Bayesian Inference for Dirichlet-Multinomials and Dirichlet Processes

Mark Johnson

Macquarie University
Sydney, Australia

MLSS "Summer School"

Random variables and "distributed according to" notation

- A *probability distribution* F is a non-negative function from some set $\mathcal{X}$ whose values sum (integrate) to 1
- A random variable X is *distributed according* to a distribution F, or more simply, X *has distribution* F, written $X \sim F$, iff:

$$\mathrm{P}(X = x) \; = \; F(x) \text{ for all } x$$

  (This is for discrete RVs).
- You'll sometimes see the notion

$$X \mid Y \; \sim \; F$$

  which means "X is generated conditional on Y with distribution F" (where F usually depends on Y), i.e.,

$$\mathrm{P}(X \mid Y) \; = \; F(X \mid Y)$$

Outline

Bayes' rule

$$\mathrm{P}(\mathrm{Hypothesis} \mid \mathrm{Data}) = \frac{\mathrm{P}(\mathrm{Data} \mid \mathrm{Hypothesis})\, \mathrm{P}(\mathrm{Hypothesis})}{\mathrm{P}(\mathrm{Data})}$$

- Bayesian's use Bayes' Rule to *update beliefs in hypotheses in response to data*
- P(Hypothesis | Data) is the *posterior distribution*,
- P(Hypothesis) is the *prior distribution*,
- P(Data | Hypothesis) is the *likelihood*, and
- P(Data) is a normalising constant sometimes called the *evidence*

Computing the normalising constant

$$
\begin{aligned}
\mathrm{P}(\mathrm{Data}) &= \sum_{\mathrm{Hypothesis}' \in \mathcal{H}} \mathrm{P}(\mathrm{Data}, \mathrm{Hypothesis}') \\
&= \sum_{\mathrm{Hypothesis}' \in \mathcal{H}} \mathrm{P}(\mathrm{Data} \mid \mathrm{Hypothesis}')\mathrm{P}(\mathrm{Hypothesis}')
\end{aligned}
$$

- If set of hypotheses $\mathcal{H}$ is small, can calculate $\mathrm{P}(\mathrm{Data})$ by enumeration
- But *often these sums are intractable*

Bayesian belief updating

- Idea: treat posterior from last observation as the prior for next
- Consistency follows because likelihood factors
  - Suppose $\boldsymbol{d} = (d_1, d_2)$. Then the posterior of a hypothesis h is:

$$\begin{aligned} P(h \mid d_1, d_2) &\propto P(h)\, P(d_1, d_2 \mid h) \\ &= P(h)\, P(d_1 \mid h)\, P(d_2 \mid h, d_1) \\ &\propto \underbrace{P(h \mid d_1)}_{\text{updated prior}}\ \underbrace{P(d_2 \mid h, d_1)}_{\text{likelihood}} \end{aligned}$$

Discrete distributions

- A *discrete distribution* has a finite set of outcomes $1, \ldots, m$
- A discrete distribution is parameterized by a vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m)$, where $\mathrm{P}(X = j|\boldsymbol{\theta}) = \theta_j$ (so $\sum_{j=1}^{m} \theta_j = 1$)
  - Example: An m-sided die, where $\theta_j =$ prob. of face j
- Suppose $\boldsymbol{X} = (X_1, \ldots, X_n)$ and each $X_i|\boldsymbol{\theta} \sim \textsc{Discrete}(\boldsymbol{\theta})$. Then:

$$\mathrm{P}(\boldsymbol{X}|\boldsymbol{\theta}) = \prod_{i=1}^{n} \textsc{Discrete}(X_i; \boldsymbol{\theta}) = \prod_{j=1}^{m} \theta_j^{N_j}$$

  where N_j is the number of times j occurs in $\boldsymbol{X}$.
- Goal of next few slides: compute $\mathrm{P}(\boldsymbol{\theta}|\boldsymbol{X})$

Multinomial distributions

- Suppose $X_i \sim \textsc{Discrete}(\boldsymbol{\theta})$ for $i = 1, \ldots, n$, and N_j is the number of times j occurs in $\boldsymbol{X}$
- Then $\boldsymbol{N}|n, \boldsymbol{\theta} \sim \textsc{Multi}(\boldsymbol{\theta}, n)$, and

$$\mathrm{P}(\boldsymbol{N}|n, \boldsymbol{\theta}) = \frac{n!}{\prod_{j=1}^{m} N_j!} \prod_{j=1}^{m} \theta_j^{N_j}$$

  where $n!/\prod_{j=1}^{m} N_j!$ is the number of sequences of values with occurence counts $\boldsymbol{N}$
- The vector $\boldsymbol{N}$ is known as a *sufficient statistic* for $\boldsymbol{\theta}$ because it supplies as much information about $\boldsymbol{\theta}$ as the original sequence $\boldsymbol{X}$ does.

Dirichlet distributions

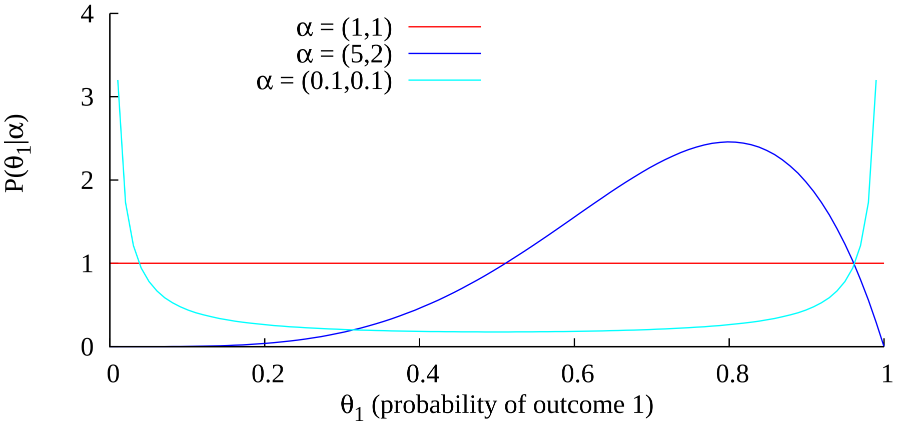
- *Dirichlet distributions* are probability distributions over multinomial parameter vectors
  - called *Beta distributions* when $m = 2$
- Parameterized by a vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$ where $\alpha_j > 0$ that determines the shape of the distribution

$$\begin{aligned} \textsc{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) &= \frac{1}{\mathrm{C}(\boldsymbol{\alpha})} \prod_{j=1}^{m} \theta_j^{\alpha_j - 1} \\ \mathrm{C}(\boldsymbol{\alpha}) &= \int_\Delta \prod_{j=1}^{m} \theta_j^{\alpha_j - 1} \, d\boldsymbol{\theta} = \frac{\prod_{j=1}^{m} \Gamma(\alpha_j)}{\Gamma(\sum_{j=1}^{m} \alpha_j)} \end{aligned}$$

- Γ is a generalization of the factorial function
- $\Gamma(k) = (k-1)!$ for positive integer k
- $\Gamma(x) = (x-1)\Gamma(x-1)$ for all x

Plots of the Dirichlet distribution

$$P(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) = \frac{\Gamma(\sum_{j=1}^{m} \alpha_j)}{\prod_{j=1}^{m} \Gamma(\alpha_j)} \prod_{k=1}^{m} \theta_k^{\alpha_k - 1}$$

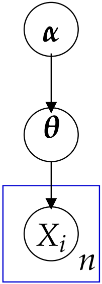


Dirichlet distributions as priors for $\boldsymbol{\theta}$

- Generative model:

$$
\begin{array}{rcccl}
\boldsymbol{\theta} & | & \boldsymbol{\alpha} & \sim & \text{DIR}(\boldsymbol{\alpha}) \\
X_i & | & \boldsymbol{\theta} & \sim & \text{DISCRETE}(\boldsymbol{\theta}), \quad i = 1, \ldots, n
\end{array}
$$

- We can depict this as a Bayes net using *plates*, which indicate *replication*



Inference for $\boldsymbol{\theta}$ with Dirichlet priors

- Data $\boldsymbol{X} = (X_1, \ldots, X_n)$ generated i.i.d. from $\text{DISCRETE}(\boldsymbol{\theta})$
- Prior is $\text{DIR}(\boldsymbol{\alpha})$. By Bayes Rule, posterior is:

$$
\begin{aligned}
\mathrm{P}(\boldsymbol{\theta}|\boldsymbol{X}) &\propto \mathrm{P}(\boldsymbol{X}|\boldsymbol{\theta})\ \mathrm{P}(\boldsymbol{\theta}) \\
&\propto \left(\prod_{j=1}^{m} \theta_j^{N_j}\right) \left(\prod_{j=1}^{m} \theta_j^{\alpha_j - 1}\right) \\
&= \prod_{j=1}^{m} \theta_j^{N_j + \alpha_j - 1}, \text{ so} \\
\mathrm{P}(\boldsymbol{\theta}|\boldsymbol{X}) &= \text{DIR}(\boldsymbol{N} + \boldsymbol{\alpha})
\end{aligned}
$$

