Non-parametric Bayesian Methods

Uncertainty in Artificial Intelligence
Tutorial July 2005

Zoubin Ghahramani

Gatsby Computational Neuroscience Unit[1]
University College London, UK

Center for Automated Learning and Discovery
Carnegie Mellon University, USA

zoubin@gatsby.ucl.ac.uk
http://www.gatsby.ucl.ac.uk

[1]Starting Jan 2006: Department of Engineering
University of Cambridge, UK

Bayes Rule Applied to Machine Learning

$$P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)P(\theta)}{P(\mathcal{D})}$$

$P(\mathcal{D}\|\theta)$	likelihood of θ
$P(\theta)$	prior probability of θ
$P(\theta\|\mathcal{D})$	posterior of θ given $\mathcal{D}$

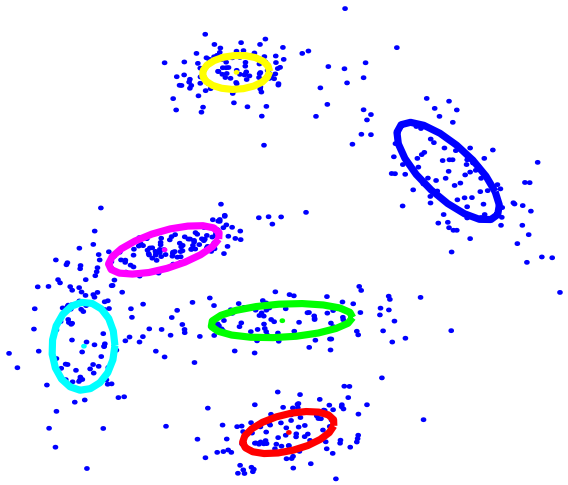
Model Comparison:

$$\begin{aligned} P(m|\mathcal{D}) &= \frac{P(\mathcal{D}|m)P(m)}{P(\mathcal{D})} \\ P(\mathcal{D}|m) &= \int P(\mathcal{D}|\theta, m)P(\theta|m)\, d\theta \end{aligned}$$

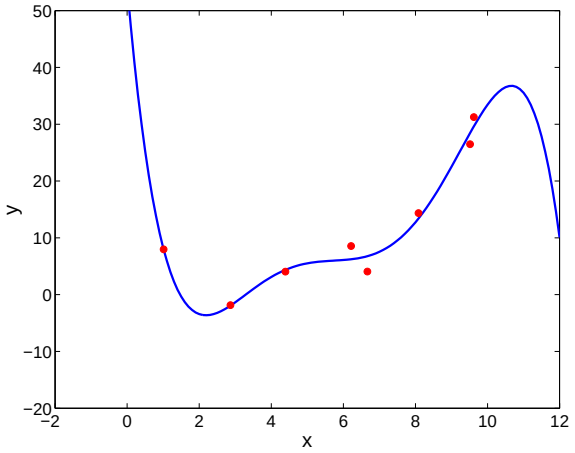
Prediction:

$$\begin{aligned} P(x|\mathcal{D}, m) &= \int P(x|\theta, \mathcal{D}, m)P(\theta|\mathcal{D}, m)d\theta \\ P(x|\mathcal{D}, m) &= \int P(x|\theta, m)P(\theta|\mathcal{D}, m)d\theta \qquad \text{(if } x \text{ is iid given } \theta\text{)} \end{aligned}$$

Model Comparison: two examples



e.g. selecting m, the number of Gaussians in a mixture model



e.g. selecting m the order of a polynomial in a nonlinear regression model

$$P(m|\mathcal{D}) = \frac{P(\mathcal{D}|m)P(m)}{P(\mathcal{D})}, \qquad P(\mathcal{D}|m) = \int P(\mathcal{D}|\theta, m)P(\theta|m)\, d\theta$$

A possible procedure:

1. place a prior on m, $P(m)$
2. given data, use Bayes rule to infer $P(m|\mathcal{D})$

What is the problem with this procedure?

Real data is complicated

Example 1:
You are trying to model people's patterns of movie preferences. You believe there are "clusters" of people, so you use a mixture model...

- How should you pick $P(m)$, your prior over how many clusters there are? teenagers, people who like action movies, people who like romantic comedies, people who like horror movies, people who like movies with Marlon Brando, people who like action movies but not science fiction, etc etc...

- Even if there are a few well defined clusters, they are unlikely to be Gaussian in the variables you measure. To model complicated distributions you might need many Gaussians for each cluster.

- Conclusion: any small finite number seems unreasonable

Real data is complicated

Example 2:
You are trying to model crop yield as a function of rainfall, amount of sunshine, amount of fertilizer, etc. You believe this relationship is nonlinear, so you decide to model it with a polynomial.

- How should you pick $P(m)$, your prior over what is the order of the polynomial?
- Do you believe the relationship could be linear? quadratic? cubic? What about the interactions between input variabes?
- Conclusion: any order polynomial seems unreasonable.

How do we adequately capture our beliefs?

Non-parametric Bayesian Models

- Bayesian methods are most powerful when your prior adequately captures your beliefs.
- Inflexible models (e.g. mixture of 5 Gaussians, 4th order polynomial) yield unreasonable inferences.
- Non-parametric models are a way of getting very flexible models.
- Many can be derived by starting with a finite parametric model and taking the limit as number of parameters $\rightarrow \infty$
- Non-parametric models can automatically infer an adequate model size/complexity from the data, without needing to explicitly do Bayesian model comparison.[2]

[2] Even if you believe there are infinitely many possible clusters, you can still infer how many clusters are *represented* in a finite set of n data points.

Outline

- Introduction
- Gaussian Processes (GP)
- Dirichlet Processes (DP), different representations:
  - Chinese Restaurant Process (CRP)
  - Urn Model
  - Stick Breaking Representation
  - Infinite limit of mixture models and Dirichlet process mixtures (DPM)
- Hierarchical Dirichlet Processes
- Infinite Hidden Markov Models
- Polya Trees
- Dirichlet Diffusion Trees
- Indian Buffet Processes

Gaussian Processes

A Gaussian process defines a distribution over functions, f, where f is a function mapping some input space $\mathcal{X}$ to $\Re$.

$$f : \mathcal{X} \rightarrow \Re.$$

Notice that f can be an infinite-dimensional quantity (e.g. if $\mathcal{X} = \Re$)

Let's call this distribution $P(f)$

Let $\mathbf{f} = (f(x_1), f(x_2), \ldots, f(x_n))$ be an n-dimensional vector of function values evaluated at n points $x_i \in \mathcal{X}$. Note $\mathbf{f}$ is a random variable.

Definition: $P(f)$ is a Gaussian process if for *any* finite subset $\{x_1, \ldots, x_n\} \subset \mathcal{X}$, the marginal distribution over that finite subset $P(\mathbf{f})$ has a multivariate Gaussian distribution.

Gaussian process covariance functions

$P(f)$ is a Gaussian process if for *any* finite subset $\{x_1, \ldots, x_n\} \subset \mathcal{X}$, the marginal distribution over that finite subset $P(\mathbf{f})$ has a multivariate Gaussian distribution.

Gaussian processes (GPs) are parameterized by a mean function, $\mu(x)$, and a covariance function, $c(x, x')$.

$$P(f(x), f(x')) = \mathsf{N}(\boldsymbol{\mu}, \Sigma)$$

where

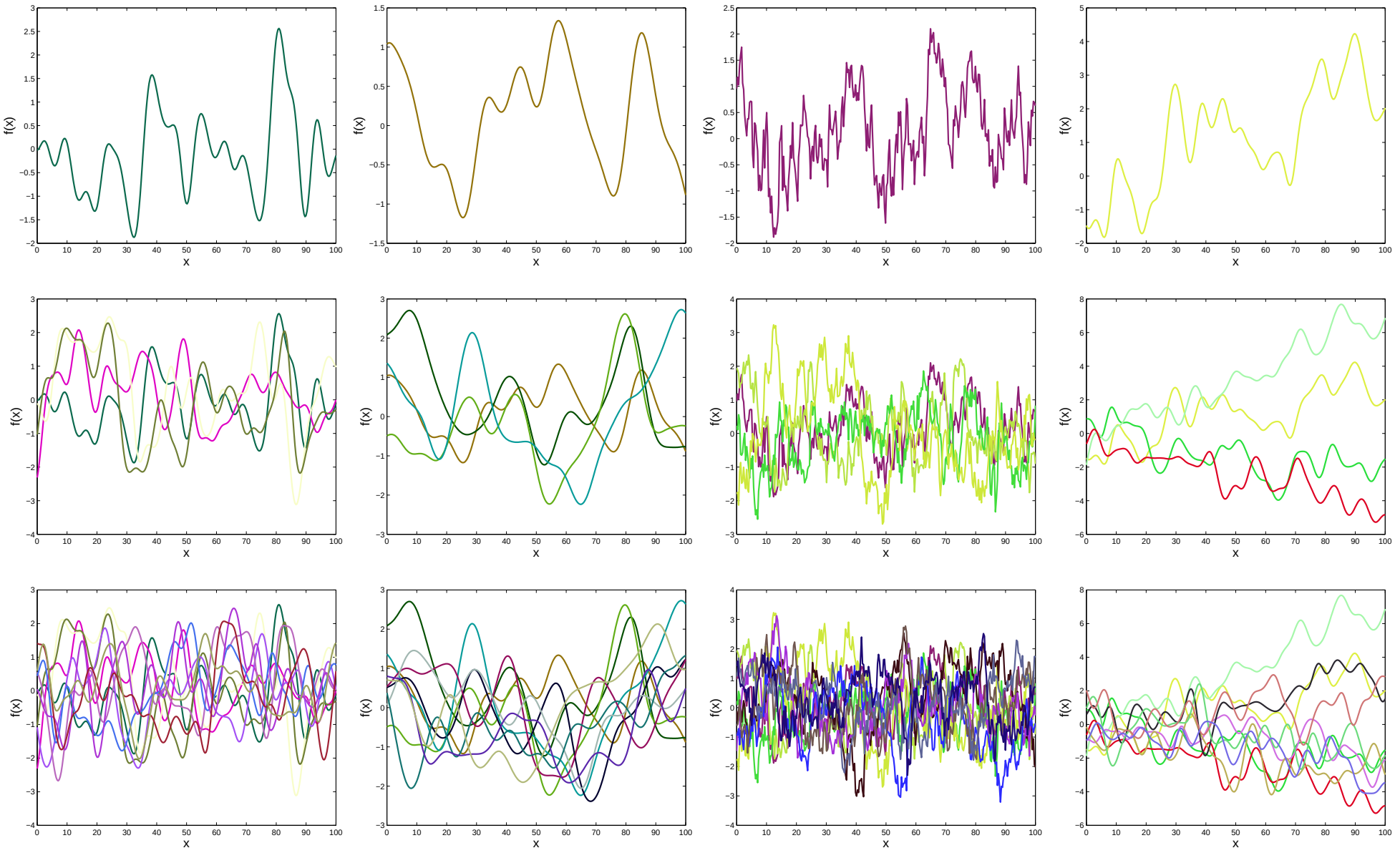
$$\boldsymbol{\mu} = \begin{bmatrix} \mu(x) \\ \mu(x') \end{bmatrix} \quad \Sigma = \begin{bmatrix} c(x,x) & c(x,x') \\ c(x',x) & c(x',x') \end{bmatrix}$$

and similarly for $P(f(x_1), \ldots, f(x_n))$ where now $\boldsymbol{\mu}$ is an $n \times 1$ vector and Σ is an $n \times n$ matrix.

