The Indian buffet process

Tom Griffiths
Psychology and Cognitive Science
University of California, Berkeley

Joint work with Zoubin Ghahramani, Frank Wood and Dan Navarro
Identifying dimensionality

How many latent dimensions are expressed in our data?
Example 1: Identifying objects

- Can we learn to code images based on their contents?

- Want to infer a binary matrix encoding image features (one row per image, one column per object)

- How many objects appear in a collection of images?

(Griffiths & Ghahramani, 2006)
Example 2: Learning hidden causes

- Can we infer the hidden causes responsible for producing observed data?

- Want to infer adjacency matrix of a bipartite graph (one row per observed variable, one column per latent)

- How many hidden causes are responsible?

(Wood, Griffiths, & Ghahramani, 2006)
Example 3: Additive clustering

- What features do people associate with different stimuli?

- Additive clustering: infer features from human similarity judgments, assuming that \( s_{ij} \approx \sum_k w_k f_{ik} f_{jk} \) for \( i \neq j \)

- Want to infer a binary matrix identifying features (one row per stimulus, one column per feature)

- How many features should we consider?

(Navarro & Griffiths, 2005)
Perspectives on model selection

- Compare multiple models of different dimensionality
  - Bayes factors, cross-validation, etc.
  - Hard to apply to large model spaces

- Define a single model of unbounded dimensionality
  - Posterior on dimensionality via posterior on parameters
  - Allows dimensionality to grow with new data
  - Pursued in nonparametric Bayesian density estimation (e.g., Antoniak, 1974; Escobar & West, 1995)
Outline

• Nonparametric Bayes and the Chinese restaurant process
  – distribution on partitions

• Latent features and the Indian buffet process
  – distribution on binary matrices

• Applications and extensions
Mixture models

- Associate each datapoint $x_i$ with a latent class $z_i$

$$P(x_i) = \sum_{k=1}^{K} P(x_i | z_i = k)P(z_i = k)$$
Mixture models

• Associate each datapoint $x_i$ with a latent class $z_i$

$$P(x_i) = \sum_{k=1}^{K} P(x_i | z_i = k) P(z_i = k)$$

• e.g., Gaussian mixture model:

$$z_i \sim \text{Discrete}(\theta)$$

$$x_i | z_i, \beta \sim \text{Gaussian}(\beta_{z_i}, \sigma_X)$$

$$\theta \sim \text{Dirichlet}(\alpha)$$

$$\beta_k \sim \text{Gaussian}(0, \sigma_\beta)$$
Mixture models

• Associate each datapoint \( x_i \) with a latent class \( z_i \)

\[
P(x_i) = \sum_{k=1}^{K} P(x_i|z_i = k)P(z_i = k)
\]

• e.g., Gaussian mixture model:

\[
\begin{align*}
z_i & \sim \text{Discrete}(\theta) \\
x_i|z_i, \beta & \sim \text{Gaussian}(\beta_{z_i}, \sigma_X) \\
\theta & \sim \text{Dirichlet}(\alpha) \\
\beta_k & \sim \text{Gaussian}(0, \sigma_{\beta})
\end{align*}
\]

• How do we choose \( K \)?
Chinese restaurant process (CRP)

- Chinese restaurant with infinitely many infinite tables
- $N$ customers sit down
  - the first customer sits at the first table
  - the $i$th customer chooses a table at random

\[
P(\text{occupied table } k | \text{previous customers}) = \frac{m_k}{\alpha + i - 1}
\]
\[
P(\text{next unoccupied table} | \text{previous customers}) = \frac{\alpha}{\alpha + i - 1}
\]
Chinese restaurant process (CRP)

- Defines a distribution over partitions
- e.g., \((1 3 4 8) (2 5 10) (6) (7 9)\)
- Exchangeable distribution (Aldous, 1985; Pitman, 2002)

\[
P(\text{partition}) = \alpha^{K_+} \left( \prod_{k=1}^{K_+} (m_k - 1)! \right) \frac{\Gamma(\alpha)}{\Gamma(N + \alpha)}
\]
CRP and mixture modeling

Each table $k$
- corresponds to a mixture component
- associated with a parameter $\beta_k$ drawn from a prior

e.g., Gaussian CRP mixture model:

$$z \sim \text{CRP}(\alpha)$$
$$x_i|z_i, \beta \sim \text{Gaussian}(\beta_{z_i}, \sigma_X)$$
$$\beta_k \sim \text{Gaussian}(0, \sigma_\beta)$$
CRP and mixture modeling

