n$ is in the same family:

$$p(\theta|\mathbf{x}) = \exp\left(t(\theta)^\top (\nu + \textstyle\sum_i s(x_i)) - (\eta + n)\psi(\theta) - \xi(\nu + \sum_i s(x_i), \eta + n)\right)$$

- The marginal probability is:

$$p(\mathbf{x}) = \exp\left(\xi(\nu + \textstyle\sum_i s(x_i), \eta + n) - \xi(\nu, \eta) - \sum_i \phi(x_i)\right)$$

Aside: Exponential Families

- An exponential family of distributions is parametrized as:

$$
\begin{aligned}
p(x|\theta) &= \exp\left(t(\theta)^\top s(x) - \phi(x) - \psi(\theta)\right) \\
s(x) &= \text{sufficient statistics vector.} \\
t(\theta) &= \text{natural parameter vector.} \\
\psi(\theta) &= \log \sum_{x'} \exp\left(t(\theta)^\top s(x') - \phi(x')\right) \quad \text{(log normalization)}
\end{aligned}
$$

- The conjugate prior is an exponential family distribution over θ:

$$
p(\theta) = \exp\left(t(\theta)^\top \nu - \eta\psi(\theta) - \xi(\nu, \eta)\right)
$$

- The posterior given observations $x_1, \ldots, x_n$ is in the same family:

$$
p(\theta|\mathbf{x}) = \exp\left(t(\theta)^\top \left(\nu + \textstyle\sum_i s(x_i)\right) - (\eta + n)\psi(\theta) - \xi(\nu + \textstyle\sum_i s(x_i), \eta + n)\right)
$$

- The marginal probability is:

$$
p(\mathbf{x}) = \exp\left(\xi(\nu + \textstyle\sum_i s(x_i), \eta + n) - \xi(\nu, \eta) - \textstyle\sum_i \phi(x_i)\right)
$$

Aside: Exponential Families

- An exponential family of distributions is parametrized as:

$$
\begin{aligned}
p(x|\theta) &= \exp\left(t(\theta)^\top s(x) - \phi(x) - \psi(\theta)\right) \\
s(x) &= \text{sufficient statistics vector.} \\
t(\theta) &= \text{natural parameter vector.} \\
\psi(\theta) &= \log \sum_{x'} \exp\left(t(\theta)^\top s(x') - \phi(x')\right) \quad \text{(log normalization)}
\end{aligned}
$$

- The conjugate prior is an exponential family distribution over θ:

$$p(\theta) = \exp\left(t(\theta)^\top \nu - \eta\psi(\theta) - \xi(\nu,\eta)\right)$$

- The posterior given observations $x_1,\ldots,x_n$ is in the same family:

$$p(\theta|\mathbf{x}) = \exp\left(t(\theta)^\top \left(\nu + \textstyle\sum_i s(x_i)\right) - (\eta+n)\psi(\theta) - \xi(\nu + \textstyle\sum_i s(x_i), \eta+n)\right)$$

- The marginal probability is:

$$p(\mathbf{x}) = \exp\left(\xi(\nu + \textstyle\sum_i s(x_i), \eta+n) - \xi(\nu,\eta) - \textstyle\sum_i \phi(x_i)\right)$$

Aside: Exponential Families

- An exponential family of distributions is parametrized as:

$$\begin{aligned} p(x|\theta) &= \exp\left(t(\theta)^\top s(x) - \phi(x) - \psi(\theta)\right) \\ s(x) &= \text{sufficient statistics vector.} \\ t(\theta) &= \text{natural parameter vector.} \\ \psi(\theta) &= \log \sum_{x'} \exp\left(t(\theta)^\top s(x') - \phi(x')\right) \quad \text{(log normalization)} \end{aligned}$$

- The conjugate prior is an exponential family distribution over θ:

$$p(\theta) = \exp\left(t(\theta)^\top \nu - \eta\psi(\theta) - \xi(\nu, \eta)\right)$$

- The posterior given observations $x_1, \ldots, x_n$ is in the same family:

$$p(\theta|\mathbf{x}) = \exp\left(t(\theta)^\top \left(\nu + \textstyle\sum_i s(x_i)\right) - (\eta + n)\psi(\theta) - \xi(\nu + \textstyle\sum_i s(x_i), \eta + n)\right)$$

- The marginal probability is:

$$p(\mathbf{x}) = \exp\left(\xi(\nu + \textstyle\sum_i s(x_i), \eta + n) - \xi(\nu, \eta) - \textstyle\sum_i \phi(x_i)\right)$$

Dirichlet Process Mixture Models

Back to Collapsed Gibbs Sampling in Finite Mixture Models

- Finite mixture model:

$$\begin{aligned} \theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\ \pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \\ z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\ x_i|\theta_{z_i}^* &\sim F(\theta_{z_i}^*) \end{aligned}$$