- So if prior is Dirichlet with parameters $\boldsymbol{\alpha}$, posterior is Dirichlet with parameters $\boldsymbol{N} + \boldsymbol{\alpha}$

$\Rightarrow$ can regard Dirichlet parameters $\boldsymbol{\alpha}$ as “pseudo-counts” from “pseudo-data”

Conjugate priors

- If prior is $\text{DIR}(\boldsymbol{\alpha})$ and likelihood is i.i.d. $\text{DISCRETE}(\boldsymbol{\theta})$, then posterior is $\text{DIR}(\boldsymbol{N} + \boldsymbol{\alpha})$
  $\Rightarrow$ prior parameters $\boldsymbol{\alpha}$ specify "pseudo-observations"
- A class $\mathcal{C}$ of prior distributions $\mathrm{P}(H)$ is *conjugate* to a class of likelihood functions $\mathrm{P}(D|H)$ iff the posterior $\mathrm{P}(H|D)$ is also a member of $\mathcal{C}$
- In general, conjugate priors encode "pseudo-observations"
  - the difference between prior $\mathrm{P}(H)$ and posterior $\mathrm{P}(H|D)$ are the observations in D
  - but $\mathrm{P}(H|D)$ belongs to same family as $\mathrm{P}(H)$, and can serve as prior for inferences about more data D'

  $\Rightarrow$ must be possible to encode observations D using parameters of prior
- In general, the likelihood functions that have conjugate priors belong to the *exponential family*

Point estimates from Bayesian posteriors

- A "true" Bayesian prefers to use the full $\mathrm{P}(H|D)$, but sometimes we have to choose a "best" hypothesis
- The *Maximum a posteriori* (MAP) or *posterior mode* is

$$\widehat{H} = \underset{H}{argmax}\, \mathrm{P}(H|D) = \underset{H}{argmax}\, \mathrm{P}(D|H)\, \mathrm{P}(H)$$

- The *expected value* $\mathrm{E}_{\mathrm{P}}[X]$ of X under distribution P is:

$$\mathrm{E}_{\mathrm{P}}[X] = \int x\, \mathrm{P}(X = x)\, dx$$

The expected value is a kind of average, weighted by $\mathrm{P}(X)$. The *expected value* $\mathrm{E}[\boldsymbol{\theta}]$ of $\boldsymbol{\theta}$ is an estimate of $\boldsymbol{\theta}$.

The posterior mode of a Dirichlet

- The *Maximum a posteriori* (MAP) or *posterior mode* is

$$\widehat{H} \;=\; \underset{H}{argmax}\, \mathrm{P}(H|D) \;=\; \underset{H}{argmax}\, \mathrm{P}(D|H)\, \mathrm{P}(H)$$

- For Dirichlets with parameters $\boldsymbol{\alpha}$, the MAP estimate is:

$$\hat{\theta}_j \;=\; \frac{\alpha_j - 1}{\sum_{j'=1}^{m}(\alpha_{j'} - 1)}$$

so if the posterior is $\textsc{Dir}(\boldsymbol{N} + \boldsymbol{\alpha})$, the MAP estimate for $\boldsymbol{\theta}$ is:

$$\hat{\theta}_j \;=\; \frac{N_j + \alpha_j - 1}{n + \sum_{j'=1}^{m}(\alpha_{j'} - 1)}$$

- If $\boldsymbol{\alpha} = \mathbf{1}$ then $\hat{\theta}_j = N_j/n$, which is also the *maximum likelihood estimate* (MLE) for $\boldsymbol{\theta}$

The expected value of $\boldsymbol{\theta}$ for a Dirichlet

- The *expected value* $\mathrm{E}_{\mathrm{P}}[X]$ of X under distribution P is:

$$\mathrm{E}_{\mathrm{P}}[X] \;=\; \int x\, \mathrm{P}(X = x)\, dx$$

- For Dirichlets with parameters $\boldsymbol{\alpha}$, the expected value of θ_j is:

$$\mathrm{E}_{\mathrm{DIR}(\boldsymbol{\alpha})}[\theta_j] \;=\; \frac{\alpha_j}{\sum_{j'=1}^{m} \alpha_{j'}}$$

- Thus if the posterior is $\mathrm{DIR}(\boldsymbol{N} + \boldsymbol{\alpha})$, the expected value of θ_j is:

$$\mathrm{E}_{\mathrm{DIR}(\boldsymbol{N}+\boldsymbol{\alpha})}[\theta_j] \;=\; \frac{N_j + \alpha_j}{n + \sum_{j'=1}^{m} \alpha_{j'}}$$

- $\mathrm{E}[\boldsymbol{\theta}]$ *smooths* or *regularizes* the MLE by adding pseudo-counts $\boldsymbol{\alpha}$ to $\boldsymbol{N}$

Sampling from a Dirichlet

$$\boldsymbol{\theta} \mid \boldsymbol{\alpha} \sim \textsc{Dir}(\boldsymbol{\alpha}) \quad \text{iff} \quad \mathrm{P}(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \frac{1}{\mathrm{C}(\boldsymbol{\alpha})} \prod_{j=1}^{m} \theta_j^{\alpha_j - 1}, \text{ where:}$$

$$\mathrm{C}(\boldsymbol{\alpha}) = \frac{\prod_{j=1}^{m} \Gamma(\alpha_j)}{\Gamma(\sum_{j=1}^{m} \alpha_j)}$$

- There are several algorithms for producing samples from $\textsc{Dir}(\boldsymbol{\alpha})$. A simple one relies on the following result:
- If $V_k \sim \textsc{Gamma}(\alpha_k)$ and $\theta_k = V_k / (\sum_{k'=1}^{m} V_{k'})$, then $\boldsymbol{\theta} \sim \textsc{Dir}(\boldsymbol{\alpha})$
- This leads to the following algorithm for producing a sample $\boldsymbol{\theta}$ from $\textsc{Dir}(\boldsymbol{\alpha})$
  - Sample v_k from $\textsc{Gamma}(\alpha_k)$ for $k = 1, \ldots, m$
  - Set $\theta_k = v_k / (\sum_{k'=1}^{m} v_{k'})$

Posterior with Dirichlet priors

$$\begin{array}{rcl} \boldsymbol{\theta} \mid \boldsymbol{\alpha} & \sim & \text{Dir}(\boldsymbol{\alpha}) \\ X_i \mid \boldsymbol{\theta} & \sim & \text{Discrete}(\boldsymbol{\theta}), \quad i = 1, \ldots, n \end{array}$$

- *Integrate out* $\boldsymbol{\theta}$ to calculate posterior probability of $\boldsymbol{X}$

$$\begin{aligned} \mathrm{P}(\boldsymbol{X}|\boldsymbol{\alpha}) &= \int \mathrm{P}(\boldsymbol{X}, \boldsymbol{\theta}|\alpha)\, d\boldsymbol{\theta} = \int_\Delta \mathrm{P}(\boldsymbol{X}|\boldsymbol{\theta})\, \mathrm{P}(\boldsymbol{\theta}|\boldsymbol{\alpha})\, d\boldsymbol{\theta} \\ &= \int_\Delta \left(\prod_{j=1}^m \theta_j^{N_j} \right) \left(\frac{1}{\mathrm{C}(\boldsymbol{\alpha})} \prod_{j=1}^m \theta_j^{\alpha_j - 1} \right) d\boldsymbol{\theta} \\ &= \frac{1}{\mathrm{C}(\boldsymbol{\alpha})} \int_\Delta \prod_{j=1}^m \theta_j^{N_j + \alpha_j - 1}\, d\boldsymbol{\theta} \\ &= \frac{\mathrm{C}(\boldsymbol{N} + \boldsymbol{\alpha})}{\mathrm{C}(\boldsymbol{\alpha})}, \text{ where } \mathrm{C}(\boldsymbol{\alpha}) = \frac{\prod_{j=1}^m \Gamma(\alpha_j)}{\Gamma(\sum_{j=1}^m \alpha_j)} \end{aligned}$$