E.g.: $c(x_i, x_j) = v_0 \exp\left\{ -\left(\frac{|x_i - x_j|}{\lambda} \right)^{\alpha} \right\} + v_1 + v_2\, \delta_{ij}$ with params $(v_0, v_1, v_2, \lambda, \alpha)$

Once the mean and covariance functions are defined, everything else about GPs follows from the basic rules of probability applied to mutivariate Gaussians.

Samples from Gaussian processes with different $c(x, x')$



Using Gaussian processes for nonlinear regression

Imagine observing a data set $\mathcal{D} = \{(x_i, y_i)_{i=1}^n\} = (\mathbf{x}, \mathbf{y})$.

Model:

$$\begin{aligned} y_i &= f(x_i) + \epsilon_i \\ f &\sim \mathsf{GP}(\cdot|0, c) \\ \epsilon_i &\sim \mathsf{N}(\cdot|0, \sigma^2) \end{aligned}$$

Prior on f is a GP, likelihood is Gaussian, therefore posterior on f is also a GP.

We can use this to make predictions

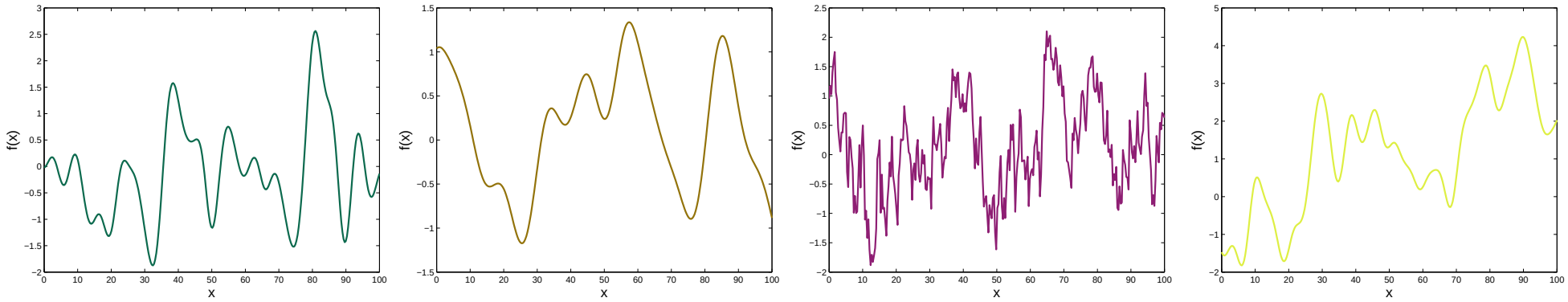
$$P(y'|x', \mathcal{D}) = \int df P(y'|x', f, \mathcal{D}) P(f|\mathcal{D})$$

We can also compute the marginal likelihood (evidence) and use this to compare or tune covariance functions

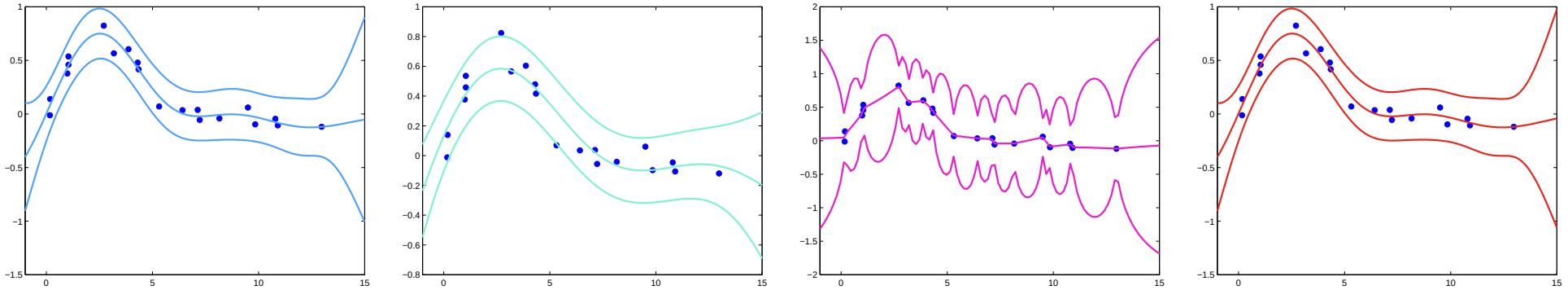
$$P(\mathbf{y}|\mathbf{x}) = \int df P(\mathbf{y}|f, \mathbf{x}) P(f)$$

Prediction using GPs with different $c(x, x')$

A sample from the prior for each covariance function:



Corresponding predictions, mean with two standard deviations:



From linear regression to GPs:

- Linear regression with inputs x_i and outputs y_i: $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$
- Linear regression with K basis functions: $y_i = \sum_{k=1}^{K} \beta_k \, \phi_k(x_i) + \epsilon_i$
- Bayesian linear regression with basis functions:

$$\beta_k \sim \mathsf{N}(\cdot|0, \lambda_k) \quad \text{(independent of } \beta_\ell, \, \forall \ell \neq k), \qquad \epsilon_i \sim \mathsf{N}(\cdot|0, \sigma^2)$$

- Integrating out the coefficients, β_j, we find:

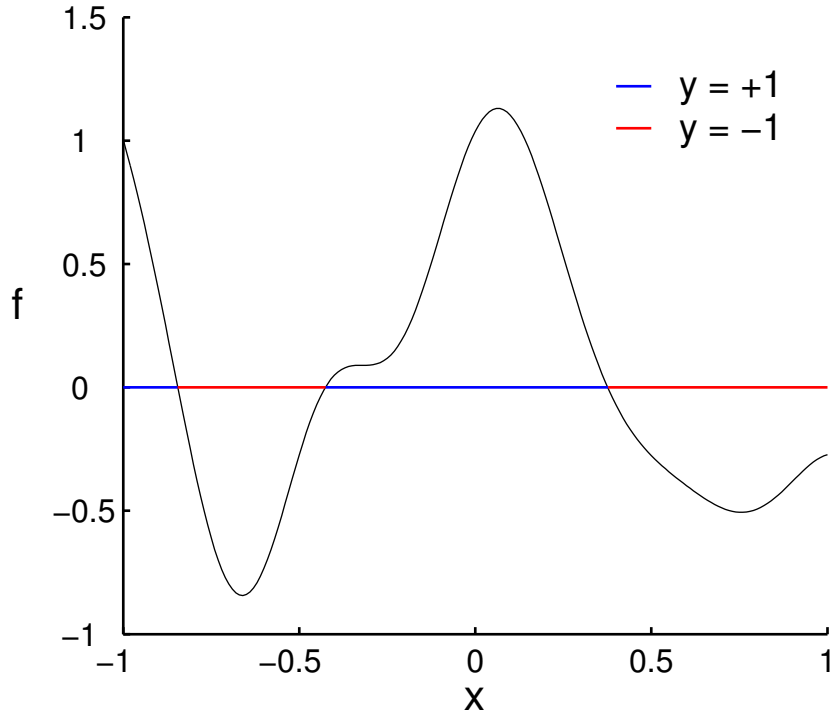
$$E[y_i] = 0, \qquad Cov(y_i, y_j) = C_{ij} \stackrel{\text{def}}{=} \sum_k \lambda_k \, \phi_k(x_i) \, \phi_k(x_j) + \delta_{ij} \sigma^2$$

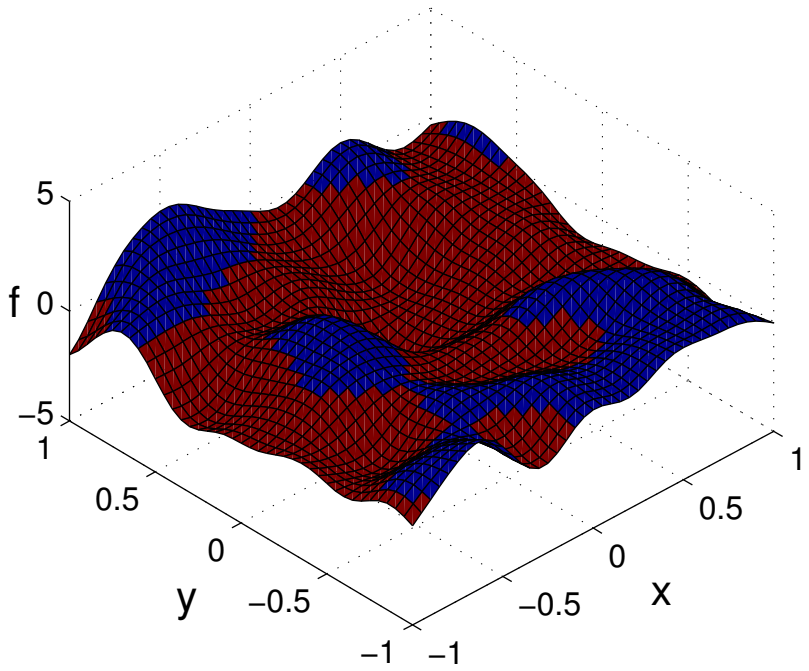
This is a Gaussian process with covariance function $c(x_i, x_j) = C_{ij}$.

This Gaussian process has a finite number (K) of basis functions. Many useful GP covariance functions correspond to infinitely many basis functions.

Using Gaussian Processes for Classification

Binary classification problem: Given a data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, with binary class labels $y_i \in \{-1, +1\}$, infer class label probabilities at new points.





There are many ways to relate function values $f(\mathbf{x}_i)$ to class probabilities:

$$p(y|f) = \begin{cases} \frac{1}{1+\exp(-yf)} & \text{sigmoid (logistic)} \\ \Phi(yf) & \text{cumulative normal (probit)} \\ \boldsymbol{H}(yf) & \text{threshold} \\ \epsilon + (1-2\epsilon)\boldsymbol{H}(yf) & \text{robust threshold} \end{cases}$$

Outline

- Introduction
- Gaussian Processes (GP)
- Dirichlet Processes (DP), different representations:
  - Chinese Restaurant Process (CRP)
  - Urn Model
  - Stick Breaking Representation
  - Infinite limit of mixture models and Dirichlet process mixtures (DPM)
- Hierarchical Dirichlet Processes
- Infinite Hidden Markov Models
- Polya Trees
- Dirichlet Diffusion Trees
- Indian Buffet Processes

Dirichlet Distribution

The Dirichlet distribution is a distribution over the K-dim probability simplex.

