Markov Chain Sampling Methods for Dirichlet Process Mixture Models

K component mixture model

$$\pi \sim \text{Dirichlet}(\alpha/K, ..., \alpha/K) \tag{1}$$
$$z_i|\pi \sim \text{Categorical}(\pi) \tag{2}$$
$$\theta_k|H \sim H \tag{3}$$
$$y_i|z_i, \theta \sim F(\theta_{z_i}), \tag{4}$$

$\alpha > 0$ and H is the prior over the parameters $\theta_k \in \Theta$.

K component mixture model

When K is known, we have seen that EM-algorithm can be applied to estimate the parameters.

But in many settings, K is unknown and we need to experiment with different values of K.

Applications

- Topic modeling: organize/label a corpus of documents into K topics. How do you choose K?

Applications

- Topic modeling: organize/label a corpus of documents into K topics. How do you choose K?
- Cancer clonal detection: given an admixture of cells, detect the number of cancer subpopulations.

Applications

- Topic modeling: organize/label a corpus of documents into K topics. How do you choose K?
- Cancer clonal detection: given an admixture of cells, detect the number of cancer subpopulations.
- Density estimation: modeling multi-modal distribution with unknown components.

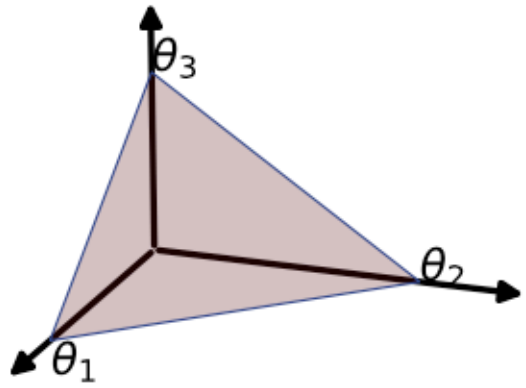
Dirichlet distribution

- "Distribution" of (discrete) distributions over K categories.

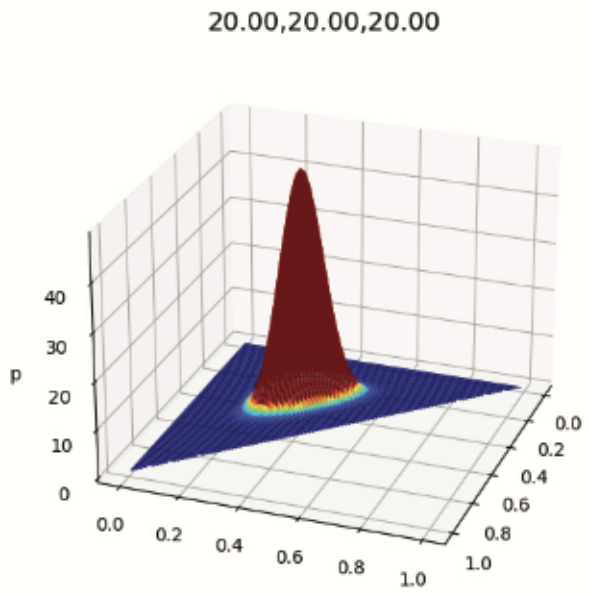
$$\pi \sim \text{Dirichlet}(\alpha_1, ..., \alpha_K), \quad \alpha_k > 0$$

- $\pi_k \in [0, 1]$.
- $\sum_{k=1}^{K} \pi_k = 1$.

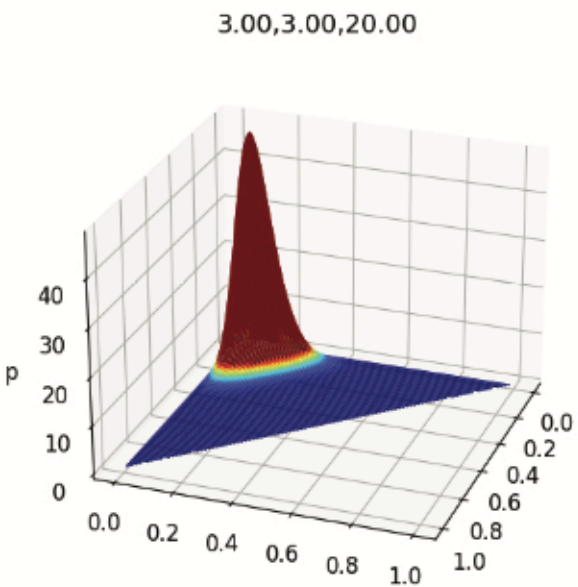
Dirichlet distribution

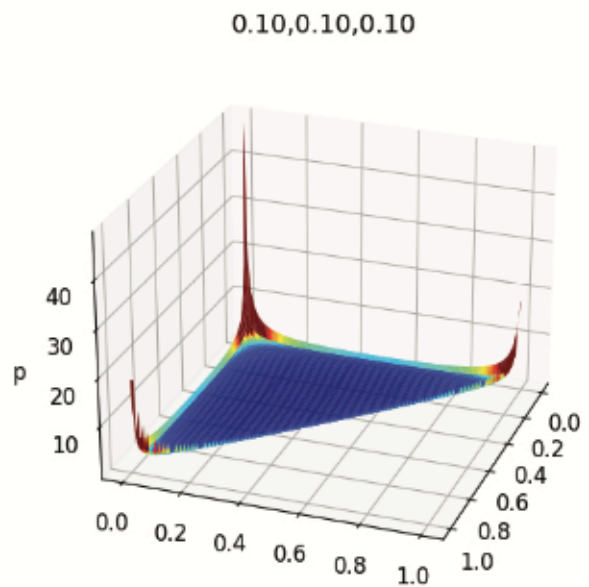


(a)



(b)