- *Collapsed Gibbs samplers* and the *Chinese Restaurant Process* rely on this result

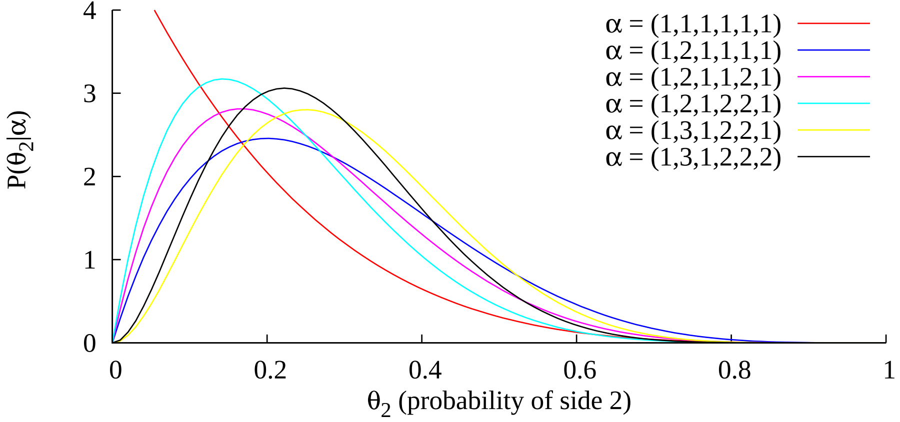
Predictive distribution for Dirichlet-Multinomial

- The *predictive distribution* is the distribution of observation X_{n+1} given observations $\boldsymbol{X} = (X_1, \ldots, X_n)$ and prior $\mathrm{DIR}(\boldsymbol{\alpha})$

$$\begin{aligned}
\mathrm{P}(X_{n+1} = k \mid \boldsymbol{X}, \boldsymbol{\alpha}) &= \int_\Delta \mathrm{P}(X_{n+1} = k \mid \boldsymbol{\theta})\, \mathrm{P}(\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{\alpha})\, d\boldsymbol{\theta} \\
&= \int_\Delta \theta_k \, \mathrm{DIR}(\boldsymbol{\theta} \mid \boldsymbol{N} + \boldsymbol{\alpha})\, d\boldsymbol{\theta} \\
&= \frac{N_k + \alpha_k}{\sum_{j=1}^m N_j + \alpha_j}
\end{aligned}$$

Example: rolling a die

- Data $d = (2, 5, 4, 2, 6)$



Inference in complex models

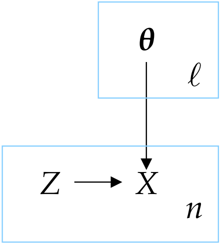
- If the model is simple enough we can calculate the posterior exactly (conjugate priors)
- When the model is more complicated, we can only approximate the posterior
- *Variational Bayes* calculate the function closest to the posterior within a class of functions
- *Sampling algorithms* produce samples from the posterior distribution
  - *Markov chain Monte Carlo algorithms* (MCMC) use a Markov chain to produce samples
  - A *Gibbs sampler* is a particular MCMC algorithm
- *Particle filters* are a kind of *on-line* sampling algorithm (on-line algorithms only make one pass through the data)

Outline

Mixture models

- Observations X_i are a *mixture* of ℓ source distributions $F(\boldsymbol{\theta}_k), k = 1, \ldots, \ell$
- The value of Z_i specifies which source distribution is used to generate X_i (Z is like a switch)
- If $Z_i = k$, then $X_i \sim F(\boldsymbol{\theta}_k)$
- Here we assume the Z_i are not observed, i.e., *hidden*

$$X_i \mid Z_i, \boldsymbol{\theta} \sim F(\boldsymbol{\theta}_{Z_i}) \quad i = 1, \ldots, n$$

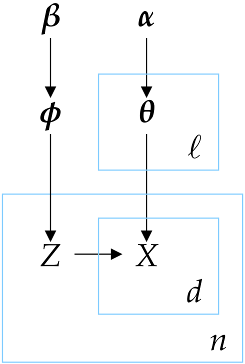


Applications of mixture models

- *Blind source separation*: data X_i come from ℓ different sources
  - Which X_i come from which source?
    (Z_i specifies the source of X_i)
  - What are the sources?
    ($\boldsymbol{\theta}_k$ specifies properties of source k)
- X_i could be a document and Z_i the topic of X_i
- X_i could be an image and Z_i the object(s) in X_i
- X_i could be a person's actions and Z_i the "cause" of X_i
- These are *unsupervised learning problems*, which are kinds of *clustering problems*
- In a Bayesian setting, compute posterior $\mathrm{P}(\boldsymbol{Z}, \boldsymbol{\theta} | \boldsymbol{X})$
  But how can we compute this?

Dirichlet Multinomial mixtures

$$
\begin{array}{rcll}
\boldsymbol{\phi} \mid \boldsymbol{\beta} & \sim & \textsc{Dir}(\boldsymbol{\beta}) & \\
Z_i \mid \boldsymbol{\phi} & \sim & \textsc{Discrete}(\boldsymbol{\phi}) & i = 1, \ldots, n \\
\boldsymbol{\theta}_k \mid \boldsymbol{\alpha} & \sim & \textsc{Dir}(\boldsymbol{\alpha}) & k = 1, \ldots, \ell \\
X_{i,j} \mid Z_i, \boldsymbol{\theta} & \sim & \textsc{Discrete}(\boldsymbol{\theta}_{Z_i}) & i = 1, \ldots, n; j = 1, \ldots, d_i
\end{array}
$$



- Z_i is generated from a multinomial $\boldsymbol{\phi}$
- Dirichlet priors on $\boldsymbol{\phi}$ and $\boldsymbol{\theta}_k$
- Easy to modify this framework for other applications
- Why does each observation $\boldsymbol{X}_i$ consist of d_i elements?
- What effect do the priors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ have?

Outline

Why sample?

- Setup: Bayes net has variables $\boldsymbol{X}$, whose value $\boldsymbol{x}$ we observe, and variables $\boldsymbol{Y}$, whose value we don't know
  - $\boldsymbol{Y}$ includes any *parameters* we want to estimate, such as $\boldsymbol{\theta}$
- Goal: compute the *expected value* of some function f:

$$\mathrm{E}[f|\boldsymbol{X}=\boldsymbol{x}] \;=\; \sum_{\boldsymbol{y}} f(\boldsymbol{x},\boldsymbol{y})\,\mathrm{P}(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{X}=\boldsymbol{x})$$

  - E.g., $f(\boldsymbol{x},\boldsymbol{y}) = 1$ if x_1 and x_2 are both generated from same hidden state, and 0 otherwise
- In what follows, everything is conditioned on $\boldsymbol{X}=\boldsymbol{x}$, so take $\mathrm{P}(\boldsymbol{Y})$ to mean $\mathrm{P}(\boldsymbol{Y}|\boldsymbol{X}=\boldsymbol{x})$
- Suppose we can produce n samples $\boldsymbol{y}^{(t)}$, where $\boldsymbol{Y}^{(t)} \sim \mathrm{P}(\boldsymbol{Y})$. Then we can estimate:

$$\mathrm{E}[f|\boldsymbol{X}=\boldsymbol{x}] \;=\; \frac{1}{n}\sum_{t=1}^{n} f(\boldsymbol{x},\boldsymbol{y}^{(t)})$$

Markov chains

- A (first-order) *Markov chain* is a distribution over random variables $S^{(0)}, \ldots, S^{(n)}$ all ranging over the same *state space* $\mathcal{S}$, where:

$$\mathrm{P}(S^{(0)}, \ldots, S^{(n)}) \quad = \quad \mathrm{P}(S^{(0)}) \prod_{t=0}^{n-1} \mathrm{P}(S^{(t+1)} | S^{(t)})$$

  $S^{(t+1)}$ is *conditionally independent* of $S^{(0)}, \ldots, S^{(t-1)}$ given $S^{(t)}$
- A Markov chain in *homogeneous* or *time-invariant* iff:

$$\mathrm{P}(S^{(t+1)} = s' | S^{(t)} = s) \quad = \quad P_{s',s} \quad \text{for all } t, s, s'$$

  The matrix P is called the *transition probability matrix* of the Markov chain
- If $\mathrm{P}(S^{(t)} = s) = \pi_s^{(t)}$ (i.e., $\pi^{(t)}$ is a vector of state probabilities at time t) then:
  - $\pi^{(t+1)} \; = \; P \, \pi^{(t)}$
  - $\pi^{(t)} \; = \; P^t \, \pi^{(0)}$