Let $\mathbf{p}$ be a K-dimensional vector s.t. $\forall j : p_j \geq 0$ and $\sum_{j=1}^{K} p_j = 1$

$$P(\mathbf{p}|\boldsymbol{\alpha}) = \text{Dir}(\alpha_1, \ldots, \alpha_K) \stackrel{\text{def}}{=} \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_{j=1}^{K} p_j^{\alpha_j - 1}$$

where the first term is a normalization constant[3] and $E(p_j) = \alpha_j / (\sum_k \alpha_k)$

The Dirichlet is conjugate to the multinomial distribution. Let

$$c|\mathbf{p} \sim \text{Multinomial}(\cdot|\mathbf{p})$$

That is, $P(c = j|\mathbf{p}) = p_j$. Then the posterior is also Dirichlet:

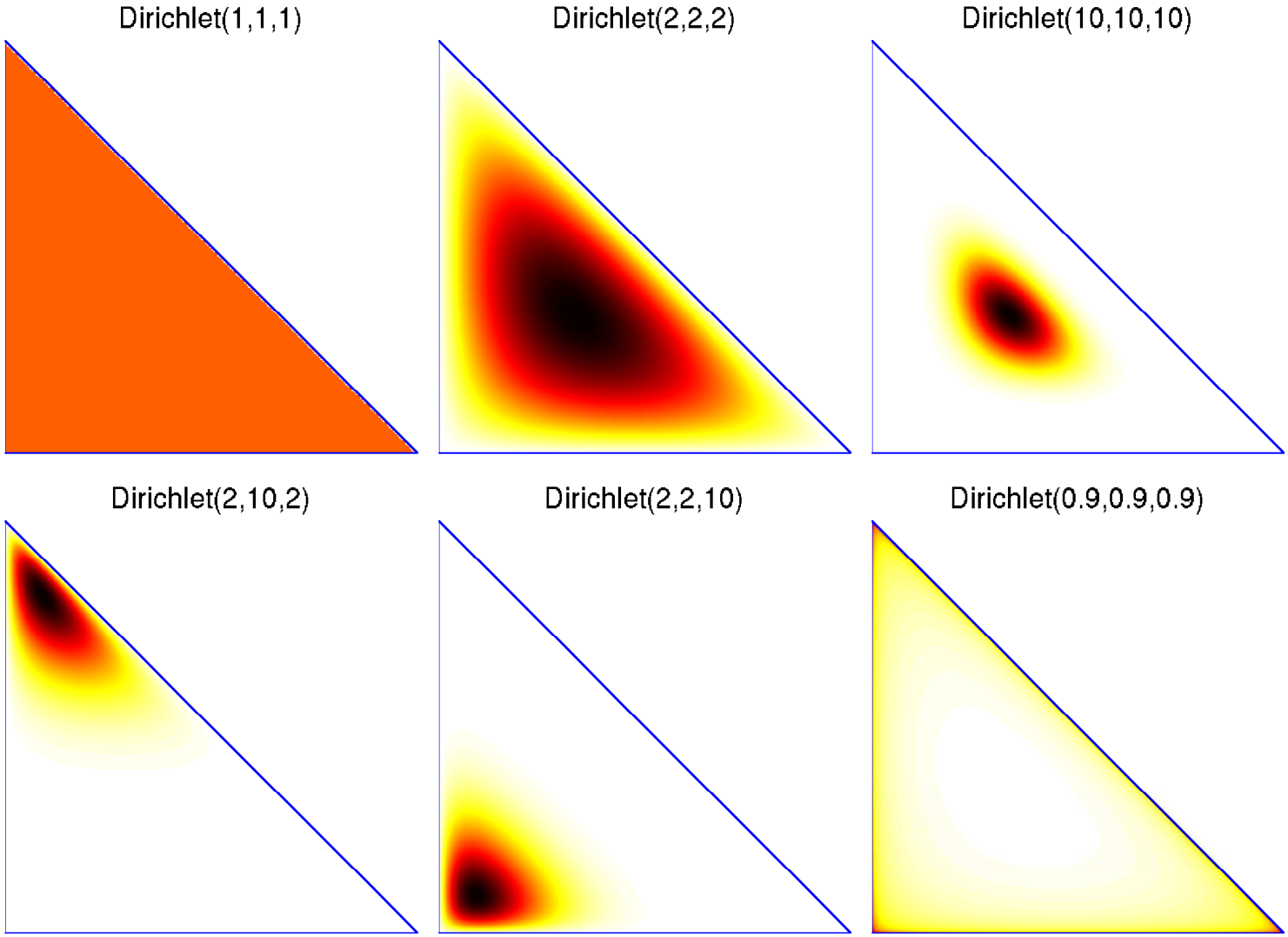
$$P(\mathbf{p}|c = j, \boldsymbol{\alpha}) = \frac{P(c = j|\mathbf{p})P(\mathbf{p}|\boldsymbol{\alpha})}{P(c = j|\boldsymbol{\alpha})} = \text{Dir}(\boldsymbol{\alpha}')$$

where $\alpha'_j = \alpha_j + 1$, and $\forall \ell \neq j : \alpha'_\ell = \alpha_\ell$

[3] $\Gamma(x) = (x-1)\Gamma(x-1) = \int_0^\infty t^{x-1} e^{-t} dt$. For integer n, $\Gamma(n) = (n-1)!$

Dirichlet Distributions

Examples of Dirichlet distributions over $\mathbf{p} = (p_1, p_2, p_3)$ which can be plotted in 2D since $p_3 = 1 - p_1 - p_2$:



Dirichlet Processes

- Gaussian processes define a distribution over functions

$$f \sim \mathsf{GP}(\cdot|\mu, c)$$

  where μ is the mean function and c is the covariance function.
  We can think of GPs as "infinite-dimensional" Gaussians

- Dirichlet processes define a distribution over distributions (a measure on measures)

$$G \sim \mathsf{DP}(\cdot|G_0, \alpha)$$

  where $\alpha > 0$ is a scaling parameter, and G_0 is the base measure.
  We can think of DPs as "infinite-dimensional" Dirichlet distributions.

Note that both f and G are infinite dimensional objects.

Dirichlet Process

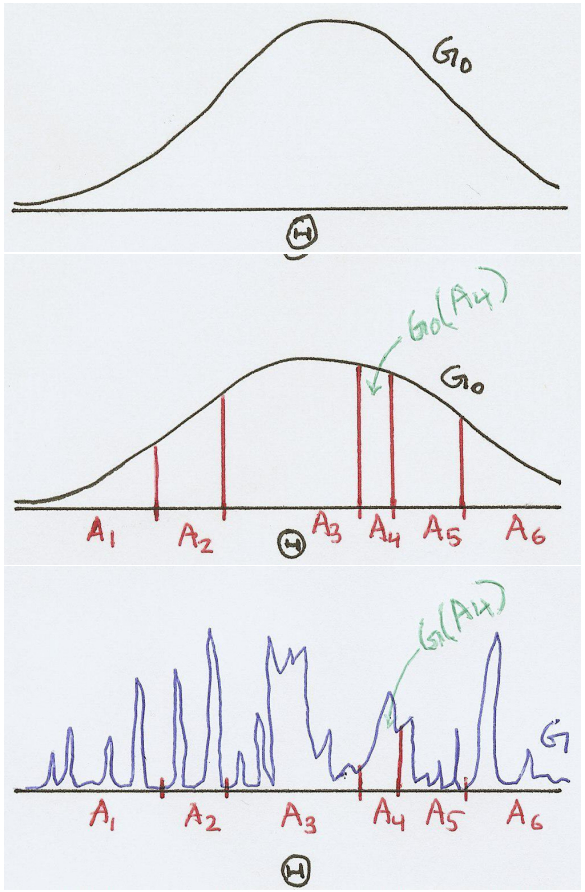
Let Θ be a measurable space, G_0 be a probability measure on Θ, and α a positive real number.

For all $(A_1, \ldots A_K)$ finite partitions of Θ,

$$G \sim \mathsf{DP}(\cdot | G_0, \alpha)$$

means that

$$(G(A_1), \ldots, G(A_K)) \sim \mathsf{Dir}(\alpha G_0(A_1), \ldots, \alpha G_0(A_K))$$



(Ferguson, 1973)

Dirichlet Process

$G \sim \text{DP}(\cdot|G_0, \alpha)$ OK, but what does it look like?

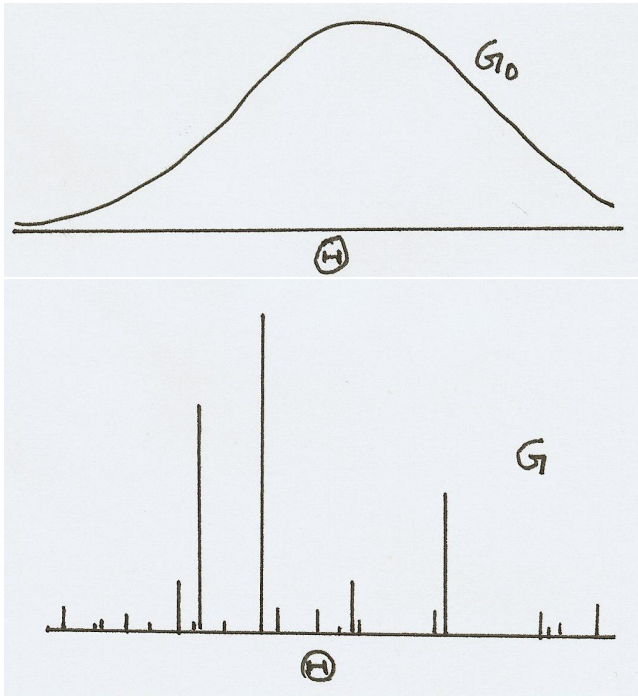
Samples from a DP are discrete with probability one:

$$G(\theta) = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}(\theta)$$

where $\delta_{\theta_k}(\cdot)$ is a Dirac delta at θ_k, and $\theta_k \sim G_0(\cdot)$.

Note: $E(G) = G_0$

As $\alpha \to \infty$, G looks more like G_0.



Dirichlet Process: Conjugacy

$$G \sim \mathsf{DP}(\cdot|G_0, \alpha)$$

If the prior on G is a DP:

$$P(G) = \mathsf{DP}(G|G_0, \alpha)$$

...and you observe θ...

$$P(\theta|G) = G(\theta)$$

...then the posterior is also a DP:

$$P(G|\theta) = \mathsf{DP}\left(\frac{\alpha}{\alpha+1}G_0 + \frac{1}{\alpha+1}\delta_\theta,\ \alpha+1\right)$$

Generalization for n observations:

$$P(G|\theta_1, \ldots, \theta_n) = \mathsf{DP}\left(\frac{\alpha}{\alpha+n}G_0 + \frac{1}{\alpha+n}\sum_{i=1}^{n}\delta_{\theta_i},\ \alpha+n\right)$$

Analogous to Dirichlet being conjugate to multinomial observations.