Dirichlet Process

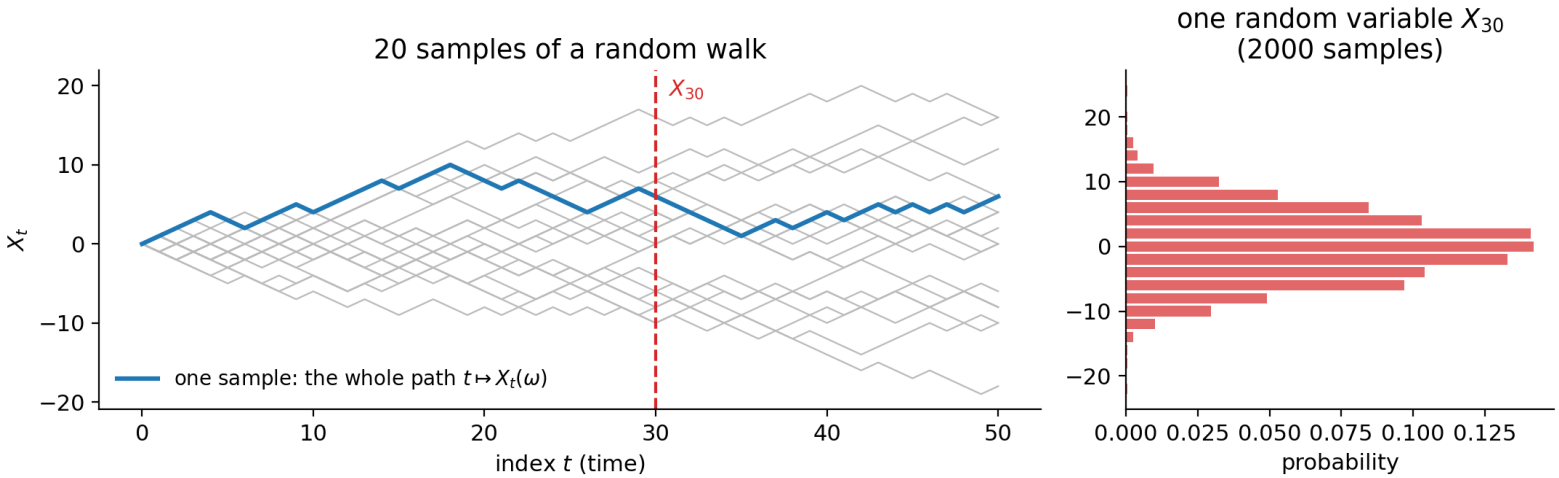
When K is known, we use Dirichlet distribution.

When K is not known, we use a Dirichlet Process to place a prior over distributions (or, an unbounded mixture). Let's see how that works and what is a distribution over distributions?

What is a stochastic process?

A stochastic process is a collection of random variables $\{X_t : t \in T\}$, indexed by a set T.

You have seen one: a Markov chain $z_1, z_2, \ldots$ (HMM, MCMC), with $T = \{1, 2, \ldots\}$.



- Fix t: X_t is an ordinary random variable (right).
- **One sample of the process is the whole path** $t \mapsto X_t$, drawn all at once (blue).

Same idea, bigger index set

Process	Index t	Value X_t	One sample is a...
Random walk	time $t = 1, 2, ...$	position X_t	path
Dirichlet process	measurable set $A \subseteq \Theta$	mass $G(A) \in [0, 1]$	probability measure

Same idea, bigger index set

Process	Index t	Value X_t	One sample is a...
Random walk	time $t = 1, 2, ...$	position X_t	path
Dirichlet process	measurable set $A \subseteq \Theta$	mass $G(A) \in [0, 1]$	probability measure

The DP is a stochastic process too, but indexed by (measurable) sets $A \subseteq \Theta$ rather than by time as we are used to: the value at A is how much probability G puts on that region.

Each sample is a distribution

Let $\mathcal{B}$ be the Borel σ-algebra of Θ (the measurable sets). A probability distribution on Θ *is* a function $P : \mathcal{B} \to [0, 1]$ with

- $P(\Theta) = 1$,
- $P\left(\bigcup_{k=1}^{\infty} A_k\right) = \sum_{k=1}^{\infty} P(A_k)$ for pairwise disjoint $A_1, A_2, ... \in \mathcal{B}$.

Each sample is a distribution

Let $\mathcal{B}$ be the Borel σ-algebra of Θ (the measurable sets). A probability distribution on Θ *is* a function $P : \mathcal{B} \to [0, 1]$ with

- $P(\Theta) = 1$,
- $P\left(\bigcup_{k=1}^{\infty} A_k\right) = \sum_{k=1}^{\infty} P(A_k)$ for pairwise disjoint $A_1, A_2, ... \in \mathcal{B}$.

The DP is the process $\{G(A) : A \in \mathcal{B}\}$. With probability 1, its path $A \mapsto G(A)$ satisfies both rules, i.e., it is a probability measure on $(\Theta, \mathcal{B})$. So "sample a distribution" = "sample a path of $\{G(A)\}_{A \in \mathcal{B}}$".

(Precisely: there is a version of the process with this property (Ferguson 1973); stick breaking constructs one directly.)

Dirichlet Process

Distribution over infinite-dimensional discrete probability measures:

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

where H is the base measure defined on Θ and $\alpha > 0$ is the concentration parameter.

- G is a random probability measure defined on Θ.

Intuition: what is a "random distribution"?

Think of a factory that manufactures loaded dice.

Intuition: what is a "random distribution"?

Think of a factory that manufactures loaded dice.

- **Dirichlet**: each draw $\pi \sim \text{Dirichlet}(\alpha_1, ..., \alpha_K)$ is one K-sided die, i.e., a distribution over K fixed categories.

Intuition: what is a "random distribution"?

Think of a factory that manufactures loaded dice.

- **Dirichlet**: each draw $\pi \sim \text{Dirichlet}(\alpha_1, ..., \alpha_K)$ is one K-sided die, i.e., a distribution over K fixed categories.
- **DP**: each draw $G \sim \text{DP}(\alpha, H)$ is one die with countably many faces. Each face is labelled by a location $\theta_k \sim H$ and is rolled with probability π_k:

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

Intuition: what is a "random distribution"?

Think of a factory that manufactures loaded dice.