- Integrating out both π and θ_k^*'s, the Gibbs sampling conditional distributions for $\mathbf{z}$ are:

$$\begin{aligned} p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}) &\propto p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}^{\neg i})p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}^{\neg i}) &\propto (n_k^{\neg i} + \alpha/K) \\ p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) &= \int p(x_i|\theta_k^*)p(\theta_k^*|\{x_j : j \neq i, z_j = k\})\, d\theta_k^* \\ &= \exp\Big(\xi(\nu + s(x_i) + \textstyle\sum_{j\neq i:z_j=k} s(x_j), \eta + 1 + n_k^{\neg i}) \\ &\quad - \xi(\nu + \textstyle\sum_{j\neq i:z_j=k} s(x_j), \eta + n_k^{\neg i}) - \phi(x_i)\Big) \end{aligned}$$

- Demo: fm_demo2d

- Finite mixture model:

$$\begin{aligned} \theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\ \pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \\ z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\ x_i|\theta_{z_i}^* &\sim F(\theta_{z_i}^*) \end{aligned}$$

- Integrating out both π and θ_k^*'s, the Gibbs sampling conditional distributions for **z** are:

$$\begin{aligned} p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}) &\propto p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}^{\neg i})p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}^{\neg i}) &\propto (n_k^{\neg i} + \alpha/K) \\ p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) &= \textstyle\int p(x_i|\theta_k^*)p(\theta_k^*|\{x_j : j \neq i, z_j = k\})\, d\theta_k^* \\ &= \exp\big(\xi(\nu + s(x_i) + \textstyle\sum_{j\neq i: z_j = k} s(x_j), \eta + 1 + n_k^{\neg i}) \\ &\qquad - \xi(\nu + \textstyle\sum_{j\neq i: z_j = k} s(x_j), \eta + n_k^{\neg i}) - \phi(x_i)\big) \end{aligned}$$

- Demo: fm_demo2d

- Finite mixture model:

$$\begin{aligned} \theta_k^* &\sim H && \text{for } k = 1, \ldots, K \\ \pi &\sim \text{Dirichlet}(\alpha/K, \ldots, \alpha/K) \\ z_i|\pi &\sim \text{Discrete}(\pi) && \text{for } i = 1, \ldots, n \\ x_i|\theta_{z_i}^* &\sim F(\theta_{z_i}^*) \end{aligned}$$

- Integrating out both π and θ_k^*'s, the Gibbs sampling conditional distributions for **z** are:

$$\begin{aligned} p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}) &\propto p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}^{\neg i})p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ p(z_i = k|\mathbf{z}^{\neg i}, \mathbf{x}^{\neg i}) &\propto (n_k^{\neg i} + \alpha/K) \\ p(x_i|\mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) &= \int p(x_i|\theta_k^*)p(\theta_k^*|\{x_j : j \neq i, z_j = k\})\, d\theta_k^* \\ &= \exp\Big(\xi(\nu + s(x_i) + \textstyle\sum_{j\neq i: z_j=k} s(x_j), \eta + 1 + n_k^{\neg i}) \\ &\qquad - \xi(\nu + \textstyle\sum_{j\neq i: z_j=k} s(x_j), \eta + n_k^{\neg i}) - \phi(x_i)\Big) \end{aligned}$$

- Demo: fm_demo2d

Dirichlet Process Mixture Models

Taking the Infinite Limit

- Imagine that $K \gg 0$ is really large.
- Only a few components will be "active" (i.e. with $n_k > 0$), while most are "inactive".

$$p(z_i = k | \mathbf{z}^{\neg i}, \mathbf{x}) \propto \begin{cases} (n_k^{\neg i} + \alpha/K) p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) & \text{if } k \text{ active;} \\ (\alpha/K) p(x_i) & \text{if } k \text{ inactive.} \end{cases}$$

$$\begin{aligned} p(z_i = k \text{ active} | \mathbf{z}^{\neg i}, \mathbf{x}) &\propto (n_k^{\neg i} + \alpha/K) p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ &\approx n_k^{\neg i} p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ p(z_i \text{ inactive} | \mathbf{z}^{\neg i}, \mathbf{x}) &\propto (\alpha(K - K_{\text{active}})/K) p(x_i) \\ &\approx \alpha p(x_i) \end{aligned}$$

- This gives an inference algorithm for DP mixture models in Chinese restaurant process representation.

- Imagine that $K \gg 0$ is really large.
- Only a few components will be "active" (i.e. with $n_k > 0$), while most are "inactive".

$$p(z_i = k | \mathbf{z}^{\neg i}, \mathbf{x}) \propto \begin{cases} (n_k^{\neg i} + \alpha/K) p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) & \text{if } k \text{ active;} \\ (\alpha/K) p(x_i) & \text{if } k \text{ inactive.} \end{cases}$$

$$\begin{aligned} p(z_i = k \text{ active} | \mathbf{z}^{\neg i}, \mathbf{x}) &\propto (n_k^{\neg i} + \alpha/K) p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ &\approx n_k^{\neg i} p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i}) \\ p(z_i \text{ inactive} | \mathbf{z}^{\neg i}, \mathbf{x}) &\propto (\alpha(K - K_{\text{active}})/K) p(x_i) \\ &\approx \alpha p(x_i) \end{aligned}$$

- This gives an inference algorithm for DP mixture models in Chinese restaurant process representation.