- Given data $x$, posterior on $z$ gives
  - # of classes (# of occupied tables)
  - which data are assigned to each class
  - parameter for each class, $P(\beta_k | \text{data assigned to table } k)$

- Posterior inference via Gibbs sampling (e.g., Escobar & West, 1995; Neal, 1998)
Gibbs sampling

- Sequentially sample class assignments

\[ P(z_i|\mathbf{x}, \mathbf{z}_{-i}) \propto P(x_i|\mathbf{x}_{-i}, \mathbf{z})P(z_i|\mathbf{z}_{-i}) \]

- CRP provides \( P(z_i|\mathbf{z}_{-i}) \)

\[
P(z_i = \text{occupied class } k|\mathbf{z}_{-i}) = \frac{m_{k,-i}}{\alpha + N - 1}
\]

\[
P(z_i = \text{new class}|\mathbf{z}_{-i}) = \frac{\alpha}{\alpha + N - 1}
\]

- Allows datapoints to come from new classes

- Also split-merge algorithms (Jain & Neal, 2000; Dahl, 2003)
Beyond the CRP

• The CRP allows number of classes to be inferred

• But. . .
  – testing multiple models still feasible for mixtures
  – many kinds of data require other representations

• Can we apply a parallel strategy with other structures?
  – trees (Blei, Griffiths, Jordan, & Tenenbaum, 2004)
  – binary matrices (Griffiths & Ghahramani, 2005)
Latent feature representations

- Many statistical models represent objects with latent features
  - binary features
  - factorial structures
  - continuous dimensions
Latent feature representations

• Many statistical models represent objects with latent features
  – binary features
  – factorial structures
  – continuous dimensions

• A common assumption: sparsity
Latent feature representations

- Many statistical models represent objects with latent features
  - binary features
  - factorial structures
  - continuous dimensions

- A common assumption: sparsity

- Define a prior for sparse latent feature representations by defining a prior on (infinite column) binary matrices
• Binary features
Different feature representations

- Binary features
- Factorial features

<table>
<thead>
<tr>
<th>N objects</th>
<th>K features</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 3 0 0 4</td>
<td></td>
</tr>
<tr>
<td>5 0 3 0</td>
<td></td>
</tr>
<tr>
<td>0 1 4</td>
<td></td>
</tr>
<tr>
<td>2 0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>
### Different feature representations

- **Binary features**
- **Factorial features**
- **Continuous features**

<table>
<thead>
<tr>
<th>( N ) objects</th>
<th>( K ) features</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 0.9 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( -3.2 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( 0 )</td>
<td>( -0.3 )</td>
</tr>
<tr>
<td>( 0 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( 1.8 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( -0.1 )</td>
<td>( -2.8 )</td>
</tr>
</tbody>
</table>
Priors on binary matrices

- Start with priors on $N \times K$ matrices, take $K \to \infty$

- Two cases:
  - “class matrices”: one 1 per row
  - “feature matrices”: general binary matrices

- Two priors:
  - the Chinese restaurant process
  - the Indian buffet process
Class matrices

\[ z_i | \theta \sim \text{Discrete}(\theta) \]
\[ \theta \sim \text{Dirichlet}(\alpha/K) \]
Class matrices

\[ P(Z) = \int_{\Delta} \prod_{i=1}^{N} P(z_i|\theta) P(\theta) \, d\theta \]
Left-ordered form

- History $h$ of each class: binary column vector
- $lof$ orders columns by values of binary histories
\textit{lof} equivalence classes

- \textbf{X} and \textbf{Y} are \textit{lof} equivalent iff $\text{lof}(\mathbf{X}) = \text{lof}(\mathbf{Y})$

- Class matrices: \textit{lof} equivalence classes are partitions
$\lim_{K \to \infty} P([Z]) = \alpha^{K_+} \left( \prod_{k=1}^{K_+} (m_k - 1)! \right) \frac{\Gamma(\alpha)}{\Gamma(N + \alpha)}$

(see also Green & Richardson, 2001; Neal, 1992)
Feature matrices

• For general binary matrices

\[ z_{ik} \sim \text{Bernoulli}(\theta_k) \]
\[ \theta_k \sim \text{Beta}(\alpha/K, 1) \]
Feature matrices

• For general binary matrices

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

• For a finite matrix \( Z \)

\[
P(Z) = \int_0^1 \cdots \int_0^1 P(Z|\theta_1, \ldots, \theta_k) \prod_{k=1}^K P(\theta_k) \, d\theta_k
\]
Feature matrices