Ergodicity

- A Markov chain with tpm P is *ergodic* iff there is a positive integer m s.t. all elements of P^m are positive (i.e., there is an m-step path between any two states)
- Informally, an ergodic Markov chain "forgets" its past states
- Theorem: For each homogeneous ergodic Markov chain with tpm P there is a *unique limiting distribution* D_P, i.e., as n approaches infinity, the distribution of S_n converges on D_P
- D_P is called the *stationary distribution* of the Markov chain
- Let π be the vector representation of D_P, i.e., $D_P(y) = \pi_y$. Then:

$$\begin{aligned} \pi &= P\,\pi, && \text{and} \\ \pi &= \lim_{n\to\infty} P^n \pi^{(0)} && \text{for every initial distribution } \pi^{(0)} \end{aligned}$$

Using a Markov chain for inference of $\mathrm{P}(Y)$

- Set the state space $\mathcal{S}$ of the Markov chain to the range of $\boldsymbol{Y}$ ($\mathcal{S}$ may be *astronomically large*)
- Find a tpm P such that $\mathrm{P}(\boldsymbol{Y}) \sim D_P$
- "Run" the Markov chain, i.e.,
  - Pick $\boldsymbol{y}^{(0)}$ somehow
  - For $t = 0, \ldots, n-1$:
    - sample $\boldsymbol{y}^{(t+1)}$ from $\mathrm{P}(\boldsymbol{Y}^{(t+1)} | \boldsymbol{Y}^{(t)} = \boldsymbol{y}^{(t)})$, i.e., from $P_{\cdot,\boldsymbol{y}^{(t)}}$
  - After discarding the first *burn-in* samples, use remaining samples to calculate statistics
- *WARNING:* in general the samples $\boldsymbol{y}^{(t)}$ are *not independent*

Outline

The Gibbs sampler

- The Gibbs sampler is useful when:
  - $\boldsymbol{Y}$ is multivariate, i.e., $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$, and
  - easy to sample from $\mathrm{P}(Y_j | \boldsymbol{Y}_{-j})$
- The *Gibbs sampler* for $\mathrm{P}(Y)$ is the tpm $P = \prod_{j=1}^{m} P^{(j)}$, where:

$$P^{(j)}_{\boldsymbol{y}',\boldsymbol{y}} = \begin{cases} 0 & \text{if } \boldsymbol{y}'_{-j} \neq \boldsymbol{y}_{-j} \\ \mathrm{P}(Y_j = y'_j | \boldsymbol{Y}_{-j} = \boldsymbol{y}_{-j}) & \text{if } \boldsymbol{y}'_{-j} = \boldsymbol{y}_{-j} \end{cases}$$

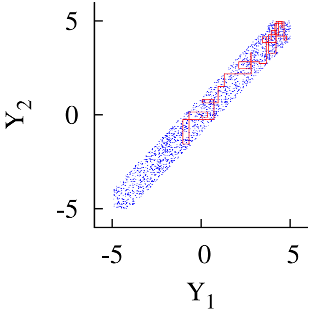
- Informally, the Gibbs sampler cycles through each of the variables Y_j, replacing the current value y_j with a sample from $\mathrm{P}(Y_j | \boldsymbol{Y}_{-j} = \boldsymbol{y}_{-j})$
- There are *sequential scan* and *random scan* variants of Gibbs sampling

A simple example of Gibbs sampling

$$P(Y_1, Y_2) = \begin{cases} c & \text{if } |Y_1| < 5, |Y_2| < 5 \text{ and } |Y_1 - Y_2| < 1 \\ 0 & \text{otherwise} \end{cases}$$

- The Gibbs sampler for $P(Y_1, Y_2)$ samples repeatedly from:

$$\begin{aligned} P(Y_2|Y_1) &= \textsc{Uniform}(\max(-5, Y_1 - 1), \min(5, Y_1 + 1)) \\ P(Y_1|Y_2) &= \textsc{Uniform}(\max(-5, Y_2 - 1), \min(5, Y_2 + 1)) \end{aligned}$$



Sample run

Y_1	Y_2
0	0
0	-0.119
0.363	-0.119
0.363	0.146
-0.681	0.146
-0.681	-1.551

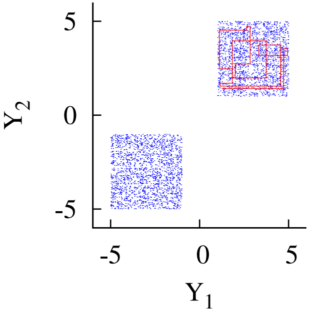
A non-ergodic Gibbs sampler

$$P(Y_1, Y_2) = \begin{cases} c & \text{if } 1 < Y_1, Y_2 < 5 \text{ or } -5 < Y_1, Y_2 < -1 \\ 0 & \text{otherwise} \end{cases}$$

- The Gibbs sampler for $P(Y_1, Y_2)$, initialized at (2,2), samples repeatedly from:

$$P(Y_2|Y_1) = \text{UNIFORM}(1,5)$$
$$P(Y_1|Y_2) = \text{UNIFORM}(1,5)$$

I.e., *never visits the negative values of Y_1, Y_2*



Sample run

Y_1	Y_2
2	2
2	2.72
2.84	2.72
2.84	4.71
2.63	4.71
2.63	4.52
1.11	4.52

Why does the Gibbs sampler work?

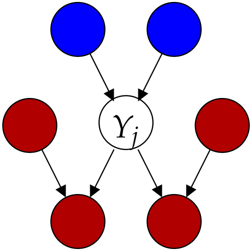
- The Gibbs sampler tpm is $P = \prod_{j=1}^{m} P^{(j)}$, where $P^{(j)}$ replaces y_j with a sample from $\mathrm{P}(Y_j | \boldsymbol{Y}_{-j} = \boldsymbol{y}_{-j})$ to produce y'
- But if $\boldsymbol{y}$ is a sample from $\mathrm{P}(\boldsymbol{Y})$, then so is $\boldsymbol{y}'$, since $\boldsymbol{y}'$ differs from $\boldsymbol{y}$ only by replacing y_j with a sample from $\mathrm{P}(Y_j | \boldsymbol{Y}_{-j} = \boldsymbol{y}_{-j})$
- Since $P^{(j)}$ maps samples from $\mathrm{P}(\boldsymbol{Y})$ to samples from $\mathrm{P}(\boldsymbol{Y})$, so does P

$\Rightarrow$ $\mathrm{P}(\boldsymbol{Y})$ is a stationary distribution for P

- If P is ergodic, then $\mathrm{P}(\boldsymbol{Y})$ is the unique stationary distribution for P, i.e., the sampler converges to $\mathrm{P}(\boldsymbol{Y})$

Gibbs sampling with Bayes nets

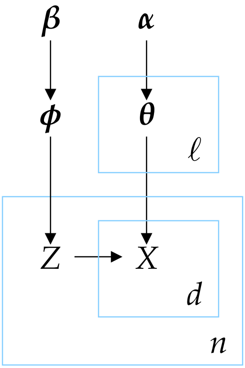
- Gibbs sampler: update y_j with sample from $\mathrm{P}(Y_j|\boldsymbol{Y}_{-j}) \propto \mathrm{P}(Y_j, \boldsymbol{Y}_{-j})$
- Only need to evaluate terms that depend on Y_j in Bayes net factorization
  - Y_j appears once in a term $\mathrm{P}(Y_j|\boldsymbol{Y}_{\mathsf{Pa}_j})$
  - Y_j can appear multiple times in terms $\mathrm{P}(Y_k|\ldots, Y_j, \ldots)$
- In graphical terms, need to know value of:
  - Y_js parents
  - Y_js children, and their other parents



Outline

Dirichlet-Multinomial mixtures

$$
\begin{array}{rcllll}
\boldsymbol{\phi} & | & \boldsymbol{\beta} & \sim & \textsc{Dir}(\boldsymbol{\beta}) & \\
Z_i & | & \boldsymbol{\phi} & \sim & \textsc{Discrete}(\boldsymbol{\phi}) & i = 1, \ldots, n \\
\boldsymbol{\theta}_k & | & \boldsymbol{\alpha} & \sim & \textsc{Dir}(\boldsymbol{\alpha}) & k = 1, \ldots, \ell \\
X_{i,j} & | & Z_i, \boldsymbol{\theta} & \sim & \textsc{Discrete}(\boldsymbol{\theta}_{Z_i}) & i = 1, \ldots, n; j = 1, \ldots, d_i
\end{array}
$$