- Rearrange mixture component indices so that $1, \ldots, K_{\text{active}}$ are active, and the rest are inactive.

$$p(z_i = k \leq K_{\text{active}} | \mathbf{z}^{\neg i}, \mathbf{x}) \propto n_k^{\neg i} p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i})$$
$$p(z_i > K_{\text{active}} | \mathbf{z}^{\neg i}, \mathbf{x}) \propto \alpha p(x_i)$$

- If z_i takes on an inactive value, instantiate a new active component, and increment K_{active}.
- If $n_k = 0$ for some k during sampling, delete that active component, and decrement K_{active}.

- Rearrange mixture component indices so that $1, \ldots, K_{\text{active}}$ are active, and the rest are inactive.

$$p(z_i = k \leq K_{\text{active}} | \mathbf{z}^{\neg i}, \mathbf{x}) \propto n_k^{\neg i} p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i})$$
$$p(z_i > K_{\text{active}} | \mathbf{z}^{\neg i}, \mathbf{x}) \propto \alpha p(x_i)$$

- If z_i takes on an inactive value, instantiate a new active component, and increment K_{active}.
- If $n_k = 0$ for some k during sampling, delete that active component, and decrement K_{active}.

- Rearrange mixture component indices so that $1, \ldots, K_{\text{active}}$ are active, and the rest are inactive.

$$p(z_i = k \leq K_{\text{active}} | \mathbf{z}^{\neg i}, \mathbf{x}) \propto n_k^{\neg i} p(x_i | \mathbf{z}^{\neg i}, \mathbf{x}_k^{\neg i})$$
$$p(z_i > K_{\text{active}} | \mathbf{z}^{\neg i}, \mathbf{x}) \propto \alpha p(x_i)$$

- If z_i takes on an inactive value, instantiate a new active component, and increment K_{active}.
- If $n_k = 0$ for some k during sampling, delete that active component, and decrement K_{active}.

Clustering NIPS Papers

- I have prepared a small subset of NIPS papers for you to try clustering them.
- We concentrate on a small subset of papers, and a small subset of "informative" words.
- Each paper is represented as a bag-of-words. Paper i is represented by a vector $\mathbf{x}_i = (x_{i1}, \ldots, x_{iW})$:

$$x_{iw} = c \qquad \text{if word } w \text{ occurs } c \text{ times in paper } i.$$

- Model papers in cluster k using a Multinomial distribution:

$$p(\mathbf{x}_i|\theta^*_k) = \frac{(\sum_w x_{iw})!}{\prod_w x_{iw}!} \prod_w (\theta^*_{kw})^{x_{iw}}$$

- The conjugate prior for θ^*_k is a Dirichlet:

$$p(\theta^*_k|b) = \frac{\Gamma(\sum_w b_w)}{\prod_w \Gamma(b_w)} \prod_w (\theta^*_{kw})^{b_w-1}$$

Clustering NIPS Papers

- I have prepared a small subset of NIPS papers for you to try clustering them.
- We concentrate on a small subset of papers, and a small subset of "informative" words.
- Each paper is represented as a bag-of-words. Paper i is represented by a vector $\mathbf{x}_i = (x_{i1}, \ldots, x_{iW})$:

$$x_{iw} = c \qquad \text{if word } w \text{ occurs } c \text{ times in paper } i.$$

- Model papers in cluster k using a Multinomial distribution:

$$p(\mathbf{x}_i|\theta_k^*) = \frac{(\sum_w x_{iw})!}{\prod_w x_{iw}!} \prod_w (\theta_{kw}^*)^{x_{iw}}$$

- The conjugate prior for θ_k^* is a Dirichlet:

$$p(\theta_k^*|b) = \frac{\Gamma(\sum_w b_w)}{\prod_w \Gamma(b_w)} \prod_w (\theta_{kw}^*)^{b_w - 1}$$

Clustering NIPS Papers

- I have prepared a small subset of NIPS papers for you to try clustering them.
- We concentrate on a small subset of papers, and a small subset of "informative" words.
- Each paper is represented as a bag-of-words. Paper i is represented by a vector $\mathbf{x}_i = (x_{i1}, \ldots, x_{iW})$:

$$x_{iw} = c \qquad \text{if word } w \text{ occurs } c \text{ times in paper } i.$$

- Model papers in cluster k using a Multinomial distribution:

$$p(\mathbf{x}_i|\theta_k^*) = \frac{(\sum_w x_{iw})!}{\prod_w x_{iw}!} \prod_w (\theta_{kw}^*)^{x_{iw}}$$

- The conjugate prior for θ_k^* is a Dirichlet:

$$p(\theta_k^*|b) = \frac{\Gamma(\sum_w b_w)}{\prod_w \Gamma(b_w)} \prod_w (\theta_{kw}^*)^{b_w - 1}$$

Clustering NIPS Papers

- I have prepared a small subset of NIPS papers for you to try clustering them.
- We concentrate on a small subset of papers, and a small subset of "informative" words.
- Each paper is represented as a bag-of-words. Paper i is represented by a vector $\mathbf{x}_i = (x_{i1}, \dots, x_{iW})$:

$$x_{iw} = c \qquad \text{if word } w \text{ occurs } c \text{ times in paper } i.$$