Dirichlet Process

Blackwell and MacQueen's (1973) urn representation

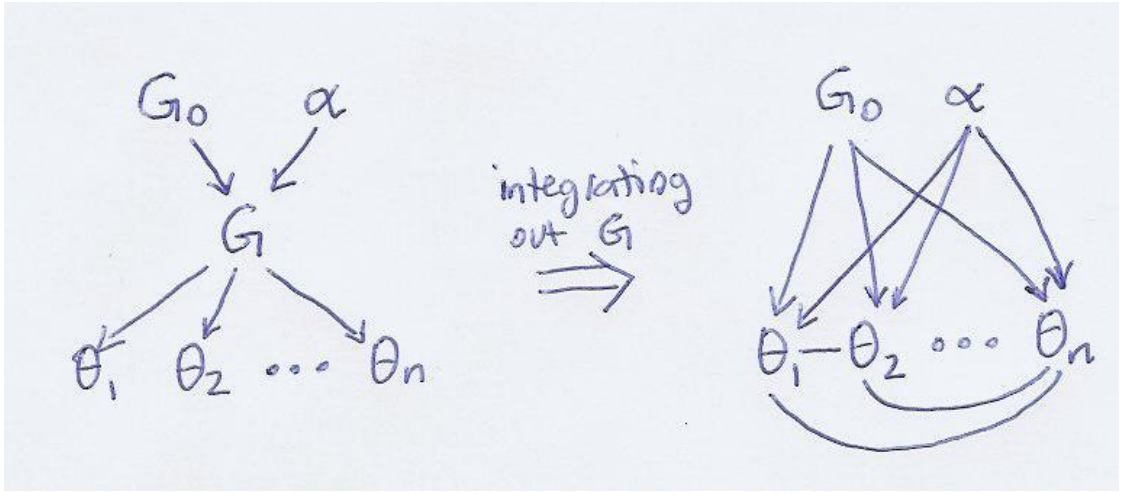
$$G \sim \mathrm{DP}(\cdot|G_0, \alpha) \quad \text{and} \quad \theta|G \sim G(\cdot)$$

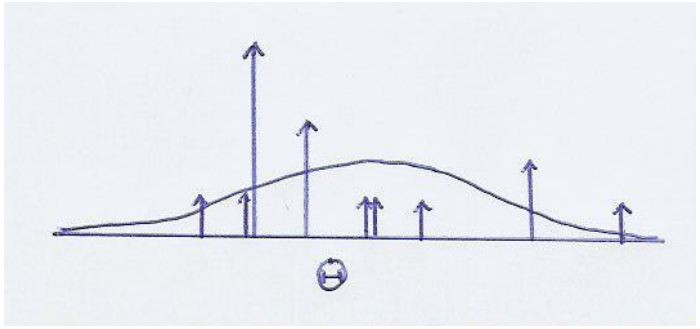
Then

$$\theta_n|\theta_1, \ldots \theta_{n-1}, G_0, \alpha \quad \sim \quad \frac{\alpha}{n-1+\alpha} G_0(\cdot) + \frac{1}{n-1+\alpha} \sum_{j=1}^{n-1} \delta_{\theta_j}(\cdot)$$

$$P(\theta_n|\theta_1, \ldots \theta_{n-1}, G_0, \alpha) \quad \propto \quad \int dG \prod_{j=1}^{n} P(\theta_j|G) P(G|G_0, \alpha)$$

The model exhibits a "clustering effect".





Chinese Restaurant Process (CRP)

This shows the clustering effect explicitly.
Restaurant has infinitely many tables $k = 1, \ldots$.
Customers are indexed by $i = 1, \ldots$, with values ϕ_i
Tables have values θ_k drawn from G_0

K = total number of occupied tables so far.
n = total number of customers so far.
n_k = number of customers seated at table k

Generating from a CRP:
customer 1 enters the restaurant and sits at table 1.
$\phi_1 = \theta_1$ where $\theta_1 \sim G_0$, $K = 1$, $n = 1$, $n_1 = 1$
for $n = 2, \ldots,$

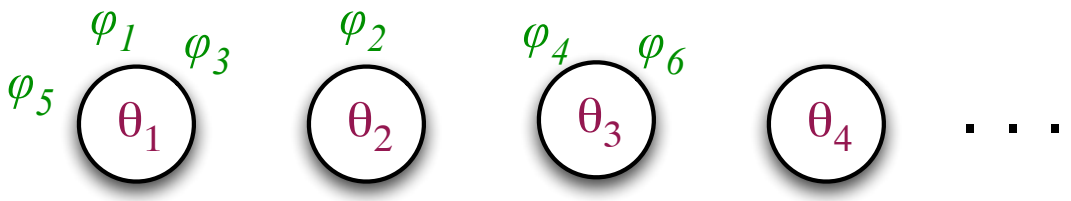
customer n sits at table $\begin{cases} k & \text{with prob } \frac{n_k}{n-1+\alpha} \quad \text{for } k = 1 \ldots K \\ K+1 & \text{with prob } \frac{\alpha}{n-1+\alpha} \quad \text{(new table)} \end{cases}$

if new table was chosen **then** $K \leftarrow K + 1$, $\theta_{K+1} \sim G_0$ **endif**
set ϕ_n to θ_k of the table k that customer n sat at; set $n_k \leftarrow n_k + 1$
endfor

Clustering effect: New students entering a school join clubs in proportion to how popular those clubs already are ($\propto n_k$). With some probability (proportional to α), a new student starts a new club.

(Aldous, 1985)

Chinese Restaurant Process



Generating from a CRP:
customer 1 enters the restaurant and sits at table 1.
$\phi_1 = \theta_1$ where $\theta_1 \sim G_0$, $K = 1$, $n = 1$, $n_1 = 1$
for $n = 2, \ldots,$

customer n sits at table $\begin{cases} k & \text{with prob } \frac{n_k}{n-1+\alpha} \quad \text{for } k = 1 \ldots K \\ K+1 & \text{with prob } \frac{\alpha}{n-1+\alpha} \quad \text{(new table)} \end{cases}$

if new table was chosen **then** $K \leftarrow K+1$, $\theta_{K+1} \sim G_0$ **endif**
set ϕ_n to θ_k of the table k that customer n sat at; set $n_k \leftarrow n_k + 1$
endfor

The resulting conditional distribution over ϕ_n:

$$\phi_n | \phi_1, \ldots, \phi_{n-1}, G_0, \alpha \sim \frac{\alpha}{n-1+\alpha} G_0(\cdot) + \sum_{k=1}^{K} \frac{n_k}{n-1+\alpha} \delta_{\theta_k}(\cdot)$$

Relationship between CRPs and DPs

- DP is a distribution over distributions
- DP results in discrete distributions, so if you draw n points you are likely to get repeated values
- A DP induces a partitioning of the n points
  e.g. $(1\,3\,4)\,(2\,5) \Leftrightarrow \phi_1 = \phi_3 = \phi_4 \neq \phi_2 = \phi_5$
- CRP is the corresponding distribution over partitions

Dirichlet Processes: Stick Breaking Representation

$$G \sim \text{DP}(\cdot|G_0, \alpha)$$

Samples G from a DP can be represented as follows:

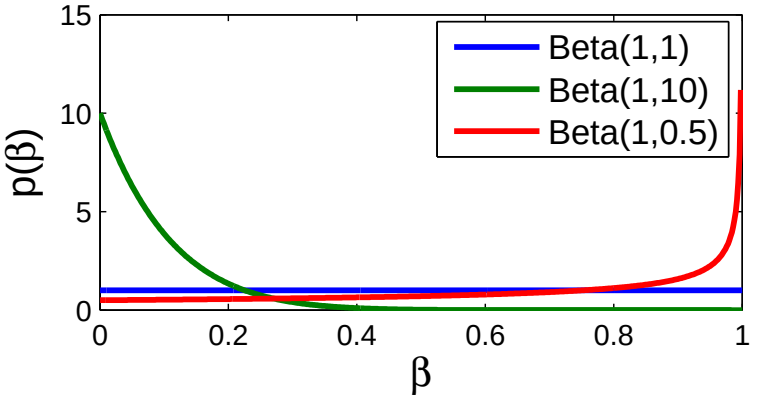
$$G(\cdot) = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}(\cdot)$$

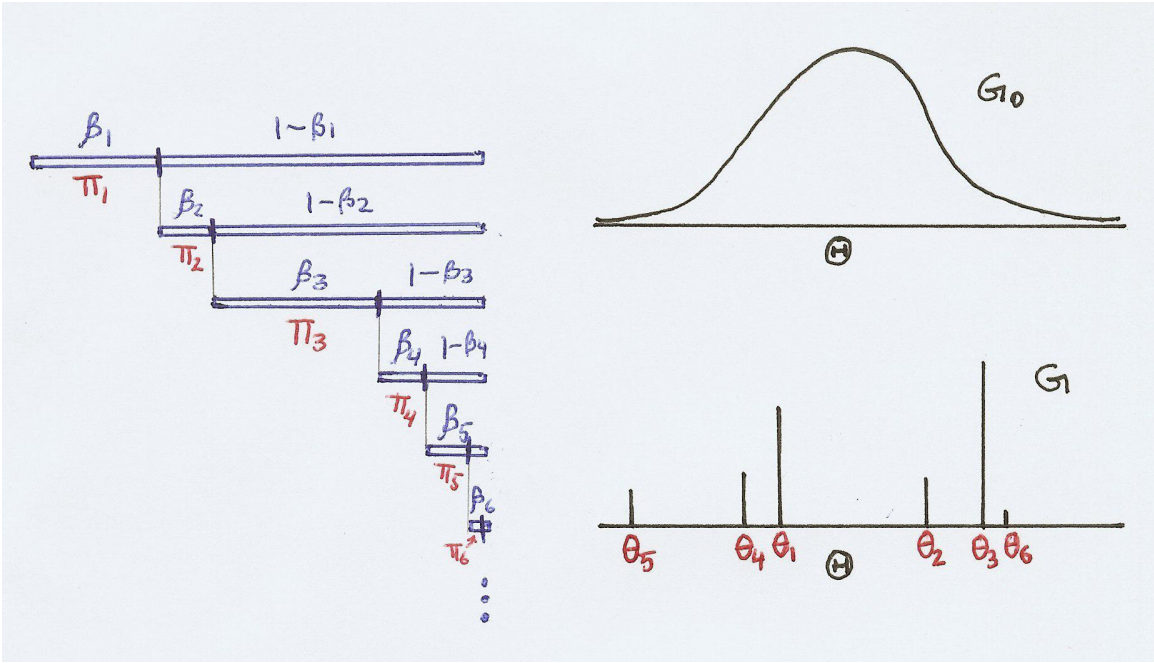
where $\theta_k \sim G_0(\cdot)$, $\sum_{k=1}^{\infty} \pi_k = 1$,

$$\pi_k = \beta_k \prod_{j=1}^{k-1} (1 - \beta_j)$$

and

$$\beta_k \sim \text{Beta}(\cdot|1, \alpha)$$





(Sethuraman, 1994)

Other Stick Breaking Processes

- Dirichlet Process (Sethuraman, 1994):

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

- Beta Two-parameter Process (Ishwaran and Zarepour, 2000):

$$\beta_k \sim \text{Beta}(a, b)$$

- Pitman-Yor Process (aka two-parameter Poisson-Dirichlet Process; Pitman & Yor (1997)):

$$\beta_k \sim \text{Beta}(1 - a, b + ka)$$

Note: mean of a $\text{Beta}(a, b)$ is $a/(a + b)$

Dirichlet Processes: Big Picture

There are many ways to derive the Dirichlet Process:

- Dirichlet distribution
- Urn model
- Chinese restaurant process
- Stick breaking
- Gamma process[4]

[4] I didn't talk about this one

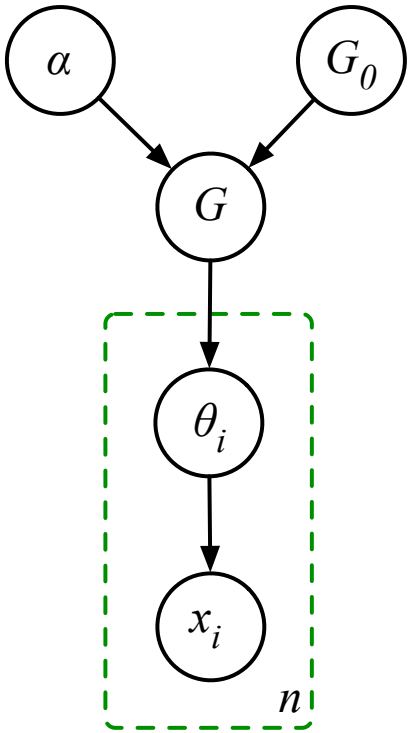
Dirichlet Process Mixtures

DPs are discrete with probability one, so they are not suitable for use as a prior on continuous densities.