- For general binary matrices

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

- For a finite matrix \( Z \)

\[
P(Z) = \int_0^1 \cdots \int_0^1 P(Z|\theta_1, \ldots, \theta_K) \prod_{k=1}^K P(\theta_k) \, d\theta_k
\]

- Taking the limit as \( K \to \infty \) ...
Indian buffet process (IBP)

- Indian restaurant with infinitely many infinite dishes
- $N$ customers serve themselves
  - the first customer samples $\text{Poisson}(\alpha)$ dishes
  - the $i$th customer
    samples a previously sampled dish with probability $\frac{m_k}{i+1}$
    then samples $\text{Poisson}(\frac{\alpha}{i})$ new dishes
Indian buffet process (IBP)

- Indian restaurant with infinitely many infinite dishes
- \( N \) customers serve themselves
  - the first customer samples \( \text{Poisson}(\alpha) \) dishes
  - the \( i \)th customer samples a previously sampled dish with probability \( \frac{m_k}{i+1} \)
    then samples \( \text{Poisson}(\frac{\alpha}{i}) \) new dishes
Indian buffet process (IBP)

• Indian restaurant with infinitely many infinite dishes

• $N$ customers serve themselves
  – the first customer samples $\text{Poisson}(\alpha)$ dishes
  – the $i$th customer
    samples a previously sampled dish with probability $\frac{m_k}{i+1}$
    then samples $\text{Poisson}\left(\frac{\alpha}{i}\right)$ new dishes
Indian buffet process (IBP)

- Indian restaurant with infinitely many infinite dishes

- $N$ customers serve themselves
  - the first customer samples $\text{Poisson}(\alpha)$ dishes
  - the $i$th customer samples a previously sampled dish with probability $\frac{m_k}{i+1}$ then samples $\text{Poisson}\left(\frac{\alpha}{i}\right)$ new dishes
Indian buffet process (IBP)

- Indian restaurant with infinitely many infinite dishes
- $N$ customers serve themselves
  - the first customer samples $\text{Poisson}(\alpha)$ dishes
  - the $i$th customer
    samples a previously sampled dish with probability $\frac{m_k}{i+1}$
    then samples $\text{Poisson}(\frac{\alpha}{i})$ new dishes
Another generating process

- $lof$-equivalence classes can be represented as vectors of history counts

\[ h : ( 1 \ 2 \ \cdots \ 2^N - 1 ) \]

\[ K_h : ( K_1 \ K_2 \ \cdots \ K_{2^N - 1} ) \]

- Generate binary matrices by sampling $K_h$ directly

\[ K_h \sim \text{Poisson}(\alpha B(m_h, N - m_h + 1)) \]

where $B(r, s)$ is the beta function
Properties of the IBP

- Exchangeability of rows (or columns)

- Number of dishes sampled by each customer $\sim \text{Poisson}(\alpha)$

- Expected number of non-zero entries in $Z$ is $N\alpha$

- Total number of dishes $K^+ \sim \text{Poisson}(\alpha \sum_{i=1}^{N} \frac{1}{i})$
Example 1: Identifying objects

• Can we learn to code images based on their contents?

• Want to infer a binary matrix encoding image features (one row per image, one column per object)

• How many objects appear in a collection of images?

(Griffiths & Ghahramani, 2006)
A linear-Gaussian model

- Likelihood $P(X|Z)$ specified by

  \[ x_i \sim \text{Gaussian}(z_i A, \sigma_X I) \]
  \[ A \sim \text{Gaussian}(0, \sigma_A I) \]

- For $Z \sim \text{CRP}(\alpha)$, spherical Gaussian mixture model

- For $Z \sim \text{IBP}(\alpha)$, binary latent factor model

- Compute posterior distribution $P(Z|X)$
Gibbs sampling

- Sequentially sample feature assignments

\[ P(z_{ik}|X, z_{(-i)k}) \propto P(x_i|X_{-i}, Z)P(z_{ik}|z_{(-i)k}) \]

- IBP provides \( P(z_{ik}|z_{(-i)k}) \)
  - for old features, \( P(z_{ik}|z_{(-i)k}) = \frac{m_{k,-i}}{N} \)
  - prior on new features is Poisson\( (\frac{\alpha}{N}) \)
Coding for the presence of objects

- Photographs of everyday objects taken with a webcam
- 100 images, each $320 \times 240$ pixels
- Each image contained from 1 to 4 (fixed position) objects
Coding for the presence of objects

(Positive)  (Negative)  (Negative)  (Negative)
Example 2: Learning hidden causes

- Can we infer the hidden causes responsible for producing observed data?