- Model papers in cluster k using a Multinomial distribution:

$$p(\mathbf{x}_i|\theta^*_k) = \frac{(\sum_w x_{iw})!}{\prod_w x_{iw}!} \prod_w (\theta^*_{kw})^{x_{iw}}$$

- The conjugate prior for θ^*_k is a Dirichlet:

$$p(\theta^*_k|b) = \frac{\Gamma(\sum_w b_w)}{\prod_w \Gamma(b_w)} \prod_w (\theta^*_{kw})^{b_w - 1}$$

Clustering NIPS Papers

- I have prepared a small subset of NIPS papers for you to try clustering them.
- We concentrate on a small subset of papers, and a small subset of "informative" words.
- Each paper is represented as a bag-of-words. Paper i is represented by a vector $\mathbf{x}_i = (x_{i1}, \ldots, x_{iW})$:

$$x_{iw} = c \qquad \text{if word } w \text{ occurs } c \text{ times in paper } i.$$

- Model papers in cluster k using a Multinomial distribution:

$$p(\mathbf{x}_i|\theta^*_k) = \frac{(\sum_w x_{iw})!}{\prod_w x_{iw}!} \prod_w (\theta^*_{kw})^{x_{iw}}$$

- The conjugate prior for θ^*_k is a Dirichlet:

$$p(\theta^*_k|b) = \frac{\Gamma(\sum_w b_w)}{\prod_w \Gamma(b_w)} \prod_w (\theta^*_{kw})^{b_w - 1}$$

Clustering NIPS Papers

Specifying the Priors

- We shall use a symmetric Dirichlet prior for the cluster parameters θ. Specifically $b_w = b/W$ for some $b > 0$.
- The model:

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

- Only two numbers to set: α and b.
- α controls the a priori expected number of clusters.
- b controls the number of words assigned to each cluster.
- What are reasonable values for α and b?

Clustering NIPS Papers

Specifying the Priors

- We shall use a symmetric Dirichlet prior for the cluster parameters θ. Specifically $b_w = b/W$ for some $b > 0$.
- The model:

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

- Only two numbers to set: α and b.
- α controls the a priori expected number of clusters.
- b controls the number of words assigned to each cluster.
- What are reasonable values for α and b?

Clustering NIPS Papers

Specifying the Priors

- We shall use a symmetric Dirichlet prior for the cluster parameters θ. Specifically $b_w = b/W$ for some $b > 0$.
- The model:

$$
\begin{aligned}
H &= \mathrm{Dirichlet}(b/W, \dots, b/W) \\
G &\sim \mathrm{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \mathrm{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

- Only two numbers to set: α and b.
- α controls the a priori expected number of clusters.
- b controls the number of words assigned to each cluster.
- What are reasonable values for α and b?

Clustering NIPS Papers

Specifying the Priors

- We shall use a symmetric Dirichlet prior for the cluster parameters θ. Specifically $b_w = b/W$ for some $b > 0$.
- The model:

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

- Only two numbers to set: α and b.
- α controls the a priori expected number of clusters.
- b controls the number of words assigned to each cluster.
- What are reasonable values for α and b?

Clustering NIPS Papers

Sensitivity to Priors

- When building models and making inferences, and one does not “trust” ones prior very much, then it is important to perform sensitivity analysis.
- Sensitivity analysis is about determining how much our inference conclusions depend on the setting of the model priors.
- If our conclusions depend strongly on the priors which we don’t trust very much, then we cannot trust our conclusions either.
- If our conclusions do not depend strongly on the priors, then we can more strongly trust our conclusions.
- What part of our model should we worry about?

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

Clustering NIPS Papers

Sensitivity to Priors

- When building models and making inferences, and one does not “trust” ones prior very much, then it is important to perform sensitivity analysis.
- Sensitivity analysis is about determining how much our inference conclusions depend on the setting of the model priors.
- If our conclusions depend strongly on the priors which we don’t trust very much, then we cannot trust our conclusions either.
- If our conclusions do not depend strongly on the priors, then we can more strongly trust our conclusions.
- What part of our model should we worry about?

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

Clustering NIPS Papers

Sensitivity to Priors

- When building models and making inferences, and one does not "trust" ones prior very much, then it is important to perform sensitivity analysis.
- Sensitivity analysis is about determining how much our inference conclusions depend on the setting of the model priors.
- If our conclusions depend strongly on the priors which we don't trust very much, then we cannot trust our conclusions either.
- If our conclusions do not depend strongly on the priors, then we can more strongly trust our conclusions.
- What part of our model should we worry about?

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

Clustering NIPS Papers

Sensitivity to Priors

- When building models and making inferences, and one does not "trust" ones prior very much, then it is important to perform sensitivity analysis.
- Sensitivity analysis is about determining how much our inference conclusions depend on the setting of the model priors.
- If our conclusions depend strongly on the priors which we don't trust very much, then we cannot trust our conclusions either.
- If our conclusions do not depend strongly on the priors, then we can more strongly trust our conclusions.
- What part of our model should we worry about?