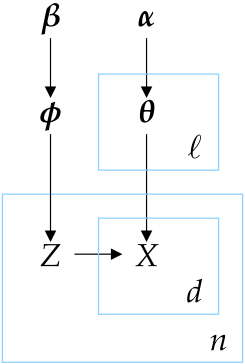


$$
\begin{aligned}
&\mathrm{P}(\boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{X} | \boldsymbol{\alpha}, \boldsymbol{\beta}) \\
&= \frac{1}{C(\boldsymbol{\beta})} \prod_{k=1}^{\ell} \left(\phi_k^{\beta_k - 1 + N_k(\boldsymbol{Z})} \frac{1}{C(\boldsymbol{\alpha})} \prod_{j=1}^{m} \theta_{k,j}^{\alpha_j - 1 + \sum_{i:Z_i = k} N_j(\boldsymbol{X}_i)} \right)
\end{aligned}
$$

$$
\text{where} \quad C(\alpha) = \frac{\prod_{j=1}^{m} \Gamma(\alpha_j)}{\Gamma(\sum_{j=1}^{m} \alpha_j)}
$$

Gibbs sampling for D-M mixtures

$$
\begin{array}{rcllll}
\boldsymbol{\phi} & | & \boldsymbol{\beta} & \sim & \textsc{Dir}(\boldsymbol{\beta}) & \\
Z_i & | & \boldsymbol{\phi} & \sim & \textsc{Discrete}(\boldsymbol{\phi}) & i = 1, \ldots, n \\
\boldsymbol{\theta}_k & | & \boldsymbol{\alpha} & \sim & \textsc{Dir}(\boldsymbol{\alpha}) & k = 1, \ldots, \ell \\
X_{i,j} & | & Z_i, \boldsymbol{\theta} & \sim & \textsc{Discrete}(\boldsymbol{\theta}_{Z_i}) & i = 1, \ldots, n; j = 1, \ldots, d_i
\end{array}
$$

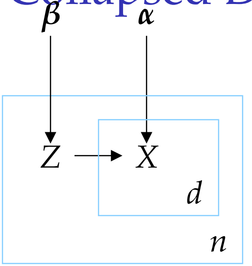


$$\mathrm{P}(\boldsymbol{\phi} | \boldsymbol{Z}, \boldsymbol{\beta}) \quad = \quad \textsc{Dir}(\boldsymbol{\phi}; \boldsymbol{\beta} + \boldsymbol{N}(\boldsymbol{Z}))$$

$$\mathrm{P}(Z_i = k | \boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{X}_i) \;\propto\; \phi_k \prod_{j=1}^{m} \theta_{k,j}^{N_j(\boldsymbol{X}_i)}$$

$$\mathrm{P}(\boldsymbol{\theta}_k | \boldsymbol{\alpha}, \boldsymbol{X}, \boldsymbol{Z}) \quad = \quad \textsc{Dir}(\boldsymbol{\theta}_k; \boldsymbol{\alpha} + \textstyle\sum_{i:Z_i=k} \boldsymbol{N}(\boldsymbol{X}_i))$$

Collapsed Dirichlet Multinomial mixtures



$$\mathrm{P}(\boldsymbol{Z}|\boldsymbol{\beta}) = \frac{C(\boldsymbol{N}(\boldsymbol{Z})+\boldsymbol{\beta})}{C(\boldsymbol{\beta})}$$

$$\mathrm{P}(\boldsymbol{X}|\boldsymbol{\alpha},\boldsymbol{Z}) = \prod_{k=1}^{\ell} \frac{C(\boldsymbol{\alpha}+\sum_{i:Z_i=k}\boldsymbol{N}(\boldsymbol{X}_i))}{C(\boldsymbol{\alpha})}, \text{ so}$$

$$\mathrm{P}(Z_i = k|\boldsymbol{Z}_{-i},\boldsymbol{\alpha},\boldsymbol{\beta}) \propto \frac{N_k(\boldsymbol{Z}_{-i})+\beta_k}{n-1+\beta_\bullet} \frac{C(\boldsymbol{\alpha}+\sum_{i'\neq i:Z_{i'}=k}\boldsymbol{N}(\boldsymbol{X}_{i'})+\boldsymbol{N}(\boldsymbol{X}_i))}{C(\boldsymbol{\alpha}+\sum_{i'\neq i:Z_{i'}=k}\boldsymbol{N}(\boldsymbol{X}_{i'}))}$$

- $\mathrm{P}(Z_i = k|\boldsymbol{Z}_{-i},\boldsymbol{\alpha},\boldsymbol{\beta})$ is proportional to the prob. of generating:
  - $Z_i = k$, given the other $\boldsymbol{Z}_{-i}$, and
  - $\boldsymbol{X}_i$ in cluster k, given $\boldsymbol{X}_{-i}$ and $\boldsymbol{Z}_{-i}$

Gibbs sampling for Dirichlet multinomial mixtures

- Each $\boldsymbol{X}_i$ could be generated from one of several Dirichlet multinomials
- The variable Z_i indicates the source for $\boldsymbol{X}_i$
- The *uncollapsed sampler* samples $\boldsymbol{Z}, \boldsymbol{\theta}$ and $\boldsymbol{\phi}$
- The *collapsed sampler* integrates out $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ and just samples $\boldsymbol{Z}$
- Collapsed samplers often (but not always) converge faster than uncollapsed samplers
- Collapsed samplers are usually‘ easier to implement

Outline

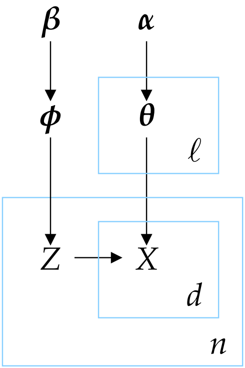
Topic modeling of child-directed speech

- Data: Adam, Eve and Sarah's mothers' child-directed utterances

  I like it .
  why don't you read Shadow yourself ?
  that's a terribly small horse for you to ride .
  why don't you look at some of the toys in the basket .
  want to ?
  do you want to see what I have ?
  what is that ?
  not in your mouth .

- 59,959 utterances, composed of 337,751 words

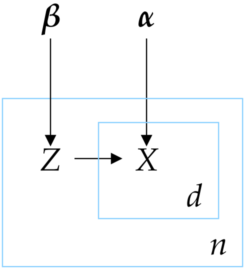
Uncollapsed Gibbs sampler for topic model



- Data consists of "documents" $\boldsymbol{X}_i$
- Each $\boldsymbol{X}_i$ is a sequence of "words" $X_{i,j}$
- Initialize by *randomly* assign each document $\boldsymbol{X}_i$ to a topic Z_i
- Repeat the following:
  - Replace $\boldsymbol{\phi}$ with a sample from a Dirichlet with parameters $\boldsymbol{\beta} + \boldsymbol{N}(\boldsymbol{Z})$
  - For each topic k, replace $\boldsymbol{\theta}_k$ with a sample from a Dirichlet with parameters $\boldsymbol{\alpha} + \sum_{i:Z_i=k} \boldsymbol{N}(\boldsymbol{X}_i))$
  - For each document i, replace Z_i with a sample from

    $$\mathrm{P}(Z_i = k | \boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{X}_i) \;\propto\; \phi_k \; \prod_{j=1}^{m} \theta_{k,j}^{N_j(\boldsymbol{X}_i)}$$

Collapsed Gibbs sampler for topic model



- Initialize by *randomly* assign each document $\boldsymbol{X}_i$ to a topic Z_i
- Repeat the following:
  - For each document i in $1, \ldots, n$ (in random order):
    - Replace Z_i with a random sample from $\mathrm{P}(Z_i | \boldsymbol{Z}_{-i}, \boldsymbol{\alpha}, \boldsymbol{\beta})$

$$\mathrm{P}(Z_i = k | \boldsymbol{Z}_{-i}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto \frac{N_k(\boldsymbol{Z}_{-i}) + \beta_k}{n - 1 + \beta_\bullet} \frac{\mathrm{C}(\boldsymbol{\alpha} + \sum_{i' \neq i : Z_{i'} = k} \boldsymbol{N}(\boldsymbol{X}_{i'}) + \boldsymbol{N}(\boldsymbol{X}_i))}{\mathrm{C}(\boldsymbol{\alpha} + \sum_{i' \neq i : Z_{i'} = k} \boldsymbol{N}(\boldsymbol{X}_{i'}))}$$