In a Dirichlet Process **Mixture**, we draw the parameters of a mixture model from a draw from a DP:

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

$$\theta_i \sim G(\cdot)$$
$$x_i \sim p(\cdot|\theta_i)$$

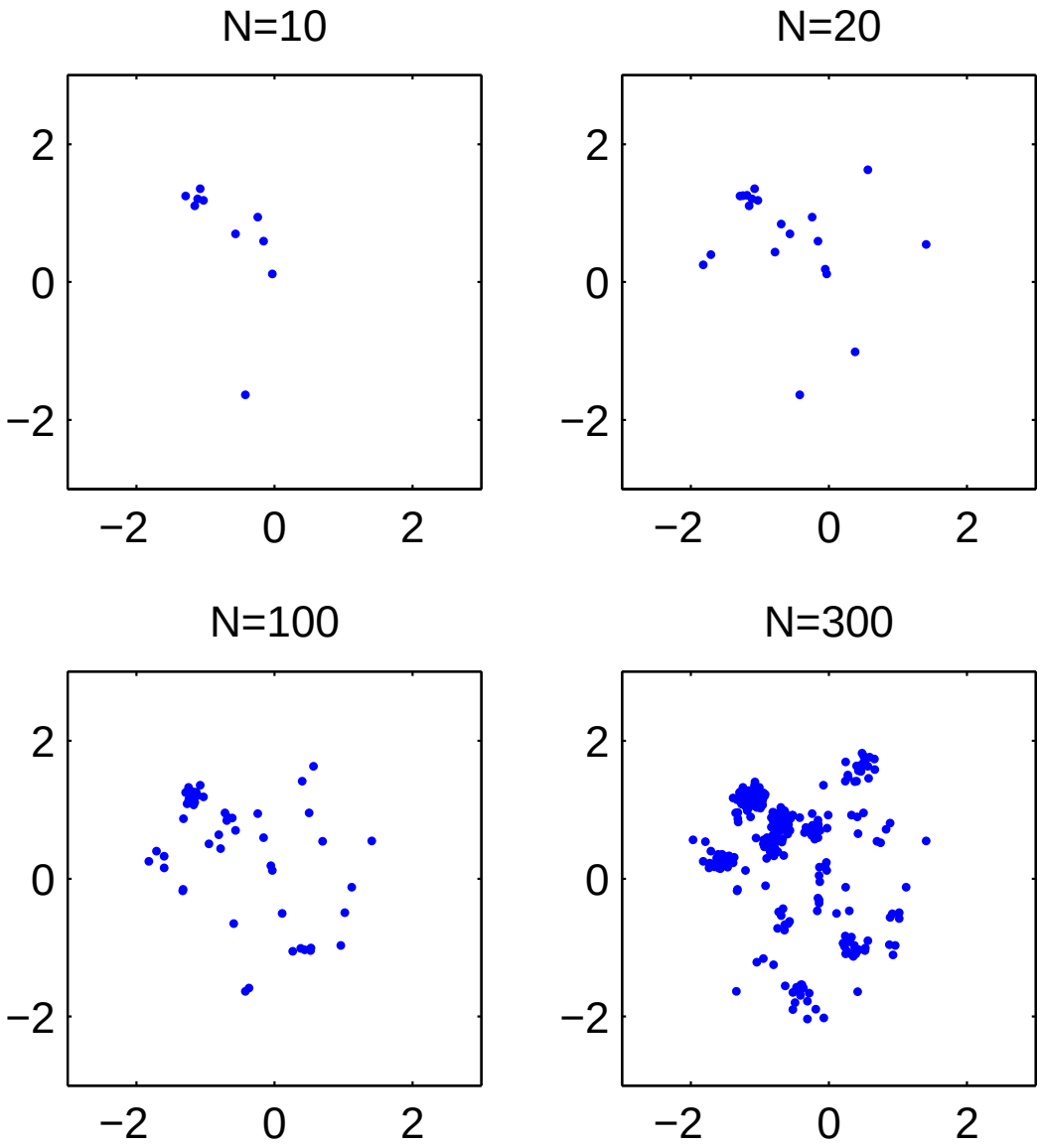


For example, if $p(\cdot|\theta)$ is a Gaussian density with parameters θ, then we have a Dirichlet Process Mixture of Gaussians

Of course, $p(\cdot|\theta)$ could be any density.

We can derive DPMs from finite mixture models (Neal)...

Samples from a Dirichlet Process Mixture of Gaussians



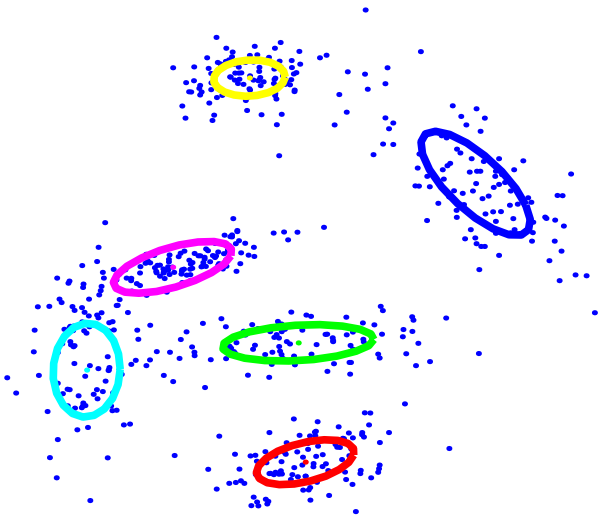
Notice that more structure (clusters) appear as you draw more points.

(figure inspired by Neal)

Dirichlet Process Mixtures (Infinite Mixtures)

Consider using a finite mixture of K components to model a data set $\mathcal{D} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\}$

$$
\begin{aligned}
p(\mathbf{x}^{(i)}|\boldsymbol{\theta}) &= \sum_{j=1}^{K} \pi_j \, p_j(\mathbf{x}^{(i)}|\boldsymbol{\theta}_j) \\
&= \sum_{j=1}^{K} P(s^{(i)} = j|\boldsymbol{\pi}) \, p_j(\mathbf{x}^{(i)}|\boldsymbol{\theta}_j, s^{(i)} = j)
\end{aligned}
$$



Distribution of indicators $\mathbf{s} = (s^{(1)}, \ldots, s^{(n)})$ given $\boldsymbol{\pi}$ is **multinomial**

$$
P(s^{(1)}, \ldots, s^{(n)}|\boldsymbol{\pi}) = \prod_{j=1}^{K} \pi_j^{n_j}, \qquad n_j \stackrel{\text{def}}{=} \sum_{i=1}^{n} \delta(s^{(i)}, j) \; .
$$

Assume mixing proportions $\boldsymbol{\pi}$ have a given symmetric conjugate **Dirichlet prior**

$$
p(\boldsymbol{\pi}|\alpha) = \frac{\Gamma(\alpha)}{\Gamma(\alpha/K)^K} \prod_{j=1}^{K} \pi_j^{\alpha/K - 1}
$$

Dirichlet Process Mixtures (Infinite Mixtures) - II

Distribution of indicators $\mathbf{s} = (s^{(1)}, \ldots, s^{(n)})$ given $\boldsymbol{\pi}$ is **multinomial**

$$P(s^{(1)}, \ldots, s^{(n)}|\boldsymbol{\pi}) = \prod_{j=1}^{K} \pi_j^{n_j}, \qquad n_j \stackrel{\text{def}}{=} \sum_{i=1}^{n} \delta(s^{(i)}, j) \ .$$

Mixing proportions $\boldsymbol{\pi}$ have a symmetric conjugate **Dirichlet prior**

$$p(\boldsymbol{\pi}|\alpha) = \frac{\Gamma(\alpha)}{\Gamma(\alpha/K)^K} \prod_{j=1}^{K} \pi_j^{\alpha/K-1}$$

Integrating out the mixing proportions, $\boldsymbol{\pi}$, we obtain

$$P(s^{(1)}, \ldots, s^{(n)}|\alpha) = \int d\boldsymbol{\pi} \ P(\mathbf{s}|\boldsymbol{\pi}) P(\boldsymbol{\pi}|\alpha) = \frac{\Gamma(\alpha)}{\Gamma(n+\alpha)} \prod_{j=1}^{K} \frac{\Gamma(n_j + \alpha/K)}{\Gamma(\alpha/K)}$$

Dirichlet Process Mixtures (Infinite Mixtures) - III

Starting from $$P(\mathbf{s}|\alpha) = \frac{\Gamma(\alpha)}{\Gamma(n+\alpha)} \prod_{j=1}^{K} \frac{\Gamma(n_j + \alpha/K)}{\Gamma(\alpha/K)}$$

Conditional Probabilities: Finite K

$$P(s^{(i)} = j|\mathbf{s}_{-i}, \alpha) = \frac{n_{-i,j} + \alpha/K}{n - 1 + \alpha}$$

where $\mathbf{s}_{-i}$ denotes all indices except i, and $n_{-i,j} \stackrel{\text{def}}{=} \sum_{\ell \neq i} \delta(s^{(\ell)}, j)$

DP: more populous classes are more more likely to be joined

Conditional Probabilities: Infinite K

Taking the limit as $K \to \infty$ yields the conditionals

$$P(s^{(i)} = j|\mathbf{s}_{-i}, \alpha) = \begin{cases} \frac{n_{-i,j}}{n-1+\alpha} & j \text{ represented} \\ \frac{\alpha}{n-1+\alpha} & \text{all } j \text{ not represented} \end{cases}$$

Left over mass, α, $\Rightarrow$ **countably infinite** number of indicator settings.
Gibbs sampling from posterior of indicators is often easy!

Approximate Inference in DPMs

- Gibbs sampling (e.g. Escobar and West, 1995; Neal, 2000; Rasmussen, 2000)
- Variational approximation (Blei and Jordan, 2005)
- Expectation propagation (Minka and Ghahramani, 2003)
- Hierarchical clustering (Heller and Ghahramani, 2005)

Hierarchical Dirichlet Processes (HDP)

Assume you have data which is divided into J groups.

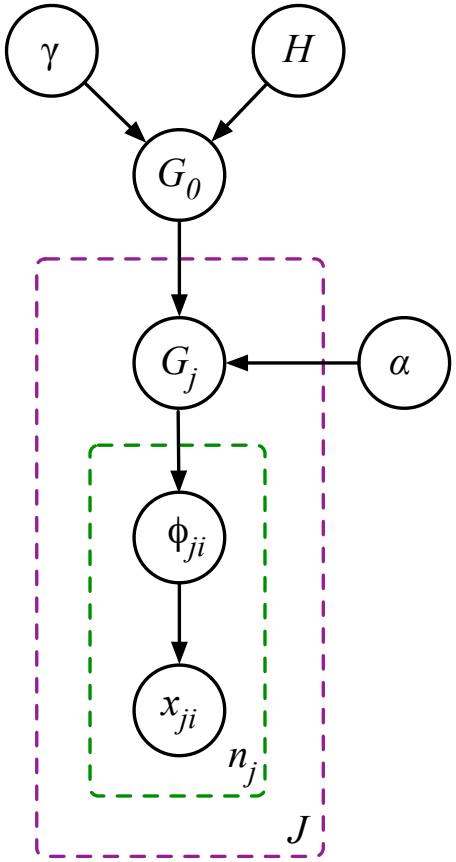
You assume there are clusters within each group, but you also believe these clusters are shared between groups (i.e. data points in different groups can belong to the same cluster).