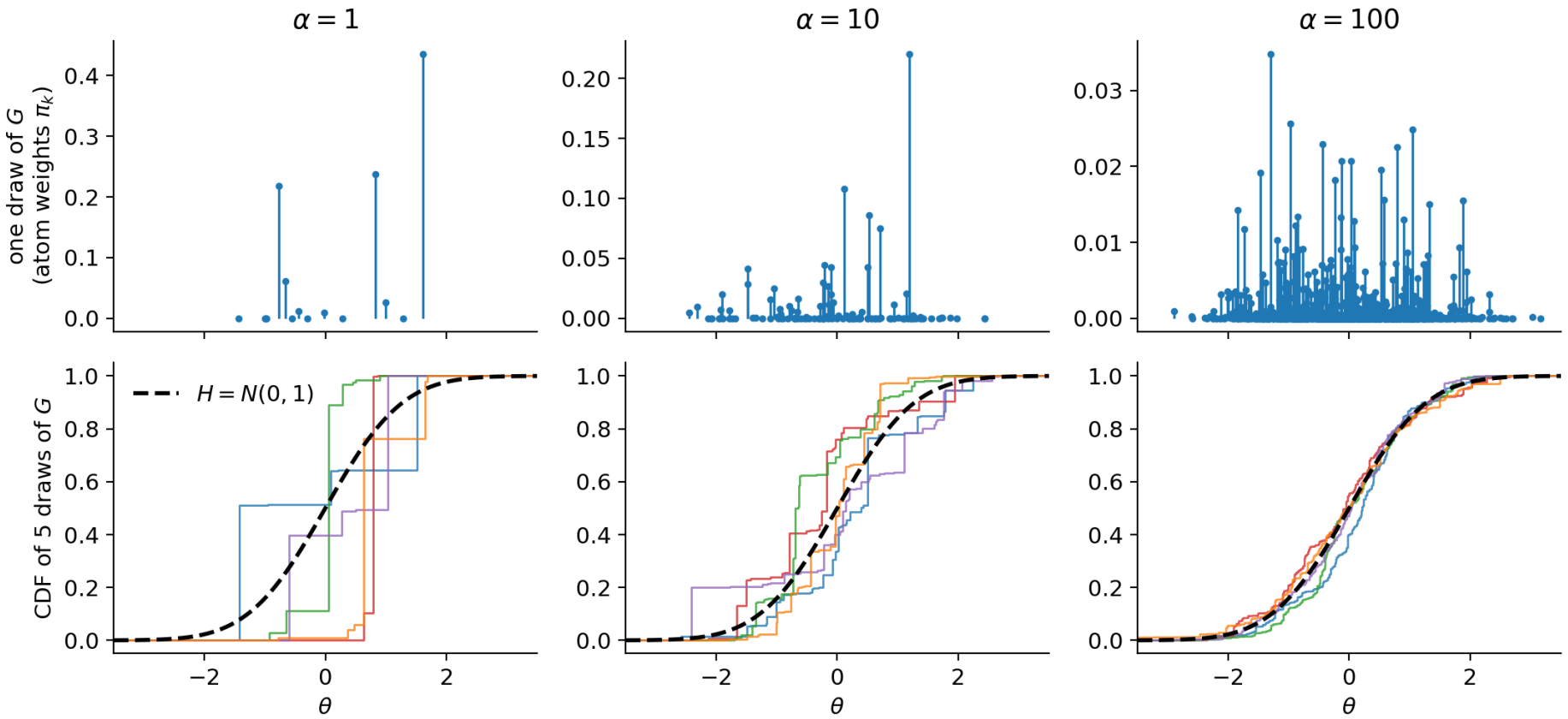
- **Dirichlet**: each draw $\pi \sim \text{Dirichlet}(\alpha_1, ..., \alpha_K)$ is one K-sided die, i.e., a distribution over K fixed categories.
- **DP**: each draw $G \sim \text{DP}(\alpha, H)$ is one die with countably many faces. Each face is labelled by a location $\theta_k \sim H$ and is rolled with probability π_k:

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

Two levels of randomness:

1. Draw the die G (once).
2. Roll the die: $\theta_1, \theta_2, ... \overset{\text{iid}}{\sim} G$. Repeated faces $\Rightarrow$ repeated θ's $\Rightarrow$ clusters.

Intuition: draws of G with $H = N(0, 1)$



- H says *where* the atoms go; α says *how spread out* the mass is.
- Small α: a few big atoms. Large α: many small atoms, $G \approx H$.
- Every draw is discrete, even though H is continuous.

Dirichlet Process

G is Dirichlet process distributed with base distribution H and concentration parameter α, if and only if

$$(G(A_1), ..., G(A_K)) \sim \text{Dirichlet}(\alpha H(A_1), ..., \alpha H(A_K))$$

for every finite measurable partition $A_1, ..., A_K$ of Θ.

Note: $G(A_k)$ is a random variable because G is a random measure.

Why define it through finite partitions?

A path is an infinite object: we cannot write a density over all paths. Instead we specify every **finite-dimensional marginal**.

- Random walk: the joint law of $(X_{t_1}, ..., X_{t_n})$ for any finite set of times.
- DP: the joint law of $(G(A_1), ..., G(A_K))$ for any finite partition $\Rightarrow$ Dirichlet.

Consistency of the finite-dimensional marginals

The marginals must agree with each other. Take the partition $\{A_1, A_2, A_3\}$ and the coarser one $\{A_1 \cup A_2, A_3\}$. The definition gives a Dirichlet law for each, and summing the first two components of the first must give the second. It does, because Dirichlet components aggregate:

$$(G(A_1) + G(A_2),\ G(A_3)) \sim \text{Dirichlet}(\alpha H(A_1) + \alpha H(A_2),\ \alpha H(A_3)),$$

and $\alpha H(A_1) + \alpha H(A_2) = \alpha H(A_1 \cup A_2)$.

Consistency of the finite-dimensional marginals

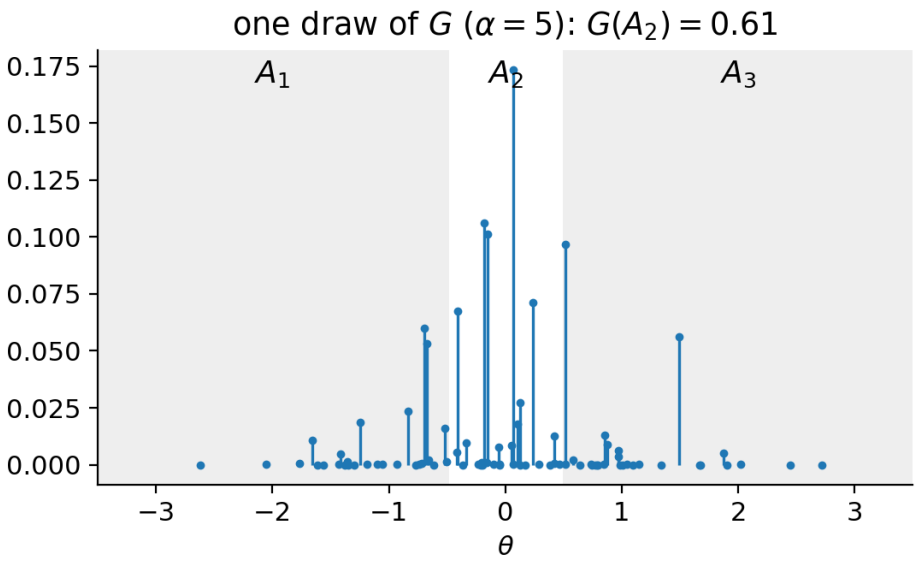
The marginals must agree with each other. Take the partition $\{A_1, A_2, A_3\}$ and the coarser one $\{A_1 \cup A_2, A_3\}$. The definition gives a Dirichlet law for each, and summing the first two components of the first must give the second. It does, because Dirichlet components aggregate:

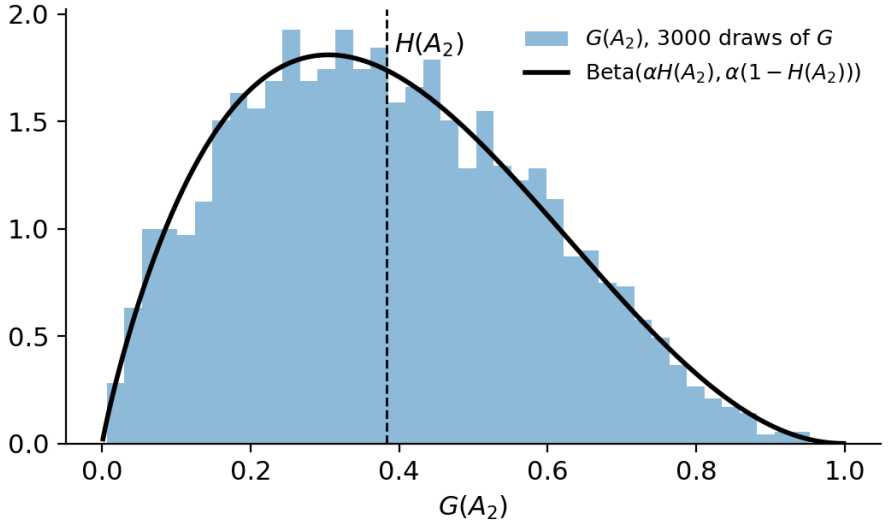
$$(G(A_1) + G(A_2),\ G(A_3)) \sim \text{Dirichlet}(\alpha H(A_1) + \alpha H(A_2),\ \alpha H(A_3)),$$

and $\alpha H(A_1) + \alpha H(A_2) = \alpha H(A_1 \cup A_2)$.

Kolmogorov's extension theorem: consistent finite-dimensional marginals determine a unique stochastic process (Ferguson 1973 used this, with some technical care, to construct the DP). That is why the DP definition looks the way it does.

Intuition: the partition view





Fix a partition A_1, A_2, A_3 of Θ. Each draw of G puts some mass on each set. Over many draws, the masses $(G(A_1), G(A_2), G(A_3))$ are Dirichlet; here $G(A_2) \sim \text{Beta}(\alpha H(A_2), \alpha(1 - H(A_2)))$.

The DP is the object that is consistent with this for *every* finite measurable partition.

What does "draw G" mean in practice?

As with a random walk, we never hold the whole sample, only finite pieces of it:

What does "draw G" mean in practice?

As with a random walk, we never hold the whole sample, only finite pieces of it:

1. **Look at finitely many regions**: $(G(A_1), ..., G(A_K))$ is a Dirichlet draw (partition view, above).

What does "draw G" mean in practice?

As with a random walk, we never hold the whole sample, only finite pieces of it:

1. **Look at finitely many regions**: $(G(A_1), ..., G(A_K))$ is a Dirichlet draw (partition view, above).
2. **Build G atom by atom**: a constructive recipe that produces (π_k, θ_k) for $k = 1, 2, ...$, stopping once the leftover mass is negligible (stick breaking, coming up).

What does "draw G" mean in practice?

As with a random walk, we never hold the whole sample, only finite pieces of it:

1. **Look at finitely many regions**: $(G(A_1), ..., G(A_K))$ is a Dirichlet draw (partition view, above).
2. **Build G atom by atom**: a constructive recipe that produces (π_k, θ_k) for $k = 1, 2, ...$, stopping once the leftover mass is negligible (stick breaking, coming up).
3. **Never draw G at all**: integrate it out and sample only $\theta_1, ..., \theta_N$ (Chinese restaurant process, later). This is what the MCMC algorithms do.

Dirichlet Process

- $\mathbb{E}[G(A)] = H(A)$.
- $\text{var}(G(A)) = H(A)(1 - H(A))/(\alpha + 1)$.

Larger the value of α, the smaller the variance (concentrated around the mean $H(A)$).

A measure G sampled from DP is discrete with probability 1.

Posterior distribution of G

Since G is a distribution, we can draw samples from G.

Let $\theta_i \sim G$.

Given $\theta_1, ..., \theta_N$, what is the posterior distribution $p(G|\theta_{1:N})$?

Posterior distribution of G

Fix a finite measurable partition $A_1, ..., A_K$ of Θ, and let $n_k = |\{i : \theta_i \in A_k\}|$, the number of points that fall in A_k (so $\sum_k n_k = N$).

- $(G(A_1), ..., G(A_K)) \sim \text{Dirichlet}(\alpha H(A_1), ..., \alpha H(A_K))$
- $(n_1, ..., n_K) \sim \text{Multinomial}(N; G(A_1), ..., G(A_K))$
- Dirichlet and Multinomial are conjugate distributions:

$$(G(A_1), ..., G(A_K))|\theta_1, ..., \theta_N \sim \text{Dirichlet}(\alpha'_k),$$

where

$$\alpha'_k = \alpha H(A_k) + n_k.$$

Posterior over G

$G \sim DP(\alpha, H)$ if and only if for every finite measurable partition $A_1, ..., A_K$ of Θ,

$$(G(A_1), ..., G(A_K)) \sim \text{Dirichlet}(\alpha H(A_1), ..., \alpha H(A_K)).$$

Since the result from previous slide holds for arbitrary partition $(A_1, ..., A_K)$, the posterior distribution $G|\theta_{1:N}$ is also a Dirichlet Process.

Therefore, DP provides a conjugate family of priors over (discrete) probability distributions.

Posterior over G

How do we update the hyperparameters?

$$G|\theta_1, ..., \theta_N \sim DP(\alpha', H')$$

$\alpha' = \alpha + N$ and

$$H' = \frac{\alpha H + \sum_{i=1}^{N} \delta_{\theta_i}}{\alpha + N},$$

a convex combination of the base measure H and the empirical measure $\hat{G}_N = \frac{1}{N} \sum_{i=1}^{N} \delta_{\theta_i}$:

$$H' = \frac{\alpha}{\alpha + N} H + \frac{N}{\alpha + N} \hat{G}_N.$$

Why?