Topics assigned after 100 iterations

1	big drum ?
3	horse .
8	who is that ?
9	those are checkers .
3	two checkers # yes .
1	play checkers ?
1	big horn ?
2	get over # Mommy .
1	shadow ?
9	I like it .
1	why don't you read Shadow yourself ?
9	that's a terribly small horse for you to ride .
2	why don't you look at some of the toys in the basket .
1	want to ?
1	do you want to see what I have ?
8	what is that ?
2	not in your mouth .
2	let me put them together .
2	no # put floor .
3	no # that's his pencil .
3	that's not Daddy # that's Colin

Most probable words in each cluster

P(Z=4) = 0.4334		P(Z=9) = 0.3111		P(Z=7) = 0.2555		P(Z=3) = 5.003e	
X	P(X | Z)	X	P(X | Z)	X	P(X | Z)	X	P(X | Z
.	0.12526	?	0.19147	.	0.2258	quack	0.85
#	0.045402	you	0.062577	#	0.0695	.	0.15
you	0.040475	what	0.061256	that's	0.034538		
the	0.030259	that	0.022295	a	0.034066		
it	0.024154	the	0.022126	no	0.02649		
I	0.021848	#	0.021809	oh	0.023558		
to	0.018473	is	0.021683	yeah	0.020332		
don't	0.015473	do	0.016127	the	0.014907		
a	0.013662	it	0.015927	xxx	0.014288		
?	0.013459	a	0.015092	not	0.013864		
in	0.011708	to	0.013783	it's	0.013343		
on	0.011064	did	0.012631	?	0.013033		
your	0.010145	are	0.011427	yes	0.011795		
and	0.009578	what's	0.011195	right	0.0094166		
that	0.0093303	your	0.0098961	alright	0.0088953		
have	0.0088019	huh	0.0082591	is	0.0087975		
no	0.0082514	want	0.0076782	you're	0.0076571		
put	0.0067486	where	0.0072346	one	0.006647		
know	0.0064239	why	0.0070656	!	0.0057673		

Remarks on cluster results

- The samplers cluster words by clustering the documents they appear in, and cluster documents by clustering the words that appear in them
- Even though there were $\ell = 10$ clusters and $\boldsymbol{\alpha} = \mathbf{1}$, $\boldsymbol{\beta} = \mathbf{1}$, typically only 4 clusters were occupied after convergence
- Words x with high marginal probability $\mathrm{P}(X = x)$ are typically so frequent that they occur in all clusters

⇒ Listing the most probable words in each cluster may not be a good way of characterizing the clusters

- Instead, we can Bayes invert and find *the words that are most strongly associated with each class*

$$\mathrm{P}(Z = k \mid X = x) \quad = \quad \frac{N_{k,x}(\boldsymbol{Z}, \boldsymbol{X}) + \epsilon}{N_x(\boldsymbol{X}) + \epsilon\ell}$$

Purest words of each cluster

P(Z=4) = 0.4334		P(Z=9) = 0.3111		P(Z=7) = 0.2555		P(Z=3) = 5.0	
X	P(Z|X)	X	P(Z|X)	X	P(Z|X)	X	P(Z
I'll	0.97168	d(o)	0.97138	0	0.94715	quack	0.64
we'll	0.96486	what's	0.95242	mmhm	0.944	.	0.00
c(o)me	0.95319	what're	0.94348	www	0.90244		
you'll	0.95238	happened	0.93722	m:hm	0.83019		
may	0.94845	hmm	0.93343	uhhuh	0.81667		
let's	0.947	whose	0.92437	uh(uh)	0.78571		
thought	0.94382	what	0.9227	uhuh	0.77551		
won't	0.93645	where's	0.92241	that's	0.7755		
come	0.93588	doing	0.90196	yep	0.76531		
let	0.93255	where'd	0.9009	um	0.76282		
I	0.93192	don't]	0.89157	oh+boy	0.73529		
(h)ere	0.93082	whyn't	0.89157	d@l	0.72603		
stay	0.92073	who	0.88527	goodness	0.7234		
later	0.91964	how's	0.875	s@l	0.72		
thank	0.91667	who's	0.85068	sorry	0.70588		
them	0.9124	[:	0.85047	thank+you	0.6875		
can't	0.90762	?	0.84783	o:h	0.68		
never	0.9058	matter	0.82963	nope	0.67857		
em	0.89922	what'd	0.8125	hi	0.67213		

Summary

- Complex models often don't have analytic solutions
- Approximate inference can be used on many such models
- Monte Carlo Markov chain methods produce samples from (an approximation to) the posterior distribution
- Gibbs sampling is an MCMC procedure that resamples each variable conditioned on the values of the other variables
- If you can sample from the conditional distribution of each hidden variable in a Bayes net, you can use Gibbs sampling to sample from the joint posterior distribution
- We applied Gibbs sampling to Dirichlet-multinomial mixtures to cluster sentences

Outline

Introduction to Bayesian Inference

Mixture models

Sampling with Markov Chains

The Gibbs sampler

Gibbs sampling for Dirichlet-Multinomial mixtures

Topic modeling with Dirichlet multinomial mixtures

Chinese Restaurant Processes

Bayesian inference for Dirichlet-multinomials

- Probability of next event with *uniform Dirichlet prior* with mass α over m outcomes and observed data $\boldsymbol{Z}_{1:n} = (Z_1, \ldots, Z_n)$

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{Z}_{1:n}, \alpha) \quad \propto \quad n_k(\boldsymbol{Z}_{1:n}) + \alpha / m$$

  where $n_k(\boldsymbol{Z}_{1:n})$ is number of times k appears in $\boldsymbol{Z}_{1:n}$
- Example: Coin ($m = 2$), $\alpha = 1$, $\boldsymbol{Z}_{1:2} = (\text{heads}, \text{heads})$
  - $\mathrm{P}(Z_3 = \text{heads} \mid \boldsymbol{Z}_{1:2}, \alpha) \propto 2.5$
  - $\mathrm{P}(Z_3 = \text{tails} \mid \boldsymbol{Z}_{1:2}, \alpha) \propto 0.5$

Dirichlet-multinomials with many outcomes

- Predictive probability:

$$P(Z_{n+1} = k \mid \mathbf{Z}_{1:n}, \alpha) \;\propto\; n_k(\mathbf{Z}_{1:n}) + \alpha/m$$

- Suppose the number of outcomes $m \gg n$. Then:

$$P(Z_{n+1} = k \mid \mathbf{Z}_{1:n}, \alpha) \;\propto\; \begin{cases} n_k(\mathbf{Z}_{1:n}) & \text{if } n_k(\mathbf{Z}_{1:n}) > 0 \\ \alpha/m & \text{if } n_k(\mathbf{Z}_{1:n}) = 0 \end{cases}$$

- But *most outcomes will be unobserved*, so:

$$P(Z_{n+1} \notin \mathbf{Z}_{1:n} \mid \mathbf{Z}_{1:n}, \alpha) \;\propto\; \alpha$$

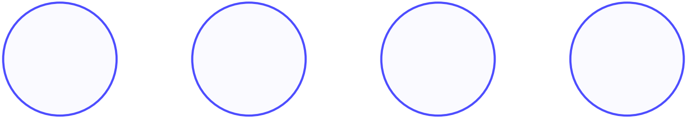
From Dirichlet-multinomials to Chinese Restaurant Processes

…

- Suppose *number of outcomes is unbounded* but *we* pick the event labels
- If we number event types in order of occurrence
  ⇒ *Chinese Restaurant Process*

$$
\begin{aligned}
Z_1 &= 1 \\
\mathrm{P}(Z_{n+1} = k \mid \mathbf{Z}_{1:n}, \alpha) &\propto \begin{cases} n_k(\mathbf{Z}_{1:n}) & \text{if } k \leq m = \max(\mathbf{Z}_{1:n}) \\ \alpha & \text{if } k = m + 1 \end{cases}
\end{aligned}
$$

Chinese Restaurant Process (0)

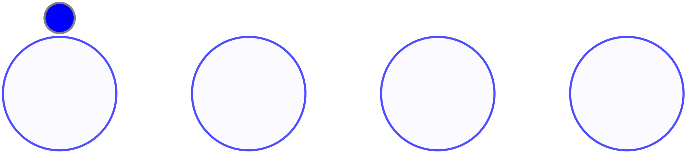


- Customer $\rightarrow$ table mapping $\boldsymbol{Z} =$
- $\mathrm{P}(\boldsymbol{z}) = 1$

- Next customer chooses a table according to:

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{Z}_{1:n}) \propto \begin{cases} n_k(\boldsymbol{Z}_{1:n}) & \text{if } k \leq m = \max(\boldsymbol{Z}_{1:n}) \\ \alpha & \text{if } k = m + 1 \end{cases}$$