In an HDP there is a common DP:

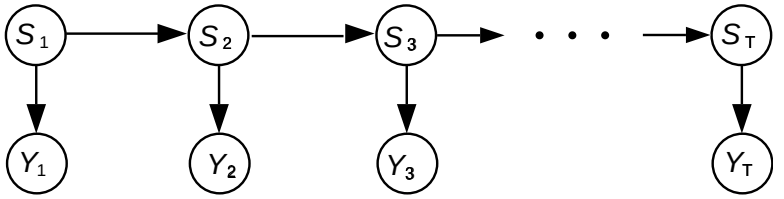
$$G_0|H,\gamma \sim \text{DP}(\cdot|H,\gamma)$$

Which forms the base measure for a draw from a DP within each group

$$G_j|G_0,\alpha \sim \text{DP}(\cdot|G_0,\alpha)$$

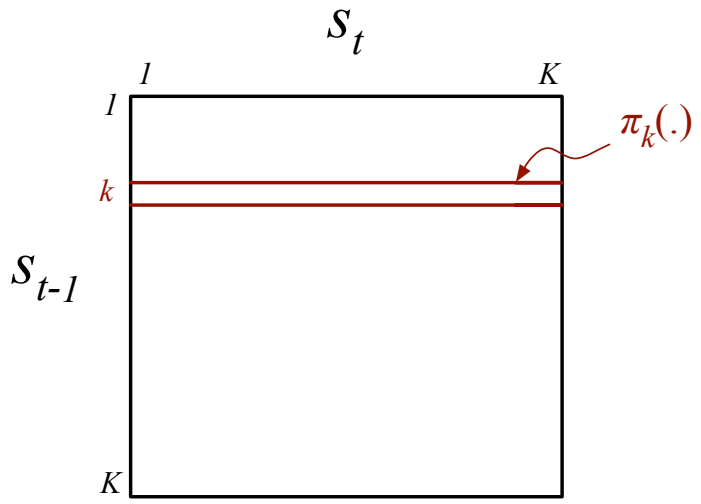


Infinite Hidden Markov Models



Can be derived from the HDP framework

In an HMM with K states, the transition matrix has $K \times K$ elements.



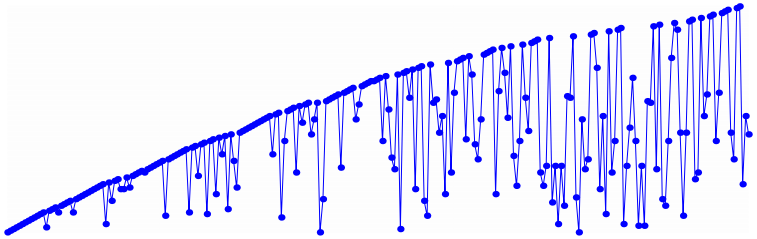
We want to let $K \to \infty$

$$
\begin{array}{rcll}
\beta|\gamma & \sim & \text{Stick}(\cdot|\gamma) & \text{(base distribution over states)} \\
\pi_k|\alpha,\beta & \sim & \text{DP}(\cdot|\alpha,\beta) & \text{(transition parameters for state } k = 1,\dots\text{)} \\
\theta_k|H & \sim & H(\cdot) & \text{(emission parameters for state } k = 1,\dots\text{)} \\
s_t|s_{t-1},(\pi_k)_{k=1}^{\infty} & \sim & \pi_{s_{t-1}}(\cdot) & \text{(transition)} \\
y_t|s_t,(\theta_k)_{k=1}^{\infty} & \sim & p(\cdot|\theta_{s_t}) & \text{(emission)}
\end{array}
$$

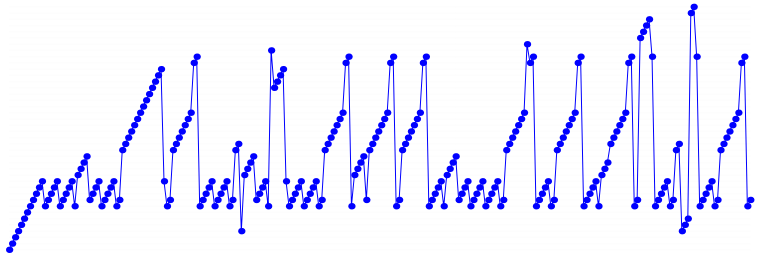
(Beal, Ghahramani, and Rasmussen, 2002) (Teh et al. 2004)

Infinite HMM: Trajectories under the prior

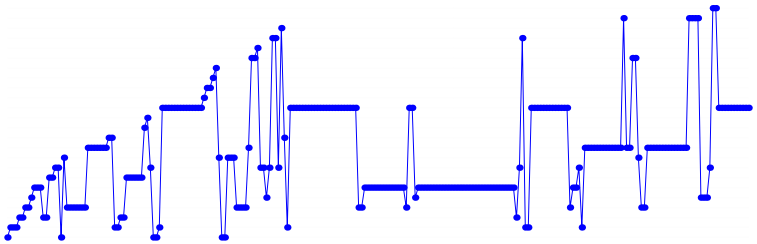
(modified to treat self-transitions specially)



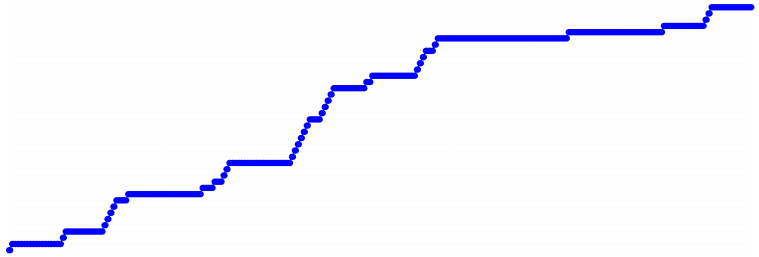
explorative: $a = 0.1,\ b = 1000,\ c = 100$



repetitive: $a = 0,\ b = 0.1,\ c = 100$



self-transitioning: $a = 2,\ b = 2,\ c = 20$



ramping: $a = 1,\ b = 1,\ c = 10000$

Just 3 hyperparameters provide:

- slow/fast dynamics (a)
- sparse/dense transition matrices (b)
- many/few states (c)
- left→right structure, with multiple interacting cycles

Polya Trees

Let Θ be some measurable space.
Assume you have a set Π of nested partitions of the space:

$$\begin{array}{rclcrcl} \Theta & = & B_0 \cup B_1 & & B_0 \cap B_1 & = & \emptyset \\ B_0 & = & B_{00} \cup B_{01} & & B_{00} \cap B_{01} & = & \emptyset \\ B_1 & = & B_{10} \cup B_{11} & & B_{10} \cap B_{11} & = & \emptyset \\ etc & & & & & & \end{array}$$

Let $\mathbf{e} = (e_1, \ldots, e_m)$ be a binary string $e_i \in \{0, 1\}$.
Let $A = \{\alpha_{\mathbf{e}} > 0 : \mathbf{e}\,\text{is a binary string}\}$ and $\Pi = \{B_{\mathbf{e}} \subset \Theta : \mathbf{e}\,\text{is a binary string}\}$

Draw

$$Y_{\mathbf{e}}|A \sim \text{Beta}(\cdot|\alpha_{\mathbf{e}0}, \alpha_{\mathbf{e}1})$$

Then

$$G \sim \text{PT}(\Pi, A)$$

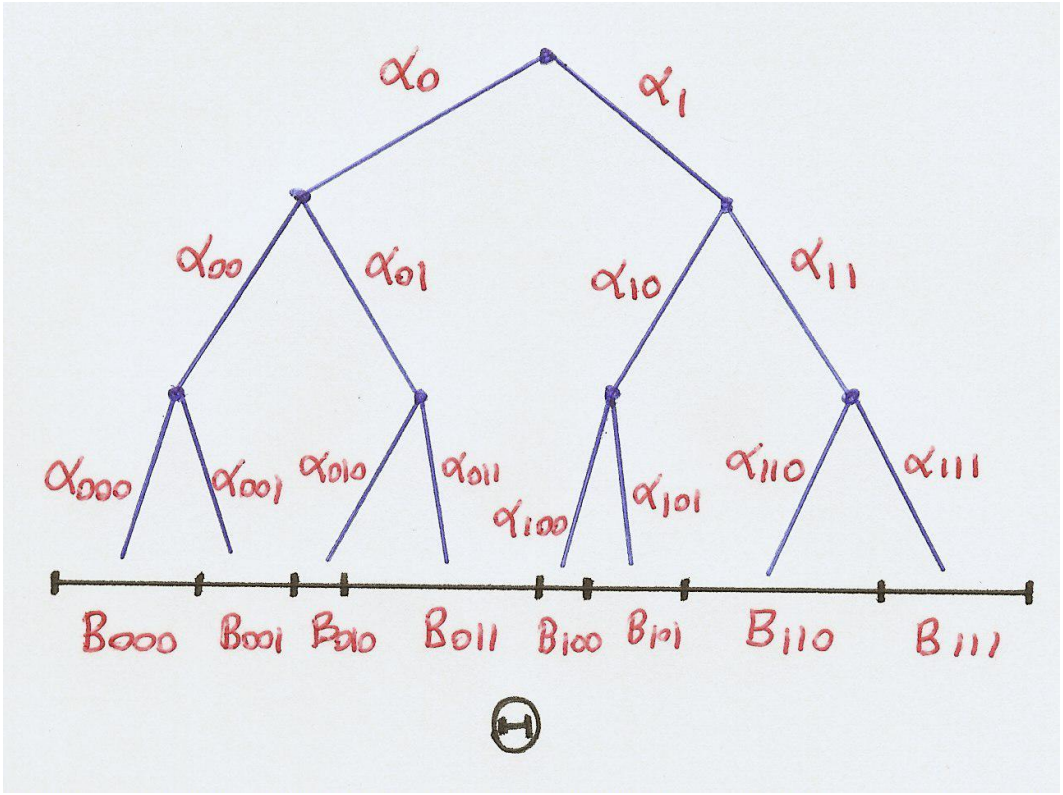
if

$$G(B_{\mathbf{e}}) = \left(\prod_{j=1:e_j=0}^{m} Y_{e_1,\ldots,e_{j-1}} \right) \left(\prod_{j=1:e_j=1}^{m} (1 - Y_{e_1,\ldots,e_{j-1}}) \right)$$

Actually this is really easy to understand...

Polya Trees

You are given a binary tree dividing up Θ, and positive α's on each branch of the tree. You can draw from a Polya tree distribution by drawing Beta random variables dividing up the mass at each branch point.



Properties:

- Polya Trees generalize DPs, a PT is a DP if $\alpha_{\mathbf{e}} = \alpha_{\mathbf{e}0} + \alpha_{\mathbf{e}1}$, for all $\mathbf{e}$.
- **Conjugacy:** $G \sim \mathrm{PT}(\Pi, A)$ and $\theta|G \sim G$, then $G|\theta \sim \mathrm{PT}(\Pi, A')$.
- **Disadvantages:** posterior discontinuities, fixed partitions

(Ferguson, 1974; Lavine, 1992)

Dirichlet Diffusion Trees (DFT)

(Neal, 2001)

In a DPM, parameters of one mixture component are independent of another components – this lack of structure is potentially undesirable.

A DFT is a generalization of DPMs with hierarchical structure between components.

To generate from a DFT, we will consider θ taking a random walk according to a Brownian motion Gaussian diffusion process.