- Want to infer adjacency matrix of a bipartite graph (one row per observed variable, one column per latent)

- How many hidden causes are responsible?

(Wood, Griffiths, & Ghahramani, 2006)
Priors on bipartite graphs

• $K \times N$ binary matrix $\Rightarrow$ bipartite graph
Priors on bipartite graphs

- $K \times N$ binary matrix $\Rightarrow$ bipartite graph
- Chinese restaurant process: one disease per symptom
Priors on bipartite graphs

- $K \times N$ binary matrix $\Rightarrow$ bipartite graph

- Chinese restaurant process: one disease per symptom

- Indian buffet process: multiple diseases per symptom
Binary matrix factorization

- With binary data and binary causes...

- Define likelihood $P(X|Z, Y)$ using “noisy-OR”

$$P(x_{ij} = 1|Y, Z) = 1 - (1 - \epsilon)(1 - \lambda)\sum_k z_{ik}y_{kj}$$
Results: Simulated data
Results: Stroke data

- Using data from the Mount Sinai Stroke Database...
  - presence of 38 “stroke signs” recorded for 50 patients
- Results roughly in accordance with recognized syndromes
Results: Stroke data
Example 3: Additive clustering

- What features do people associate with different stimuli?

- Additive clustering: infer features from human similarity judgments, assuming that
  \[ s_{ij} \approx \sum_k w_k f_{ik} f_{jk} \text{ for } (i \neq j) \]

- Want to infer a binary matrix identifying features (one row per stimulus, one column per feature)

- How many features should we consider?

(Navarro & Griffiths, 2005)
Evaluating inferred feature structures

- Use Gibbs sampling to draw from posterior distribution on feature matrices and weights $P(F, w | S)$

- A feature is defined by the stimuli to which it belongs
  - compute posterior probability feature exists
  - compute expected weight, given existence

- Compare with previously published solutions where available
Results: Numbers

- Similarity data from Shepard et al. (1975)

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>WEIGHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 4 8</td>
<td>0.444</td>
</tr>
<tr>
<td>0 1 2</td>
<td>0.345</td>
</tr>
<tr>
<td>3 6 9</td>
<td>0.331</td>
</tr>
<tr>
<td>6 7 8 9</td>
<td>0.291</td>
</tr>
<tr>
<td>2 3 4 5 6</td>
<td>0.255</td>
</tr>
<tr>
<td>1 3 5 7 9</td>
<td>0.216</td>
</tr>
<tr>
<td>1 2 3 4</td>
<td>0.214</td>
</tr>
<tr>
<td>4 5 6 7 8</td>
<td>0.172</td>
</tr>
<tr>
<td>additive constant</td>
<td>0.148</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>PROB.</th>
<th>WEIGHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 6 9</td>
<td>0.79</td>
<td>0.326</td>
</tr>
<tr>
<td>2 4 8</td>
<td>0.70</td>
<td>0.385</td>
</tr>
<tr>
<td>0 1 2</td>
<td>0.69</td>
<td>0.266</td>
</tr>
<tr>
<td>2 3 4 5 6</td>
<td>0.59</td>
<td>0.240</td>
</tr>
<tr>
<td>6 7 8 9</td>
<td>0.57</td>
<td>0.262</td>
</tr>
<tr>
<td>0 1 2 3 4</td>
<td>0.42</td>
<td>0.173</td>
</tr>
<tr>
<td>2 4 6 8</td>
<td>0.41</td>
<td>0.387</td>
</tr>
<tr>
<td>1 3 5 7 9</td>
<td>0.40</td>
<td>0.223</td>
</tr>
<tr>
<td>4 5 6 7 8</td>
<td>0.34</td>
<td>0.181</td>
</tr>
<tr>
<td>7 8 9</td>
<td>0.26</td>
<td>0.293</td>
</tr>
<tr>
<td>additive constant</td>
<td>1.00</td>
<td>0.075</td>
</tr>
</tbody>
</table>

- Model fits: (a) Tenenbaum (1996) \( r^2 = 0.909 \)  
  (b) Navarro & Griffiths (2005) \( r^2 = 0.974 \)
Results: Countries

- Similarity data from Navarro & Lee (2002)