Chinese Restaurant Process (1)

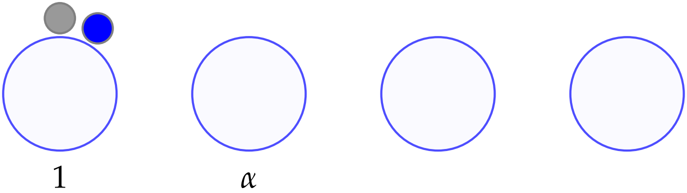


- Customer → table mapping $\boldsymbol{Z} = 1$
- $\mathrm{P}(\boldsymbol{z}) = \alpha / \alpha$

- Next customer chooses a table according to:

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{Z}_{1:n}) \propto \begin{cases} n_k(\boldsymbol{Z}_{1:n}) & \text{if } k \leq m = \max(\boldsymbol{Z}_{1:n}) \\ \alpha & \text{if } k = m + 1 \end{cases}$$

Chinese Restaurant Process (2)

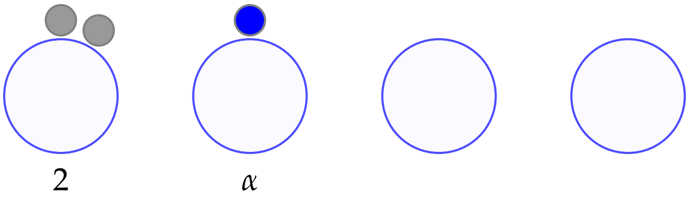


- Customer → table mapping $\boldsymbol{Z} = 1, 1$
- $\mathrm{P}(\boldsymbol{z}) = \alpha/\alpha \times 1/(1+\alpha)$

- Next customer chooses a table according to:

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{Z}_{1:n}) \propto \begin{cases} n_k(\boldsymbol{Z}_{1:n}) & \text{if } k \leq m = \max(\boldsymbol{Z}_{1:n}) \\ \alpha & \text{if } k = m+1 \end{cases}$$

Chinese Restaurant Process (3)

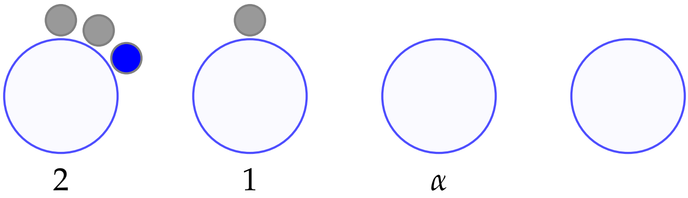


- Customer → table mapping $\mathbf{Z} = 1, 1, 2$
- $\mathrm{P}(\mathbf{z}) = \alpha/\alpha \times 1/(1+\alpha) \times \alpha/(2+\alpha)$

- Next customer chooses a table according to:

$$\mathrm{P}(Z_{n+1} = k \mid \mathbf{Z}_{1:n}) \;\propto\; \begin{cases} n_k(\mathbf{Z}_{1:n}) & \text{if } k \leq m = \max(\mathbf{Z}_{1:n}) \\ \alpha & \text{if } k = m+1 \end{cases}$$

Chinese Restaurant Process (4)

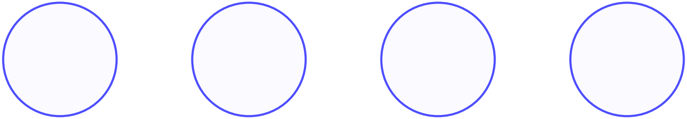


- Customer $\rightarrow$ table mapping $\boldsymbol{Z} = 1, 1, 2, 1$
- $\mathrm{P}(\boldsymbol{z}) = \alpha/\alpha \times 1/(1+\alpha) \times \alpha/(2+\alpha) \times 2/(3+\alpha)$

- Next customer chooses a table according to:

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{Z}_{1:n}) \propto \begin{cases} n_k(\boldsymbol{Z}_{1:n}) & \text{if } k \leq m = \max(\boldsymbol{Z}_{1:n}) \\ \alpha & \text{if } k = m+1 \end{cases}$$

Pitman-Yor Process (0)

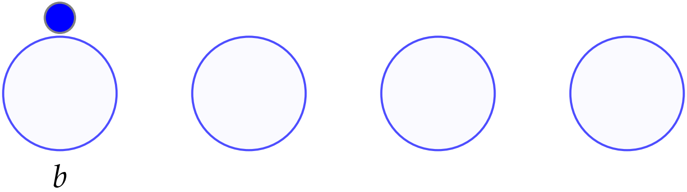


- Customer $\rightarrow$ table mapping $\boldsymbol{z} =$
- $\mathrm{P}(\boldsymbol{z}) = 1$

- In CRPs, probability of choosing a table $\propto$ number of customers $\Rightarrow$ strong *rich get richer* effect
- Pitman-Yor processes take mass a from each occupied table and give it to the new table

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{z}) \quad \propto \quad \begin{cases} n_k(\boldsymbol{z}) - a & \text{if } k \leq m = \max(\boldsymbol{z}) \\ ma + b & \text{if } k = m + 1 \end{cases}$$

Pitman-Yor Process (1)

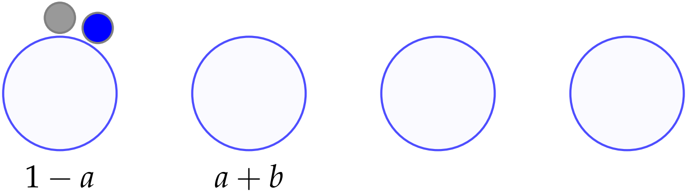


- Customer $\rightarrow$ table mapping $z = 1$
- $\mathrm{P}(z) = b/b$

- In CRPs, probability of choosing a table $\propto$ number of customers $\Rightarrow$ strong *rich get richer* effect
- Pitman-Yor processes take mass a from each occupied table and give it to the new table

$$\mathrm{P}(Z_{n+1} = k \mid z) \quad \propto \quad \begin{cases} n_k(z) - a & \text{if } k \le m = \max(z) \\ ma + b & \text{if } k = m + 1 \end{cases}$$

Pitman-Yor Process (2)

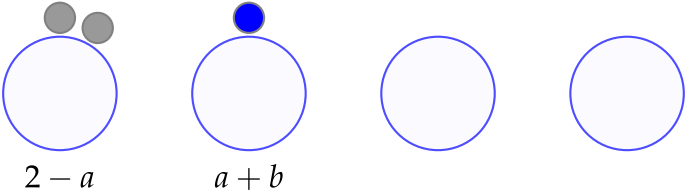


- Customer → table mapping $\boldsymbol{z} = 1, 1$
- $\mathrm{P}(\boldsymbol{z}) = b/b \times (1-a)/(1+b)$

- In CRPs, probability of choosing a table ∝ number of customers ⇒ strong *rich get richer* effect
- Pitman-Yor processes take mass a from each occupied table and give it to the new table

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{z}) \;\propto\; \begin{cases} n_k(\boldsymbol{z}) - a & \text{if } k \leq m = \max(\boldsymbol{z}) \\ ma + b & \text{if } k = m + 1 \end{cases}$$

Pitman-Yor Process (3)

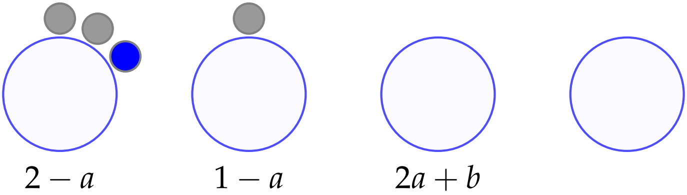


- Customer → table mapping $\boldsymbol{z} = 1, 1, 2$
- $\mathrm{P}(\boldsymbol{z}) = b/b \times (1 - a)/(1 + b) \times (a + b)/(2 + b)$

- In CRPs, probability of choosing a table ∝ number of customers ⇒ strong *rich get richer* effect
- Pitman-Yor processes take mass a from each occupied table and give it to the new table

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{z}) \quad \propto \quad \begin{cases} n_k(\boldsymbol{z}) - a & \text{if } k \leq m = \max(\boldsymbol{z}) \\ ma + b & \text{if } k = m + 1 \end{cases}$$

Pitman-Yor Process (4)