Posterior over G

We know that,

$$(G(A_1), ..., G(A_K))|\theta_{1:N} \sim \text{Dirichlet}(\alpha H(A_k) + n_k),$$

which has density

$$\propto \prod_{k=1}^{K} G(A_k)^{\alpha H(A_k)+n_k-1}.$$

This implies that $\alpha' = \sum_{k=1}^{K}(\alpha H(A_k) + n_k) = \alpha \cdot 1 + N$.

$$\alpha' H'(A_k) = \alpha H(A_k) + n_k \Rightarrow H'(A_k) = \frac{\alpha H(A_k) + n_k}{\alpha + N}.$$

Note: $n_k = \sum_{i=1}^{N} \delta_{\theta_i}(A_k)$.

Stick breaking process

How do we actually build a sample of G? Sethuraman's stick breaking construction.

Let u be a unit stick (length 1). We will break this stick infinite number of times.

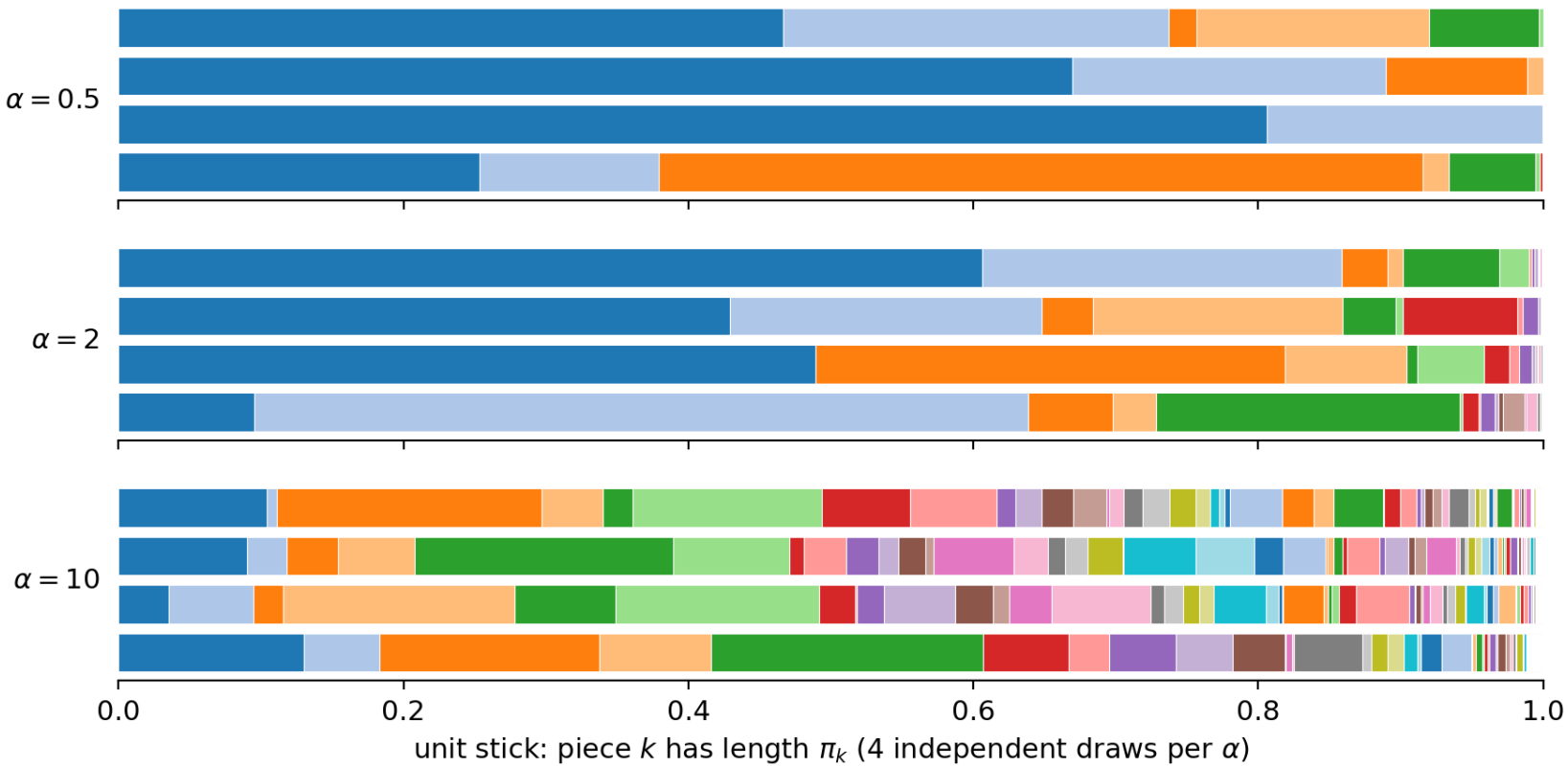
For $i = 1, ..., \infty$,

- Sample $\beta_i \sim \text{Beta}(1, \alpha)$
- Set $\pi_i = \beta_i \prod_{n=1}^{i-1}(1 - \beta_n)$; $\pi_1 = \beta_1$.
- Sample $\theta_i \sim H$.

$$G = \sum_i \pi_i \delta_{\theta_i}$$

is a realization from $DP(\alpha, H)$.

Stick breaking process



$\mathbb{E}[\beta_i] = 1/(1+\alpha)$: small α breaks off big pieces early (few dominant atoms); large α breaks off small pieces (many atoms of similar size).

Dirichlet process mixture model

Generative model.

Dirichlet process mixture model

Generative model.

We first sample a random measure: $G \sim \mathrm{DP}(\alpha, H)$.

Dirichlet process mixture model

Generative model.

We first sample a random measure: $G \sim \mathrm{DP}(\alpha, H)$.

We then sample parameters for each datum i: $\theta_i | G \sim G$.

Dirichlet process mixture model

Generative model.

We first sample a random measure: $G \sim \mathrm{DP}(\alpha, H)$.

We then sample parameters for each datum i: $\theta_i | G \sim G$.

Finally, sample the datum: $y_i | \theta_i \sim F(\theta_i)$.

Dirichlet process mixture model

Generative model.

We first sample a random measure: $G \sim \mathrm{DP}(\alpha, H)$.

We then sample parameters for each datum i: $\theta_i | G \sim G$.

Finally, sample the datum: $y_i | \theta_i \sim F(\theta_i)$.

In our usual notation: $x_O = y_{1:N}$ and $x_H = (z_{1:N}, \theta)$, where z_i indexes which atom of G generated θ_i.

Dirichlet process mixture model

Generative model.

We first sample a random measure: $G \sim \text{DP}(\alpha, H)$.

We then sample parameters for each datum i: $\theta_i | G \sim G$.