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

Clustering NIPS Papers

Sensitivity to Priors

- When building models and making inferences, and one does not “trust” ones prior very much, then it is important to perform sensitivity analysis.
- Sensitivity analysis is about determining how much our inference conclusions depend on the setting of the model priors.
- If our conclusions depend strongly on the priors which we don’t trust very much, then we cannot trust our conclusions either.
- If our conclusions do not depend strongly on the priors, then we can more strongly trust our conclusions.
- What part of our model should we worry about?

$$
\begin{aligned}
H &= \text{Dirichlet}(b/W, \ldots, b/W) \\
G &\sim \text{DP}(\alpha, H) \\
\theta_i &\sim G \\
\mathbf{x}_i &\sim \text{Multinomial}(n_i, \theta_i)
\end{aligned}
$$

Summary

- We explored some properties of the Dirichlet process.
- We implemented a Dirichlet process mixture model.
- We applied a Dirichlet process mixture model to clustering NIPS papers.
- We considered ways of specifying the hyperparameters of the model, and explored the sensitivity to these hyperparameters.

- Dirichlet processes are not that mysterious or hard.

Summary

- We explored some properties of the Dirichlet process.
- We implemented a Dirichlet process mixture model.
- We applied a Dirichlet process mixture model to clustering NIPS papers.
- We considered ways of specifying the hyperparameters of the model, and explored the sensitivity to these hyperparameters.

- Dirichlet processes are not that mysterious or hard.

Dirichlet Processes were first introduced by [Ferguson 1973], while [Antoniak 1974] further developed DPs as well as introduce the mixture of DPs. [Blackwell and MacQueen 1973] showed that the Blackwell-MacQueen urn scheme is exchangeable with the DP being its de Finetti measure. Further information on the Chinese restaurant process can be obtained at [Aldous 1985, Pitman 2002]. The DP is also related to Ewens' Sampling Formula [Ewens 1972]. [Sethuraman 1994] gave a constructive definition of the DP via a stick-breaking construction. DPs were rediscovered in the machine learning community by [**?**, Rasmussen 2000].

Hierarchical Dirichlet Processes (HDPs) were first developed by [Teh et al. 2006], although an aspect of the model was first discussed in the context of infinite hidden Markov models [Beal et al. 2002]. HDPs and generalizations have been applied across a wide variety of fields.

Dependent Dirichlet Processes are sets of coupled distributions over probability measures, each of which is marginally DP [MacEachern et al. 2001]. A variety of dependent DPs have been proposed in the literature since then [Srebro and Roweis 2005, Griffin 2007, Caron et al. 2007]. The infinite mixture of Gaussian processes of [Rasmussen and Ghahramani 2002] can also be interpreted as a dependent DP.

Indian Buffet Processes (IBPs) were first proposed in [Griffiths and Ghahramani 2006], and extended to a two-parameter family in [Griffiths et al. 2007b]. [Thibaux and Jordan 2007] showed that the de Finetti measure for the IBP is the beta process of [Hjort 1990], while [Teh et al. 2007] gave a stick-breaking construction and developed efficient slice sampling inference algorithms for the IBP.

Nonparametric Tree Models are models that use distributions over trees that are consistent and exchangeable. [Blei et al. 2004] used a nested CRP to define distributions over trees with a finite number of levels. [Neal 2001, Neal 2003] defined Dirichlet diffusion trees, which are binary trees produced by a fragmentation process. [Teh et al. 2008] used Kingman's coalescent [Kingman 1982b, Kingman 1982a] to produce random binary trees using a coalescent process. [Roy et al. 2007] proposed annotated hierarchies, using tree-consistent partitions first defined in [Heller and Ghahramani 2005] to model both relational and featural data.

Markov chain Monte Carlo Inference algorithms are the dominant approaches to inference in DP mixtures. [Neal 2000] is a good review of algorithms based on Gibbs sampling in the CRP representation. Algorithm 8 in [Neal 2000] is still one of the best algorithms based on simple local moves. [Ishwaran and James 2001] proposed blocked Gibbs sampling in the stick-breaking representation instead due to the simplicity in implementation. This has been further explored in [Porteous et al. 2006]. Since

then there has been proposals for better MCMC samplers based on proposing larger moves in a Metropolis-Hastings framework [Jain and Neal 2004, Liang et al. 2007a], as well as sequential Monte Carlo [Fearnhead 2004, Mansingkha et al. 2007].
Other Approximate Inference Methods have also been proposed for DP mixture models. [Blei and Jordan 2006] is the first variational Bayesian approximation, and is based on a truncated stick-breaking representation. [Kurihara et al. 2007] proposed an improved VB approximation based on a better truncation technique, and using KD-trees for extremely efficient inference in large scale applications. [Kurihara et al. 2007] studied improved VB approximations based on integrating out the stick-breaking weights. [Minka and Ghahramani 2003] derived an expectation propagation based algorithm. [Heller and Ghahramani 2005] derived tree-based approximation which can be seen as a Bayesian hierarchical clustering algorithm. [Daume III 2007] developed admissible search heuristics to find MAP clusterings in a DP mixture model.