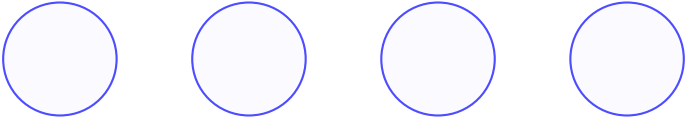


- Customer $\rightarrow$ table mapping $\boldsymbol{z} = 1, 1, 2, 1$
- $\mathrm{P}(\boldsymbol{z}) =$
  $b/b \times (1 - a)/(1 + b) \times (a + b)/(2 + b) \times (2 - a)/(3 + b)$

- In CRPs, probability of choosing a table $\propto$ number of customers $\Rightarrow$ strong *rich get richer* effect
- Pitman-Yor processes take mass a from each occupied table and give it to the new table

$$\mathrm{P}(Z_{n+1} = k \mid \boldsymbol{z}) \propto \begin{cases} n_k(\boldsymbol{z}) - a & \text{if } k \leq m = \max(\boldsymbol{z}) \\ ma + b & \text{if } k = m + 1 \end{cases}$$

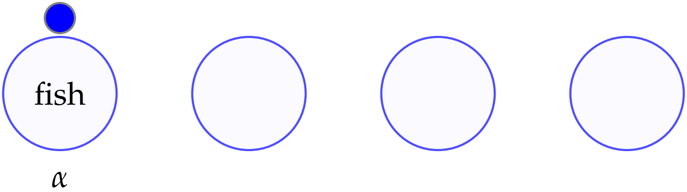
Labeled Chinese Restaurant Process (0)



- Table $\rightarrow$ label mapping $\boldsymbol{Y} =$
- Customer $\rightarrow$ table mapping $\boldsymbol{Z} =$
- Output sequence $\boldsymbol{X} =$
- $\mathrm{P}(\boldsymbol{X}) = 1$

- *Base distribution* $\mathrm{P}_0(Y)$ generates a *label* Y_k for each table k
- All customers sitting at table k (i.e., $Z_i = k$) share label Y_k
- Customer i sitting at table Z_i has label $X_i = Y_{Z_i}$

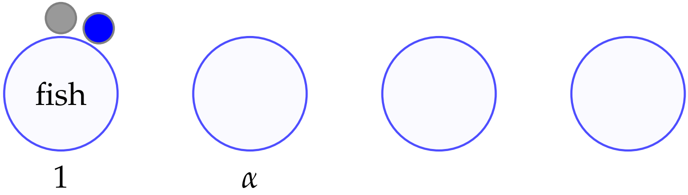
Labeled Chinese Restaurant Process (1)



- Table → label mapping $\boldsymbol{Y} = \text{fish}$
- Customer → table mapping $\boldsymbol{Z} = 1$
- Output sequence $\boldsymbol{X} = \text{fish}$
- $\mathrm{P}(\boldsymbol{X}) = \alpha/\alpha \times \mathrm{P}_0(\text{fish})$

- *Base distribution* $\mathrm{P}_0(Y)$ generates a *label* Y_k for each table k
- All customers sitting at table k (i.e., $Z_i = k$) share label Y_k
- Customer i sitting at table Z_i has label $X_i = Y_{Z_i}$

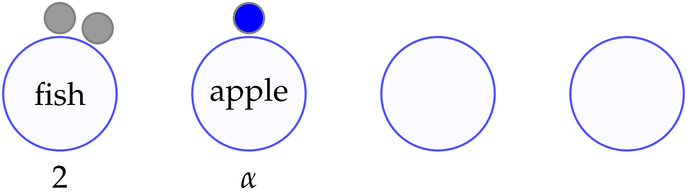
Labeled Chinese Restaurant Process (2)



- Table → label mapping $\boldsymbol{Y} = \text{fish}$
- Customer → table mapping $\boldsymbol{Z} = 1, 1$
- Output sequence $\boldsymbol{X} = \text{fish,fish}$
- $\mathrm{P}(\boldsymbol{X}) = \mathrm{P}_0(\text{fish}) \times 1/(1+\alpha)$

- *Base distribution* $\mathrm{P}_0(Y)$ generates a *label* Y_k for each table k
- All customers sitting at table k (i.e., $Z_i = k$) share label Y_k
- Customer i sitting at table Z_i has label $X_i = Y_{Z_i}$

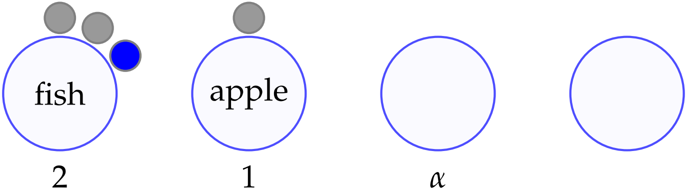
Labeled Chinese Restaurant Process (3)



- Table → label mapping $\boldsymbol{Y}$ = fish,apple
- Customer → table mapping $\boldsymbol{Z} = 1, 1, 2$
- Output sequence $\boldsymbol{X}$ = fish,fish,apple
- $\mathrm{P}(\boldsymbol{X}) = \mathrm{P}_0(\text{fish}) \times 1/(1+\alpha) \times \alpha/(2+\alpha)\mathrm{P}_0(\text{apple})$

- *Base distribution* $\mathrm{P}_0(Y)$ generates a *label* Y_k for each table k
- All customers sitting at table k (i.e., $Z_i = k$) share label Y_k
- Customer i sitting at table Z_i has label $X_i = Y_{Z_i}$

Labeled Chinese Restaurant Process (4)



- Table → label mapping $\boldsymbol{Y}$ = fish,apple
- Customer → table mapping $\boldsymbol{Z} = 1, 1, 2$
- Output sequence $\boldsymbol{X}$ = fish,fish,apple,fish
- $\mathrm{P}(\boldsymbol{X}) =$
  $\mathrm{P}_0(\text{fish}) \times 1/(1+\alpha) \times \alpha/(2+\alpha)\mathrm{P}_0(\text{apple}) \times 2/(3+\alpha)$

- *Base distribution* $\mathrm{P}_0(Y)$ generates a *label* Y_k for each table k
- All customers sitting at table k (i.e., $Z_i = k$) share label Y_k
- Customer i sitting at table Z_i has label $X_i = Y_{Z_i}$

From Chinese restaurants to Dirichlet processes

- Labeled Chinese restaurant processes take a distribution P_0 and return a stream of samples from a different distribution with the same support
- The Chinese restaurant process is a sequential process, generating the next item conditioned on the previous ones
- We can get a different distribution each time we run a CRP (allocation of customers to tables and labeling of tables are randomized)
- Abstracting away from the sequential generation of the CRP, we can view it as a mapping from a base distribution P_0 to a *distribution over distributions* $DP(\alpha, P_0)$
- $DP(\alpha, P_0)$ is called a *Dirichlet process* with *concentration parameter* α and *base distribution* P_0
- Distributions in $DP(\alpha, P_0)$ are *discrete* (w.p. 1) even if the base distribution P_0 is continuous

Gibbs sampling with Chinese restaurants

- Idea: resample z_i as if $\boldsymbol{z}_{-i}$ were "real" data
- The CRP is *exchangable:* all ways of generating an assignment of customers to labeled tables have the same probability
- This means $\mathrm{P}(z_i|\boldsymbol{z}_{-i})$ is the same as if z_i were generated *after* $\boldsymbol{s}_{-i}$
  - Exchangability means *"treat every customer as if they were your last"*
- Tables are generated and garbage-collected during sampling
- The probability of generating a new table includes the probability of generating its label
- When retracting z_i reduces the number of customers at a table to 0, garbage-collect the table
- CRPs not only estimate model parameters, they also *estimate the number of components (tables)*

A DP clustering model

- Idea: replace multinomials with Chinese restaurants
- $\mathrm{P}(z)$ is a distribution over integers (clusters), generated by a CRP
- For each cluster z, run separate Chinese restaurants for $\mathrm{P}(x|c)$
- $\mathrm{P}(x|c)$ are distributions over words, so they need generator distributions
  - generators could be uniform over the named entities/contexts in training data, or
  - (n-gram) language models generating possible named entities/contexts (unbounded vocabulary)
- In a *hierarchical Dirichlet process*, these generators could themselves be Dirichlet processes that possibly share a common vocabulary

Summary: Chinese Restaurant Processes

- *Chinese Restaurant Processes* (CRPs) generalize Dirichlet-Multinomials to an *unbounded number of outcomes*
  - *concentration parameter* α controls how likely a new outcome is
  - CRPs exhibit a *rich get richer* power-law behaviour
- *Labeled CRPs* use a *base distribution* to label each table
  - base distribution can have *infinite support*
  - concentrates mass on a countable subset
  - power-law behaviour $\Rightarrow$ Zipfian distributions