Finally, sample the datum: $y_i | \theta_i \sim F(\theta_i)$.

In our usual notation: $x_O = y_{1:N}$ and $x_H = (z_{1:N}, \theta)$, where z_i indexes which atom of G generated θ_i.

How does this model solve the clustering problem with unknown K?

Dirichlet process mixture model

To simplify discussion,

- Let H be a Normal-Inverse-Gamma distribution defined on $\Theta = \mathbb{R} \times \mathbb{R}^+$ (parameter space of location and scale): $\sigma^2 \sim \text{Inv-Gamma}(a, b)$, $\mu|\sigma^2 \sim N(m, \sigma^2/\kappa)$.

Dirichlet process mixture model

To simplify discussion,

- Let H be a Normal-Inverse-Gamma distribution defined on $\Theta = \mathbb{R} \times \mathbb{R}^+$ (parameter space of location and scale): $\sigma^2 \sim \text{Inv-Gamma}(a, b)$, $\mu|\sigma^2 \sim N(m, \sigma^2/\kappa)$.
- Let F also be Normal distribution with $\theta_i = (\mu_i, \sigma_i^2) \in \Theta$.

Dirichlet process mixture model

To simplify discussion,

- Let H be a Normal-Inverse-Gamma distribution defined on $\Theta = \mathbb{R} \times \mathbb{R}^+$ (parameter space of location and scale): $\sigma^2 \sim$ Inv-Gamma(a, b), $\mu|\sigma^2 \sim N(m, \sigma^2/\kappa)$.
- Let F also be Normal distribution with $\theta_i = (\mu_i, \sigma_i^2) \in \Theta$.

G is a random distribution of infinite dimension ($K = 1, 2, 3, ...$).

Dirichlet process mixture model

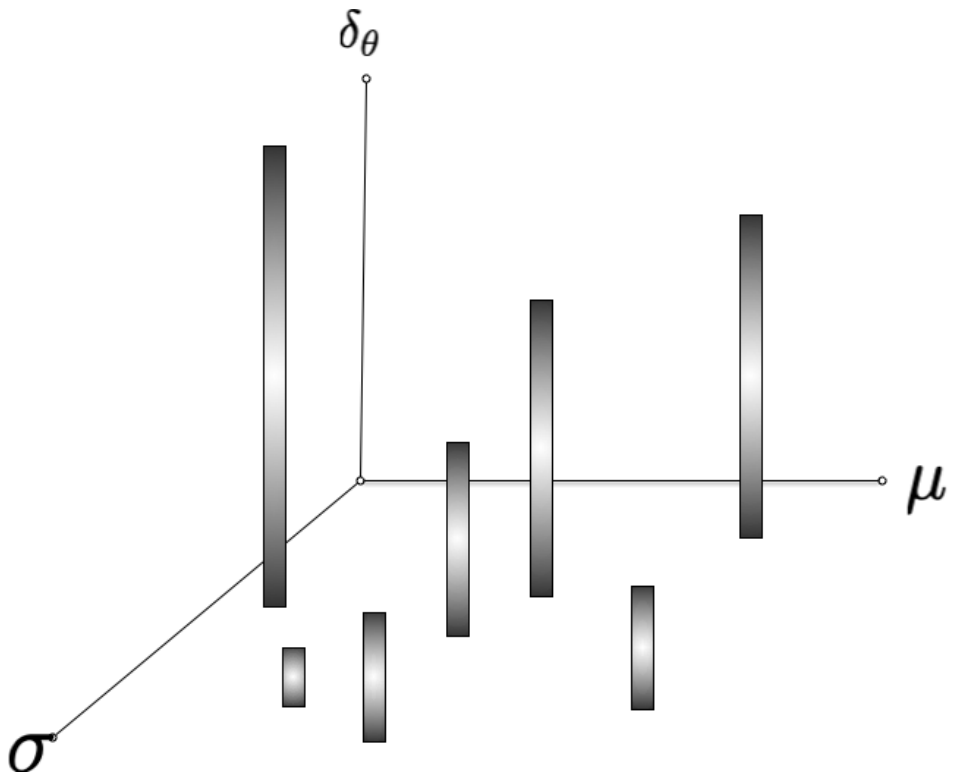
To simplify discussion,

- Let H be a Normal-Inverse-Gamma distribution defined on $\Theta = \mathbb{R} \times \mathbb{R}^+$ (parameter space of location and scale): $\sigma^2 \sim$ Inv-Gamma(a, b), $\mu|\sigma^2 \sim N(m, \sigma^2/\kappa)$.
- Let F also be Normal distribution with $\theta_i = (\mu_i, \sigma_i^2) \in \Theta$.

G is a random distribution of infinite dimension ($K = 1, 2, 3, ...$).

G is discrete with probability $1 \Rightarrow$ some θ_i will be repeated $\Rightarrow$ clustering but with undetermined dimension K.

Dirichlet process mixture model



A realization of $G = \sum_k \pi_k \delta_{\theta_k}$: each bar is an atom at a unique $\theta_k = (\mu_k, \sigma_k)$ and its height is the weight π_k.

Chinese Restaurant Process

How do we do posterior inference? We need a prior over partitions.

- Consider a Chinese restaurant with infinitely many tables

Chinese Restaurant Process

How do we do posterior inference? We need a prior over partitions.

- Consider a Chinese restaurant with infinitely many tables
- First customer enters the restaurant and chooses a table to seat and selects a dish $\theta \sim H$.

Chinese Restaurant Process

How do we do posterior inference? We need a prior over partitions.

- Consider a Chinese restaurant with infinitely many tables
- First customer enters the restaurant and chooses a table to seat and selects a dish $\theta \sim H$.
- Each subsequent customer enters the restaurant, selects one of K tables based on probability:

Chinese Restaurant Process

How do we do posterior inference? We need a prior over partitions.

- Consider a Chinese restaurant with infinitely many tables
- First customer enters the restaurant and chooses a table to seat and selects a dish $\theta \sim H$.
- Each subsequent customer enters the restaurant, selects one of K tables based on probability:

$$p(z_i = k | z_{1:i-1}) = \frac{n_k}{i - 1 + \alpha}$$

and shares the dish θ_k or seat on a new table

$$p(z_i = k' | z_{1:i-1}) = \frac{\alpha}{i - 1 + \alpha}$$

and sample a new dish $\theta \sim H$.