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>Italy</th>
<th>Vietnam</th>
<th>Germany</th>
<th>Zimbabwe</th>
<th>Zimbabwe</th>
<th>Iraq</th>
<th>Zimbabwe</th>
<th>Philippines</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Germany</td>
<td>China</td>
<td>Russia</td>
<td>USA</td>
<td>Nigeria</td>
<td>Nigeria</td>
<td>Cuba</td>
<td>Libya</td>
<td>Nigeria</td>
</tr>
<tr>
<td>Spain</td>
<td>Japan</td>
<td>China</td>
<td>Japan</td>
<td>USA</td>
<td>USA</td>
<td>Iraq</td>
<td>Libya</td>
<td>Libya</td>
</tr>
<tr>
<td>Philippines</td>
<td>Indonesia</td>
<td>Japan</td>
<td>Japan</td>
<td>USA</td>
<td>USA</td>
<td>Iraq</td>
<td>Libya</td>
<td>Libya</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prob.</th>
<th>1.00</th>
<th>1.00</th>
<th>0.99</th>
<th>0.62</th>
<th>0.52</th>
<th>0.36</th>
<th>0.33</th>
<th>0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight</td>
<td>0.593</td>
<td>0.421</td>
<td>0.267</td>
<td>0.467</td>
<td>0.209</td>
<td>0.373</td>
<td>0.299</td>
<td>0.311</td>
</tr>
</tbody>
</table>

- Model fits: Navarro & Griffiths (2005) \( r^2 = 0.854 \)
Results: Letters

• Similarity data from Rothkopf (1957)

| FEATURE | M | I | C | D | P | E | E | K | B | C | N | L | G | O | R | F | H | X | G | J | W | T | Q | R | U |
| PROB.   | 1.00 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.98 | 0.92 | 0.686 | 0.341 | 0.623 | 0.321 | 0.465 | 0.653 | 0.322 | 0.427 | 0.226 | 0.225 |
| WEIGHT  | 0.686 | 0.341 | 0.623 | 0.321 | 0.465 | 0.653 | 0.322 | 0.427 | 0.226 | 0.225 |

• Model fits: Navarro & Griffiths (2005) \( (r^2 = 0.892) \)
Extensions

- Two-parameter process
  (Ghahramani, Griffiths, & Sollich, 2006)
The two-parameter IBP

• Use Beta($\alpha\beta/K, \beta$) instead of Beta($\alpha/K, 1$) in limiting construction
  – the first customer samples Poisson($\alpha$) dishes
  – the $i$th customer
    samples a previously sampled dish with probability $\frac{m_k}{i+\beta}$
    then samples Poisson($\frac{\alpha\beta}{i+\beta}$) new dishes

• Decouples density of matrix from its dimension
  – number of dishes sampled by each customer
    $\sim$ Poisson($\alpha$)
  – expected number of non-zero entries in $Z$ is $N\alpha$
  – total number of dishes $K^+ \sim$ Poisson($\alpha \sum_{i=1}^{N} \frac{\beta}{\beta+i-1}$)
Extensions

- Two-parameter process  
  (Ghahramani, Griffiths, & Sollich, 2006)

- Particle filter  
  (Wood & Griffiths, 2006)
Particle filtering

• For the CRP, let $z_{1:n} = (z_1, \ldots, z_n)$, etc.

$$P(z_{1:n} | x_{1:n}) \propto P(x_n | z_{1:n}, x_{1:n-1})P(z_n | z_{1:n-1})P(z_{1:n-1} | x_{1:n-1})$$

• Given a particle approximation to $P(z_{1:n-1} | x_{1:n-1})$
  – generate tables for the $n$th customer via the CRP
  – assign weights to particles using $P(x_n | z_{1:n}, x_{1:n-1})$

• For the IBP, let $Z_{1:n}$ be first $n$ rows of $Z$, etc.

$$P(Z_{1:n} | X_{1:n}) \propto P(x_n | Z_{1:n}, X_{1:n-1})P(z_n | Z_{1:n-1})P(Z_{1:n-1} | X_{1:n-1})$$
Extensions

- Two-parameter process
  (Ghahramani, Griffiths, & Sollich, 2006)

- Particle filter
  (Wood & Griffiths, 2006)

- Connections to beta processes
  (Thibault & Jordan, 2006)
Conclusion

• Strategy for model selection from nonparametric Bayes: prior over combinatorial structures of variable dimension

• For mixture models, use the Chinese restaurant process
  – exchangeable distribution over partitions

• Same strategy can be extended to other representations
  – binary matrices: Indian buffet process

• Provides flexible tools for formulating models with unbounded numbers of latent features