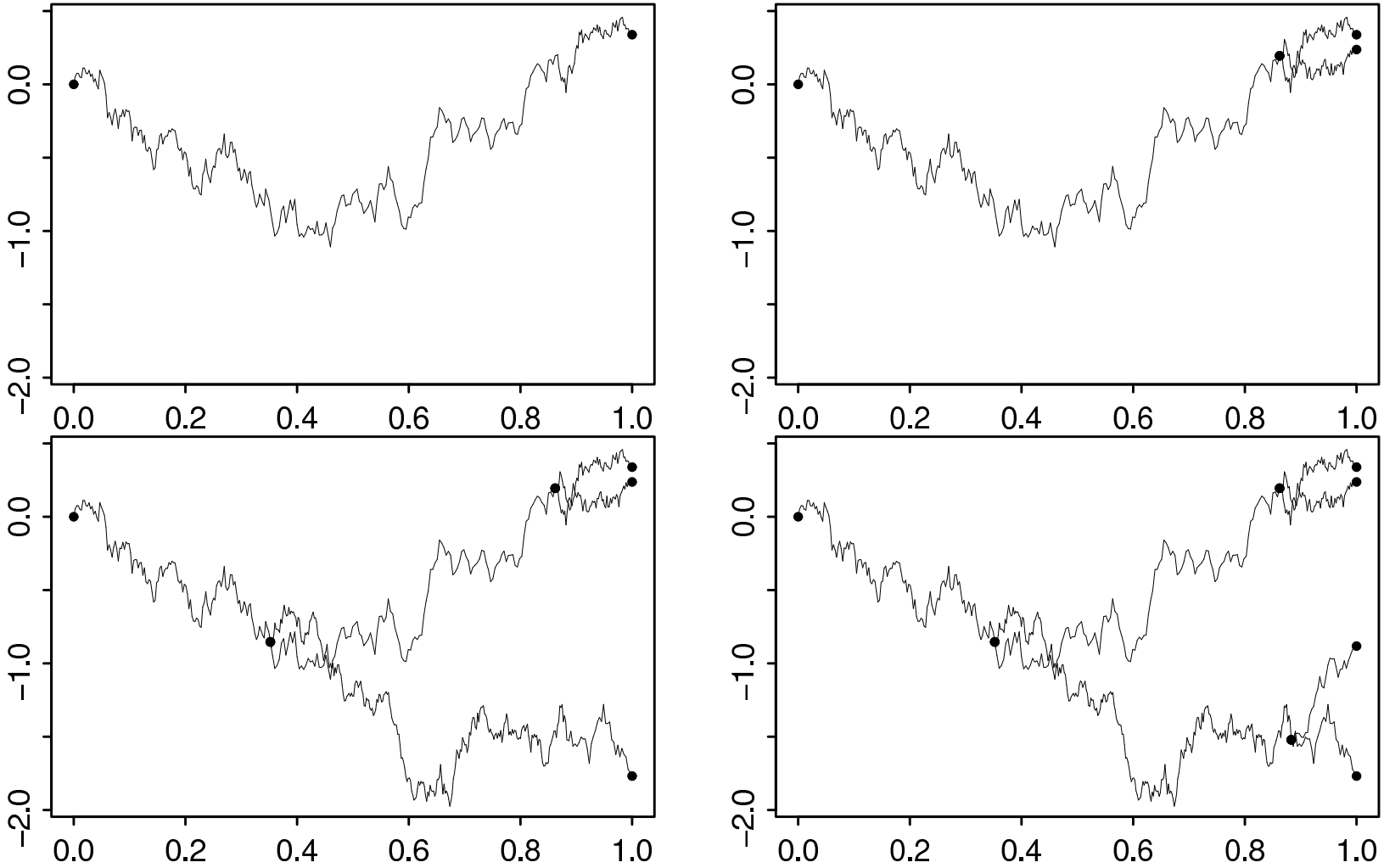
- $\theta_1(t) \sim$ Gaussian diffusion process starting at origin ($\theta_1(0) = 0$) for unit time.
- $\theta_2(t)$, also starts at the origin and follows θ_1 but diverges at some time τ_d, at which point the path followed by θ_2 becomes independent of θ_1's path.
- $a(t)$ is a divergence or hazard function, e.g. $a(t) = 1/(1-t)$. For small dt:

$$P(\theta \text{ diverges} \in (t, t+dt)) = \frac{a(t)dt}{m}$$

  where m is the number of previous points that have followed this path.
- If θ_i reaches a branch point between two paths, it picks a branch in proportion to the number of points that have followed that path.

Dirichlet Diffusion Trees (DFT)

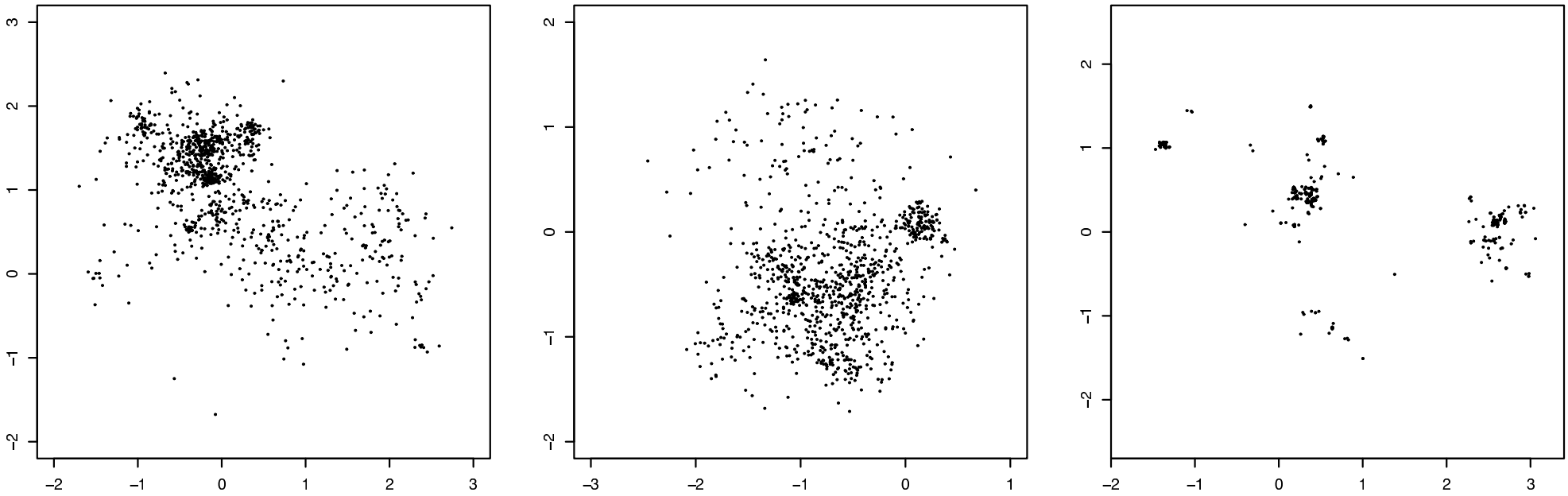
Generating from a DFT:



Figures from Neal, 2001.

Dirichlet Diffusion Trees (DFT)

Some samples from DFT priors:

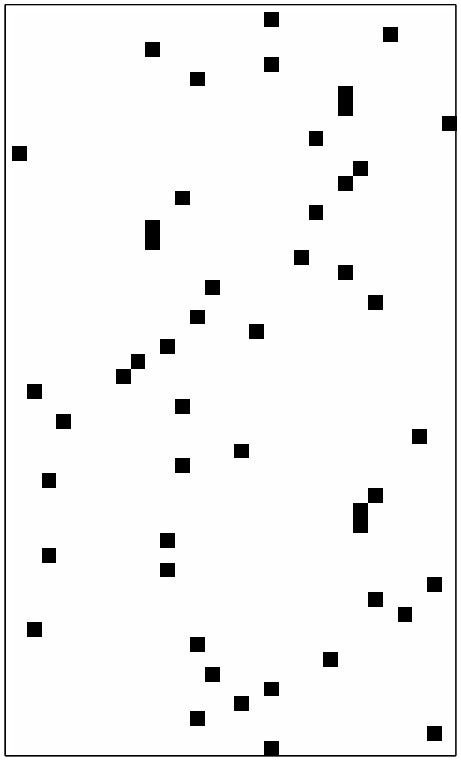


Figures from Neal, 2001.

Indian Buffet Processes (IBP)

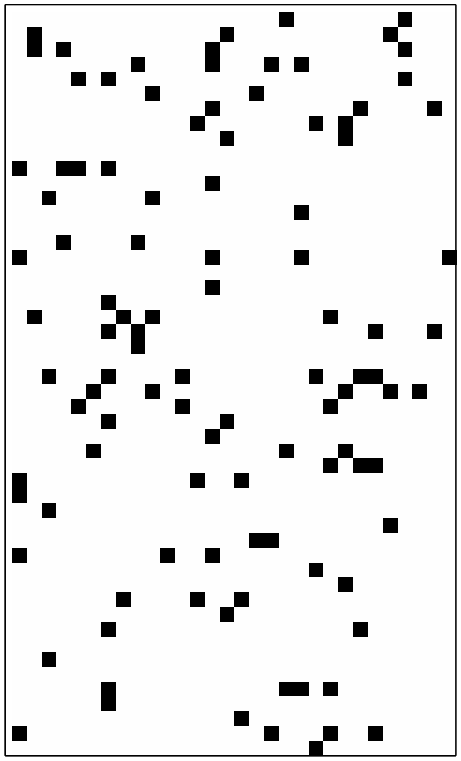
(Griffiths and Ghahramani, 2005)

Priors on Binary Matrices



- Rows are data points
- Columns are clusters
- We can think of CRPs as priors on infinite binary matrices...
- ...since each data point is assigned to one and only one cluster (class)...
- ...the rows sum to one.

More General Priors on Binary Matrices

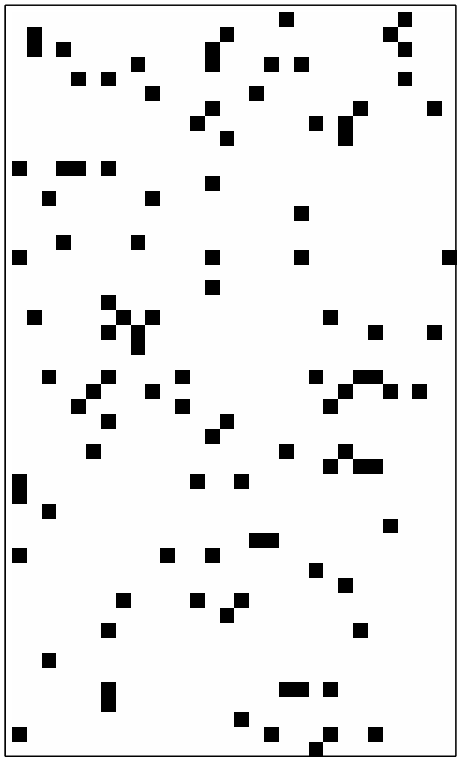


- Rows are data points
- Columns are features
- We can think of IBPs as priors on infinite binary matrices...
- ...where each data point can now have *multiple* features, so...
- ...the rows can sum to more than one.

Why?

- Many unsupervised learning algorithms can be thought of as modelling data in terms of hidden variables.
- Clustering algorithms represent data in terms of which cluster each data point belongs to.
- But clustering models are restrictive, they do not have distributed representations.
- Consider describing a person as "male", "married", "Democrat", "Red Sox fan"... these features may be unobserved (latent).
- The number of potential latent features for describing a person (or news story, gene, image, speech waveform, etc) is unlimited.