Chinese Restaurant Process

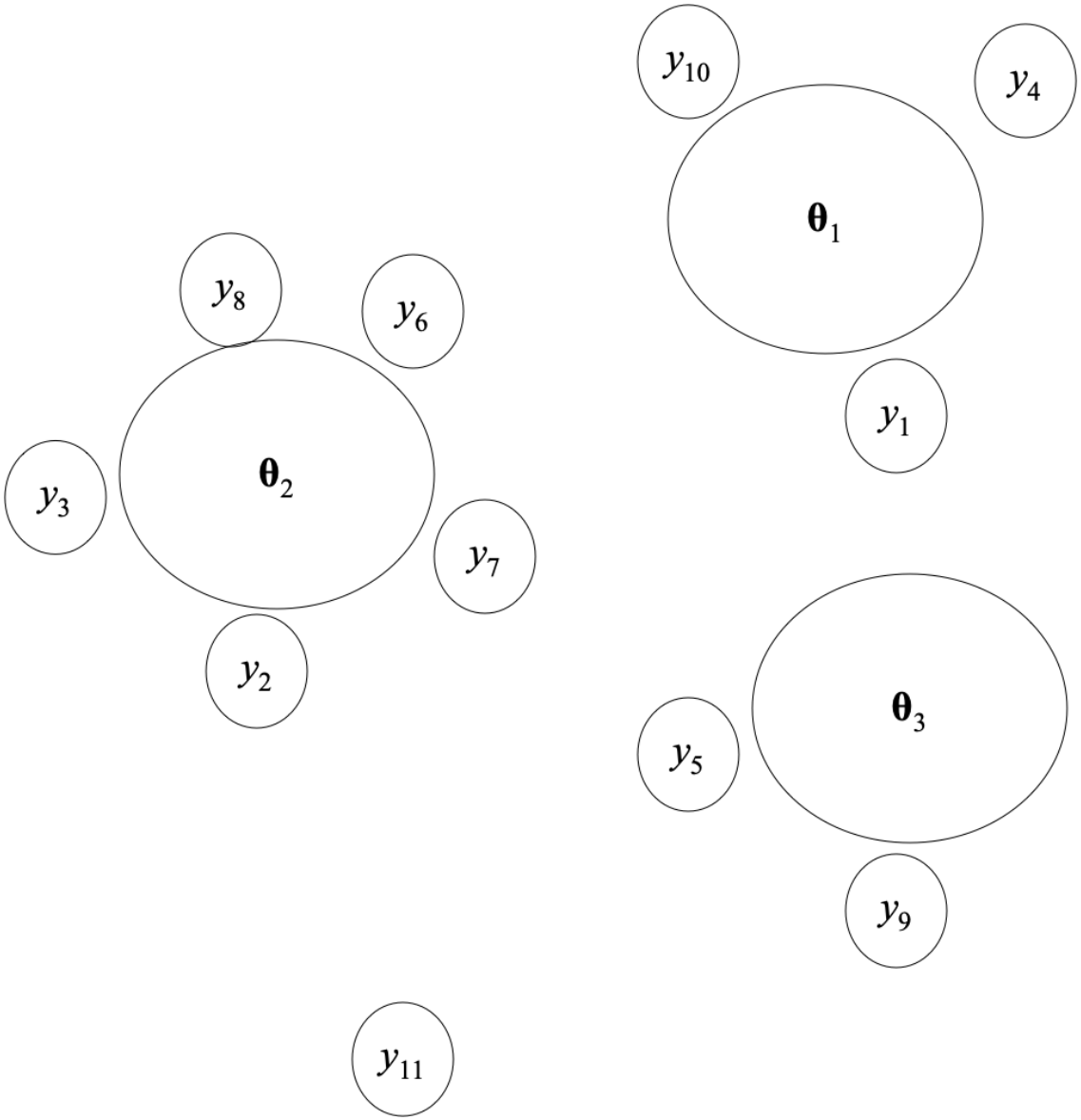
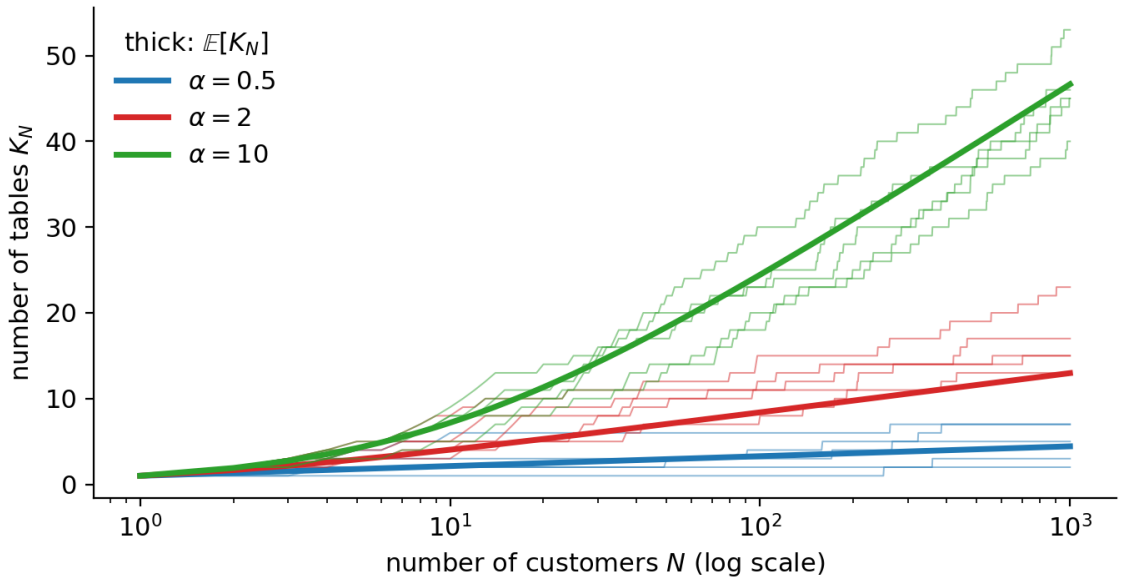


Figure 2: Compute $P(z_{11} = k|z_{1:10})$

How many tables?

Customer i opens a new table with probability $\alpha/(\alpha + i - 1)$, so

$$\mathbb{E}[K_N] = \sum_{i=1}^{N} \frac{\alpha}{\alpha + i - 1} \approx \alpha \log\left(1 + \frac{N}{\alpha}\right).$$



"Rich get richer": big tables attract more customers, and K grows only logarithmically in N.