Computer Vision and Image Processing. HDPs have been used in object tracking [Fox et al. 2006, Fox et al. 2007b, Fox et al. 2007a]. An extension called the transformed Dirichlet process has been used in scene analysis [Sudderth et al. 2006b, Sudderth et al. 2006a, Sudderth et al. 2007], a related extension has been used in fMRI image analysis [Kim and Smyth 2007, Kim 2007]. An extension of the infinite hidden Markov model called the nonparametric hidden Markov tree has been introduced and applied to image denoising [Kivinen et al. 2007].
Natural Language Processing. HDPs are essential ingredients in defining nonparametric context free grammars [Liang et al. 2007b, Finkel et al. 2007]. [Johnson et al. 2007] defined adaptor grammars, which is a framework generalizing both probabilistic context free grammars as well as a variety of nonparametric models including DPs and HDPs. DPs and HDPs have been used in information retrieval [Cowans 2004], word segmentation [Goldwater et al. 2006b], word morphology modelling [Goldwater et al. 2006a], coreference resolution [Haghighi and Klein 2007], topic modelling [Blei et al. 2004, Teh et al. 2006, Li et al. 2007]. An extension of the HDP called the hierarchical Pitman-Yor process has been applied to language modelling [Teh 2006a, Teh 2006b, Goldwater et al. 2006a].[Savova et al. 2007] used annotated hierarchies to construct syntactic hierarchies. Theses on nonparametric methods in NLP include [Cowans 2006, Goldwater 2006].
Other Applications. Applications of DPs, HDPs and infinite HMMs in bioinformatics include [Xing et al. 2004, Xing et al. 2006, Xing et al. 2007, Xing and Sohn 2007a, Xing and Sohn 2007b]. DPs have been applied in relational learning [Shafto et al. 2006, Kemp et al. 2006, Xu et al. 2006], spike sorting [Wood et al. 2006a, Görür 2007]. The HDP has been used in a cognitive model of categorization [Griffiths et al. 2007a]. IBPs have been applied to infer hidden causes [Wood et al. 2006b], in a choice model [Görür et al. 2006], to modelling dyadic data [Meeds et al. 2007], to overlapping clustering [Heller and Ghahramani 2007], and to matrix factorization [Wood and Griffiths 2006].

References I

Aldous, D. (1985).
Exchangeability and related topics.
In *École d'Été de Probabilités de Saint-Flour XIII–1983*, pages 1–198. Springer, Berlin.

Antoniak, C. E. (1974).
Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems.
Annals of Statistics, 2(6):1152–1174.

Beal, M. J., Ghahramani, Z., and Rasmussen, C. E. (2002).
The infinite hidden Markov model.
In *Advances in Neural Information Processing Systems*, volume 14.

Blackwell, D. and MacQueen, J. B. (1973).
Ferguson distributions via Pólya urn schemes.
Annals of Statistics, 1:353–355.

Blei, D., Griffiths, T., Jordan, M., and Tenenbaum, J. (2004).
Hierarchical topic models and the nested Chinese restaurant process.
In *Advances in Neural Information Processing Systems*, volume 16.

Blei, D. M. and Jordan, M. I. (2006).
Variational inference for Dirichlet process mixtures.
Bayesian Analysis, 1(1):121–144.

Caron, F., Davy, M., and Doucet, A. (2007).
Generalized Polya urn for time-varying Dirichlet process mixtures.
In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, volume 23.

References II

Cowans, P. (2004).
Information retrieval using hierarchical Dirichlet processes.
In *Proceedings of the Annual International Conference on Research and Development in Information Retrieval*, volume 27, pages 564–565.

Cowans, P. (2006).
Probabilistic Document Modelling.
PhD thesis, University of Cambridge.

Daume III, H. (2007).
Fast search for Dirichlet process mixture models.
In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, volume 11.

Ewens, W. J. (1972).
The sampling theory of selectively neutral alleles.
Theoretical Population Biology, 3:87–112.

Fearnhead, P. (2004).
Particle filters for mixture models with an unknown number of components.
Statistics and Computing, 14:11–21.

Ferguson, T. S. (1973).
A Bayesian analysis of some nonparametric problems.
Annals of Statistics, 1(2):209–230.

Finkel, J. R., Grenager, T., and Manning, C. D. (2007).
The infinite tree.
In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

References III

Fox, E. B., Choi, D. S., and Willsky, A. S. (2006).
Nonparametric Bayesian methods for large scale multi-target tracking.
In *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, volume 40.

Fox, E. B., Sudderth, E. B., Choi, D. S., and Willsky, A. S. (2007a).
Tracking a non-cooperative maneuvering target using hierarchical Dirichlet processes.
In *Proceedings of the Adaptive Sensor Array Processing Conference*.

Fox, E. B., Sudderth, E. B., and Willsky, A. S. (2007b).
Hierarchical Dirichlet processes for tracking maneuvering targets.
In *Proceedings of the International Conference on Information Fusion*.

Goldwater, S. (2006).
Nonparametric Bayesian Models of Lexical Acquisition.
PhD thesis, Brown University.

Goldwater, S., Griffiths, T., and Johnson, M. (2006a).
Interpolating between types and tokens by estimating power-law generators.
In *Advances in Neural Information Processing Systems*, volume 18.

Goldwater, S., Griffiths, T. L., and Johnson, M. (2006b).
Contextual dependencies in unsupervised word segmentation.
In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*.