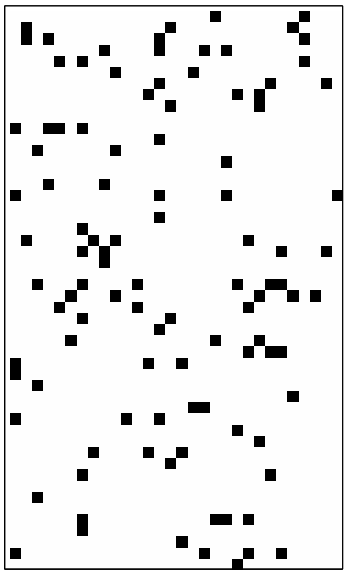
From finite to infinite binary matrices



$$z_{ik} \sim \text{Bernoulli}(\theta_k)$$
$$\theta_k \sim \text{Beta}(\alpha/K, 1)$$

- Note that $P(z_{ik} = 1|\alpha) = E(\theta_k) = \frac{\alpha/K}{\alpha/K+1}$, so as K grows larger the matrix gets sparser.
- So if $\mathbf{Z}$ is $N \times K$, the expected number of nonzero entries is $N\alpha/(1 + \alpha/K) < N\alpha$.
- Even in the $K \to \infty$ limit, the matrix is expected to have a finite number of non-zero entries.

From finite to infinite binary matrices



Just as with CRPs we can integrate out θ, leaving:

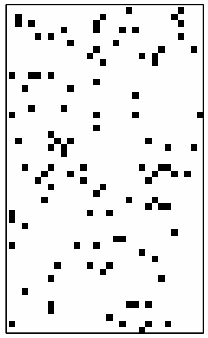
$$
\begin{aligned}
P(\mathbf{Z}|\alpha) &= \int P(\mathbf{Z}|\theta)P(\theta|\alpha)d\theta \\
&= \prod_k \frac{\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(\frac{\alpha}{K})} \frac{\Gamma(1 + \frac{\alpha}{K})}{\Gamma(N + 1 + \frac{\alpha}{K})}
\end{aligned}
$$

The conditional assignments are:

$$
\begin{aligned}
P(z_{ik} = 1|\mathbf{z}_{-i,k}) &= \int_0^1 P(z_{ik}|\theta_k)p(\theta_k|\mathbf{z}_{-i,k})\ d\theta_k \\
&= \frac{m_{-i,k} + \frac{\alpha}{K}}{N + \frac{\alpha}{K}},
\end{aligned}
$$

where $\mathbf{z}_{-i,k}$ is the set of assignments of all objects, not including i, for feature k, and $m_{-i,k}$ is the number of objects having feature k, not including i.

From finite to infinite binary matrices



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

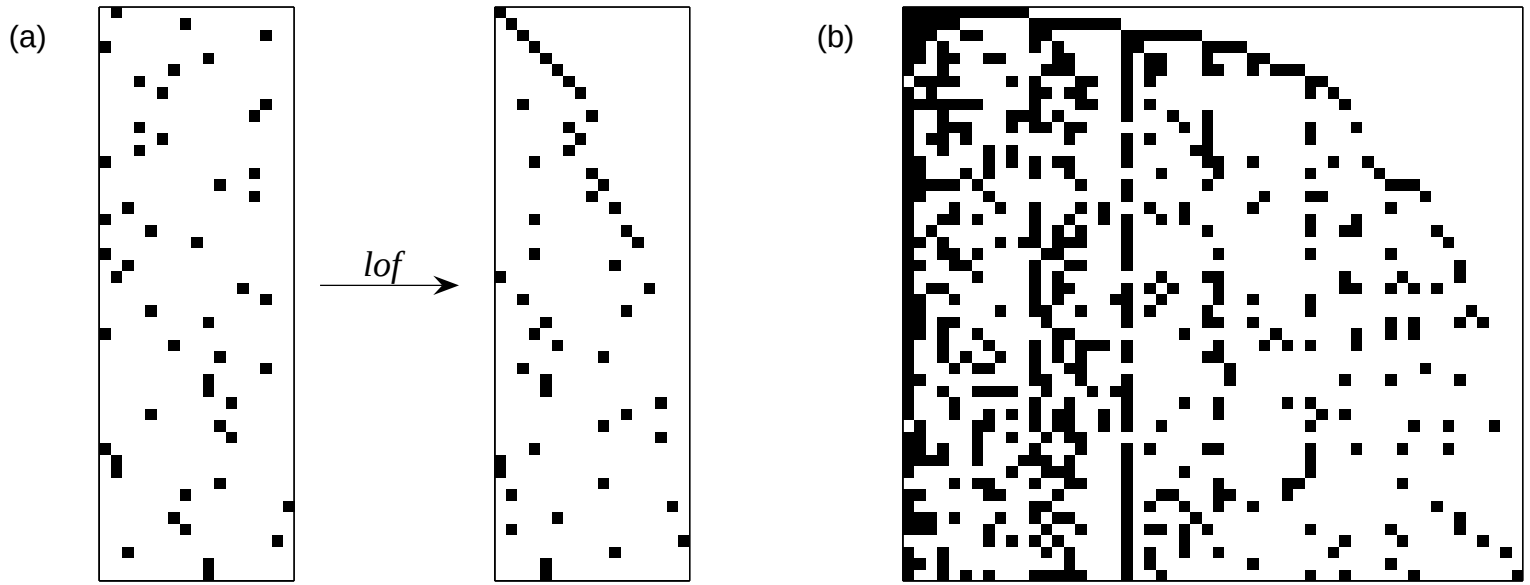
$$\lim_{K\to\infty} P(\mathbf{Z}|\alpha) \quad = \quad 0$$

However, if we consider equivalence classes of matrices in left-ordered form obtained by reordering the columns: $[\mathbf{Z}] = lof(\mathbf{Z})$ we get:

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

- K_+ is the number of features assigned (i.e. non-zero column sum).
- $H_N = \sum_{i=1}^{N} \frac{1}{i}$ is the Nth harmonic number.
- K_h are the number of features with history h (a technicality).
- This distribution is **exchangeable**, i.e. it is not affected by the ordering on objects. This is important for its use as a prior in settings where the objects have no natural ordering.

Binary matrices in left-ordered form



(a) The class matrix on the left is transformed into the class matrix on the right by the function $lof()$. The resulting left-ordered matrix was generated from a Chinese restaurant process (CRP) with $\alpha = 10$.

(b) A left-ordered feature matrix. This matrix was generated by the Indian buffet process (IBP) with $\alpha = 10$.

Indian buffet process

"Many Indian restaurants in London offer lunchtime buffets with an apparently infinite number of dishes"

- First customer starts at the left of the buffet, and takes a serving from each dish, stopping after a Poisson(α) number of dishes as her plate becomes overburdened.
- The ith customer moves along the buffet, sampling dishes in proportion to their popularity, serving himself with probability m_k/i, and trying a Poisson(α/i) number of new dishes.
- The customer-dish matrix is our feature matrix, $\mathbf{Z}$.

Conclusions

- We need flexible priors so that our Bayesian models are not based on unreasonable assumptions. Non-parametric models provide a way of defining flexible models.
- Many non-parametric models can be derived by starting from finite parametric models and taking the limit as the number of parameters goes to infinity.
- We've reviewed Gaussian processes, Dirichlet processes, and several other processes that can be used as a basis for defining non-parametric models.
- There are many open questions:
  - theoretical issues (e.g. consistency)
  - new models
  - applications
  - efficient samplers
  - approximate inference methods

`http://www.gatsby.ucl.ac.uk/~zoubin`

(for more resources, also to contact me
if interested in a PhD or postdoc)

Thanks for your patience!

Selected References

Gaussian Processes:

- O'Hagan, A. (1978). Curve Fitting and Optimal Design for Prediction (with discussion). **Journal of the Royal Statistical Society B**, 40(1):1-42.
- MacKay, D.J.C. (1997), Introduction to Gaussian Processes. http://www.inference.phy.cam.ac.uk/mackay/gpB.pdf
- Neal, R. M. (1998). Regression and classification using Gaussian process priors (with discussion). In Bernardo, J. M. et al., editors, **Bayesian statistics 6**, pages 475-501. Oxford University Press.
- Rasmussen, C.E and Williams, C.K.I. (to be published) **Gaussian processes for machine learning**

Dirichlet Processes, Chinese Restaurant Processes, and related work

- Ferguson, T. (1973), A Bayesian Analysis of Some Nonparametric Problems, **Annals of Statistics**, 1(2), pp. 209–230.
- Blackwell, D. and MacQueen, J. (1973), Ferguson Distributions via Polya Urn Schemes, **Annals of Statistics**, 1, pp. 353–355.
- Aldous, D. (1985), Exchangeability and Related Topics, in **Ecole d'Ete de Probabilites de Saint-Flour XIII 1983**, Springer, Berlin, pp. 1–198.
- Sethuraman, J. (1994), A Constructive Definition of Dirichlet Priors, **Statistica Sinica**, **4**:639–650.
- Pitman, J. and Yor, M. (1997) The two-parameter Poisson Dirichlet distribution derived from a stable subordinator. **Annals of Probability 25**: 855–900.

- Ishwaran, H. and Zarepour, M (2000) Markov chain Monte Carlo in approximate Dirichlet and beta two-parameter process hierarchical models. **Biomerika 87**(2): 371–390.

Polya Trees

- Ferguson, T.S. (1974) Prior Distributions on Spaces of Probability Measures. **Annals of Statistics**, 2:615-629.
- Lavine, M. (1992) Some aspects of Polya tree distributions for statistical modeling. **Annals of Statistics**, 20:1222-1235.

Hierarchical Dirichlet Processes and Infinite Hidden Markov Models

- Beal, M. J., Ghahramani, Z., and Rasmussen, C.E. (2002), The Infinite Hidden Markov Model, in T. G. Dietterich, S. Becker, and Z. Ghahramani (eds.) **Advances in Neural Information Processing Systems**, Cambridge, MA: MIT Press, vol. 14, pp. 577-584.
- Teh, Y.W, Jordan, M.I, Beal, M.J., and Blei, D.M. (2004) Hierarchical Dirichlet Processes. Technical Report, UC Berkeley.

Dirichlet Process Mixtures

- Antoniak, C.E. (1974) Mixtures of Dirichlet processes with applications to Bayesian non-parametric problems. **Annals of Statistics**, **2**:1152-1174.
- Escobar, M.D. and West, M. (1995) Bayesian density estimation and inference using mixtures. **J American Statistical Association**. **90**: 577-588.
- Neal, R.M. (2000). Markov chain sampling methods for Dirichlet process mixture models.**Journal of Computational and Graphical Statistics**, 9, 249–265.
- Rasmussen, C.E. (2000). The infinite gaussian mixture model. In **Advances in Neural Information Processing Systems 12**. Cambridge, MA: MIT Press.

- Blei, D.M. and Jordan, M.I. (2005) Variational methods for Dirichlet process mixtures. *Bayesian Analysis*.
- Minka, T.P. and Ghahramani, Z. (2003) Expectation propagation for infinite mixtures. **NIPS'03 Workshop on Nonparametric Bayesian Methods and Infinite Models**.
- Heller, K.A. and Ghahramani, Z. (2005) Bayesian Hierarchical Clustering. **Twenty Second International Conference on Machine Learning (ICML-2005)**

Dirichlet Diffusion Trees

- Neal, R.M. (2003) Density modeling and clustering using Dirichlet diffusion trees, in J. M. Bernardo, et al. (editors) **Bayesian Statistics 7**.

Indian Buffet Processes

- Griffiths, T. L. and Ghahramani, Z. (2005) Infinite latent feature models and the Indian Buffet Process. Gatsby Computational Neuroscience Unit Technical Report GCNU-TR 2005-001.

Other

- Müller, P. and Quintana, F.A. (2003) Nonparametric Bayesian Data Analysis.