Görür, D. (2007).
Nonparametric Bayesian Discrete Latent Variable Models for Unsupervised Learning.
PhD thesis, Technischen Universität Berlin.

Görür, D., Jäkel, F., and Rasmussen, C. E. (2006).
A choice model with infinitely many latent features.
In *Proceedings of the International Conference on Machine Learning*, volume 23.

Griffin, J. E. (2007).
The Ornstein-Uhlenbeck Dirichlet process and other time-varying processes for Bayesian nonparametric inference.
Technical report, Department of Statistics, University of Warwick.

Griffiths, T. L., Canini, K. R., Sanborn, A. N., and Navarro, D. J. (2007a).
Unifying rational models of categorization via the hierarchical Dirichlet process.
In *Proceedings of the Annual Conference of the Cognitive Science Society*, volume 29.

Griffiths, T. L. and Ghahramani, Z. (2006).
Infinite latent feature models and the Indian buffet process.
In *Advances in Neural Information Processing Systems*, volume 18.

Griffiths, T. L., Ghahramani, Z., and Sollich, P. (2007b).
Bayesian nonparametric latent feature models (with discussion and rejoinder).
In *Bayesian Statistics*, volume 8.

Haghighi, A. and Klein, D. (2007).
Unsupervised coreference resolution in a nonparametric Bayesian model.
In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Heller, K. A. and Ghahramani, Z. (2005).
Bayesian hierarchical clustering.
In *Proceedings of the International Conference on Machine Learning*, volume 22.

References V

Heller, K. A. and Ghahramani, Z. (2007).
A nonparametric Bayesian approach to modeling overlapping clusters.
In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, volume 11.

Hjort, N. L. (1990).
Nonparametric Bayes estimators based on beta processes in models for life history data.
Annals of Statistics, 18(3):1259–1294.

Ishwaran, H. and James, L. F. (2001).
Gibbs sampling methods for stick-breaking priors.
Journal of the American Statistical Association, 96(453):161–173.

Jain, S. and Neal, R. M. (2004).
A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model.
Technical report, Department of Statistics, University of Toronto.

Johnson, M., Griffiths, T. L., and Goldwater, S. (2007).
Adaptor grammars: A framework for specifying compositional nonparametric Bayesian models.
In *Advances in Neural Information Processing Systems*, volume 19.

Kemp, C., Tenenbaum, J. B., Griffiths, T. L., Yamada, T., and Ueda, N. (2006).
Learning systems of concepts with an infinite relational model.
In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 21.

Kim, S. (2007).
Learning Hierarchical Probabilistic Models with Random Effects with Applications to Time-series and Image Data.
PhD thesis, Information and Computer Science, University of California at Irvine.

References VI

Kim, S. and Smyth, P. (2007).
Hierarchical dirichlet processes with random effects.
In *Advances in Neural Information Processing Systems*, volume 19.

Kingman, J. F. C. (1982a).
The coalescent.
Stochastic Processes and their Applications, 13:235–248.

Kingman, J. F. C. (1982b).
On the genealogy of large populations.
Journal of Applied Probability, 19:27–43.
Essays in Statistical Science.

Kivinen, J., Sudderth, E., and Jordan, M. I. (2007).
Image denoising with nonparametric hidden Markov trees.
In *International Conference on Image Processing*.

Kurihara, K., Welling, M., and Vlassis, N. (2007).
Accelerated variational DP mixture models.
In *Advances in Neural Information Processing Systems*, volume 19.

Li, W., Blei, D., and McCallum, A. (2007).
Nonparametric Bayes pachinko allocation.
In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*.

Liang, P., Jordan, M. I., and Taskar, B. (2007a).
A permutation-augmented sampler for Dirichlet process mixture models.
In *Proceedings of the International Conference on Machine Learning*.

References VII

Liang, P., Petrov, S., Jordan, M. I., and Klein, D. (2007b).
The infinite PCFG using hierarchical Dirichlet processes.
In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

MacEachern, S., Kottas, A., and Gelfand, A. (2001).
Spatial nonparametric Bayesian models.
Technical Report 01-10, Institute of Statistics and Decision Sciences, Duke University.
http://ftp.isds.duke.edu/WorkingPapers/01-10.html.

Mansingkha, V. K., Roy, D. M., Rifkin, R., and Tenenbaum, J. B. (2007).
AClass: An online algorithm for generative classification.
In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, volume 11.

Meeds, E., Ghahramani, Z., Neal, R. M., and Roweis, S. T. (2007).
Modeling dyadic data with binary latent factors.
In *Advances in Neural Information Processing Systems*, volume 19.

Minka, T. P. and Ghahramani, Z. (2003).
Expectation propagation for infinite mixtures.
Presented at NIPS2003 Workshop on Nonparametric Bayesian Methods and Infinite Models.

Neal, R. M. (1993).
Probabilistic inference using Markov chain Monte Carlo methods.
Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto.

Neal, R. M. (2000).
Markov chain sampling methods for Dirichlet process mixture models.
Journal of Computational and Graphical Statistics, 9:249–265.

References VIII

Neal, R. M. (2001).
Defining priors for distributions using Dirichlet diffusion trees.
Technical Report 0104, Department of Statistics, University of Toronto.

Neal, R. M. (2003).
Density modeling and clustering using Dirichlet diffusion trees.
In *Bayesian Statistics*, volume 7, pages 619–629.

Pitman, J. (2002).
Combinatorial stochastic processes.
Technical Report 621, Department of Statistics, University of California at Berkeley.
Lecture notes for St. Flour Summer School.

Porteous, I., Ihler, A., Smyth, P., and Welling, M. (2006).
Gibbs sampling for (Coupled) infinite mixture models in the stick-breaking representation.
In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, volume 22.

Rasmussen, C. E. (2000).
The infinite Gaussian mixture model.
In *Advances in Neural Information Processing Systems*, volume 12.

Rasmussen, C. E. and Ghahramani, Z. (2001).
Occam's razor.
In *Advances in Neural Information Processing Systems*, volume 13.

Rasmussen, C. E. and Ghahramani, Z. (2002).
Infinite mixtures of Gaussian process experts.
In *Advances in Neural Information Processing Systems*, volume 14.

References IX

Roy, D. M., Kemp, C., Mansinghka, V., and Tenenbaum, J. B. (2007).
Learning annotated hierarchies from relational data.
In *Advances in Neural Information Processing Systems*, volume 19.

Savova, V., Roy, D., Schmidt, L., and Tenenbaum, J. B. (2007).
Discovering syntactic hierarchies.
In *Proceedings of the Annual Conference of the Cognitive Science Society*, volume 29.

Sethuraman, J. (1994).
A constructive definition of Dirichlet priors.
Statistica Sinica, 4:639–650.

Shafto, P., Kemp, C., Mansinghka, V., Gordon, M., and Tenenbaum, J. B. (2006).
Learning cross-cutting systems of categories.
In *Proceedings of the Annual Conference of the Cognitive Science Society*, volume 28.

Srebro, N. and Roweis, S. (2005).
Time-varying topic models using dependent Dirichlet processes.
Technical Report UTML-TR-2005-003, Department of Computer Science, University of Toronto.

Sudderth, E., Torralba, A., Freeman, W., and Willsky, A. (2006a).
Depth from familiar objects: A hierarchical model for 3D scenes.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Sudderth, E., Torralba, A., Freeman, W., and Willsky, A. (2006b).
Describing visual scenes using transformed Dirichlet processes.
In *Advances in Neural Information Processing Systems*, volume 18.

References X

Sudderth, E., Torralba, A., Freeman, W., and Willsky, A. (2007).
Describing visual scenes using transformed objects and parts.
To appear in the International Journal of Computer Vision.

Teh, Y. W. (2006a).
A Bayesian interpretation of interpolated Kneser-Ney.
Technical Report TRA2/06, School of Computing, National University of Singapore.

Teh, Y. W. (2006b).
A hierarchical Bayesian language model based on Pitman-Yor processes.
In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992.

Teh, Y. W., Daume III, H., and Roy, D. M. (2008).
Bayesian agglomerative clustering with coalescents.
In *Advances in Neural Information Processing Systems*, volume 20.

Teh, Y. W., Görür, D., and Ghahramani, Z. (2007).
Stick-breaking construction for the Indian buffet process.
In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, volume 11.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006).
Hierarchical Dirichlet processes.
Journal of the American Statistical Association, 101(476):1566–1581.

Thibaux, R. and Jordan, M. I. (2007).
Hierarchical beta processes and the Indian buffet process.
In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, volume 11.

References XI

Wood, F., Goldwater, S., and Black, M. J. (2006a).
A non-parametric Bayesian approach to spike sorting.
In *Proceedings of the IEEE Conference on Engineering in Medicine and Biologicial Systems*, volume 28.

Wood, F. and Griffiths, T. L. (2006).
Particle filtering for nonparametric Bayesian matrix factorization.
In *Advances in Neural Information Processing Systems*, volume 18.

Wood, F., Griffiths, T. L., and Ghahramani, Z. (2006b).
A non-parametric Bayesian method for inferring hidden causes.
In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, volume 22.

Xing, E., Sharan, R., and Jordan, M. (2004).
Bayesian haplotype inference via the dirichlet process.
In *Proceedings of the International Conference on Machine Learning*, volume 21.

Xing, E. P., Jordan, M. I., and Roded, R. (2007).
Bayesian haplotype inference via the Dirichlet process.
Journal of Computational Biology, 14(3):267–284.

Xing, E. P. and Sohn, K. (2007a).
Hidden Markov Dirichlet process: Modeling genetic recombination in open ancestral space.
Bayesian Analysis, 2(2).

Xing, E. P. and Sohn, K. (2007b).
A nonparametric Bayesian approach for haplotype reconstruction from single and multi-population data.
Technical Report CMU-MLD 07-107, Carnegie Mellow University.

References XII

Xing, E. P., Sohn, K., Jordan, M. I., and Teh, Y. W. (2006).
Bayesian multi-population haplotype inference via a hierarchical Dirichlet process mixture.
In *Proceedings of the International Conference on Machine Learning*, volume 23.

Xu, Z., Tresp, V., Yu, K., and Kriegel, H.-P. (2006).
Infinite hidden relational models.
In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, volume 22.