Clustering

CRP is a prior over partition. To cluster data, we need:

$$p(z_i = k|y_i, y_{-i}, z_{-i}) \propto p(y_i|z_i = k, y_{-i}, z_{-i})p(z_i = k|z_{-i}) \tag{5}$$

Clustering

CRP is a prior over partition. To cluster data, we need:

$$p(z_i = k|y_i, y_{-i}, z_{-i}) \propto p(y_i|z_i = k, y_{-i}, z_{-i})p(z_i = k|z_{-i}) \tag{5}$$

- $p(y_i|z_i, y_{-i}, z_{-i})$: likelihood.

Clustering

CRP is a prior over partition. To cluster data, we need:

$$p(z_i = k|y_i, y_{-i}, z_{-i}) \propto p(y_i|z_i = k, y_{-i}, z_{-i})p(z_i = k|z_{-i}) \tag{5}$$

- $p(y_i|z_i, y_{-i}, z_{-i})$: likelihood.
- $p(z_i = k|z_{-i})$: CRP prior, treating datum i as the last customer ($n_{-i,k} = |\{j \neq i : z_j = k\}|$):

$$p(z_i = k|z_{-i}) = \frac{n_{-i,k}}{N - 1 + \alpha}, \qquad p(z_i = \text{new}|z_{-i}) = \frac{\alpha}{N - 1 + \alpha}.$$

Clustering

CRP is a prior over partition. To cluster data, we need:

$$p(z_i = k|y_i, y_{-i}, z_{-i}) \propto p(y_i|z_i = k, y_{-i}, z_{-i})p(z_i = k|z_{-i}) \tag{5}$$

- $p(y_i|z_i, y_{-i}, z_{-i})$: likelihood.
- $p(z_i = k|z_{-i})$: CRP prior, treating datum i as the last customer ($n_{-i,k} = |\{j \neq i : z_j = k\}|$):

$$p(z_i = k|z_{-i}) = \frac{n_{-i,k}}{N - 1 + \alpha}, \qquad p(z_i = \text{new}|z_{-i}) = \frac{\alpha}{N - 1 + \alpha}.$$

Update $i = 1, ..., N$ in turn. Why can datum i be the last customer?

Clustering

CRP is a prior over partition. To cluster data, we need:

$$p(z_i = k|y_i, y_{-i}, z_{-i}) \propto p(y_i|z_i = k, y_{-i}, z_{-i})p(z_i = k|z_{-i}) \tag{5}$$

- $p(y_i|z_i, y_{-i}, z_{-i})$: likelihood.
- $p(z_i = k|z_{-i})$: CRP prior, treating datum i as the last customer ($n_{-i,k} = |\{j \neq i : z_j = k\}|$):

$$p(z_i = k|z_{-i}) = \frac{n_{-i,k}}{N - 1 + \alpha}, \qquad p(z_i = \text{new}|z_{-i}) = \frac{\alpha}{N - 1 + \alpha}.$$

Update $i = 1, ..., N$ in turn. Why can datum i be the last customer?

Exchangeability: the CRP partition probability ignores arrival order.

Connection to the K component mixture

Return to the finite mixture with $\pi \sim \text{Dirichlet}(\alpha/K, ..., \alpha/K)$.
Integrating out π:

$$p(z_i = k | z_{-i}) = \frac{n_{-i,k} + \alpha/K}{N - 1 + \alpha}.$$

Connection to the K component mixture

Return to the finite mixture with $\pi \sim \text{Dirichlet}(\alpha/K, ..., \alpha/K)$.
Integrating out π:

$$p(z_i = k | z_{-i}) = \frac{n_{-i,k} + \alpha/K}{N - 1 + \alpha}.$$

Let K_+ be the number of occupied components. As $K \to \infty$:

- Occupied component k: $\dfrac{n_{-i,k} + \alpha/K}{N - 1 + \alpha} \to \dfrac{n_{-i,k}}{N - 1 + \alpha}$.
- All $K - K_+$ empty components together:
  $\dfrac{(K - K_+)\,\alpha/K}{N - 1 + \alpha} \to \dfrac{\alpha}{N - 1 + \alpha}$.

Connection to the K component mixture

Return to the finite mixture with $\pi \sim \text{Dirichlet}(\alpha/K, ..., \alpha/K)$.
Integrating out π:

$$p(z_i = k | z_{-i}) = \frac{n_{-i,k} + \alpha/K}{N - 1 + \alpha}.$$

Let K_+ be the number of occupied components. As $K \to \infty$:

- Occupied component k: $\dfrac{n_{-i,k} + \alpha/K}{N - 1 + \alpha} \to \dfrac{n_{-i,k}}{N - 1 + \alpha}$.
- All $K - K_+$ empty components together: $\dfrac{(K - K_+)\,\alpha/K}{N - 1 + \alpha} \to \dfrac{\alpha}{N - 1 + \alpha}$.

These are exactly the CRP conditionals: the DPMM is the $K \to \infty$ limit of the finite mixture (Neal 2000, Section 2).

Gibbs algorithm for DPMM (Algorithm 2)

For $t = 1, ..., T$ (MCMC chain length):

Gibbs algorithm for DPMM (Algorithm 2)

For $t = 1, ..., T$ (MCMC chain length):

1. For each datum $i = 1, ..., N$:
   - Remove y_i from its table (delete the table and its θ_k if empty).
   - Reassign z_i with probability

$$p(z_i = k | z_{-i}, y_i, \theta) \propto \begin{cases} n_{-i,k} \, F(y_i|\theta_k) & k \text{ existing}, \\ \alpha \int F(y_i|\theta) \, dH(\theta) & k \text{ new}. \end{cases}$$

   - If k is new, draw its $\theta \sim p(\theta|y_i) \propto F(y_i|\theta)H(\theta)$.

Gibbs algorithm for DPMM (Algorithm 2)

For $t = 1, ..., T$ (MCMC chain length):

1. For each datum $i = 1, ..., N$:
   - Remove y_i from its table (delete the table and its θ_k if empty).
   - Reassign z_i with probability

$$p(z_i = k | z_{-i}, y_i, \theta) \propto \begin{cases} n_{-i,k}\, F(y_i|\theta_k) & k \text{ existing}, \\ \alpha \int F(y_i|\theta)\, dH(\theta) & k \text{ new}. \end{cases}$$

   - If k is new, draw its $\theta \sim p(\theta|y_i) \propto F(y_i|\theta)H(\theta)$.
2. Sample $\theta_k | \{y_j : z_j = k\}$ for each table k.

Gibbs algorithm for DPMM (Algorithm 2)

For $t = 1, ..., T$ (MCMC chain length):

1. For each datum $i = 1, ..., N$:
   - Remove y_i from its table (delete the table and its θ_k if empty).
   - Reassign z_i with probability

$$p(z_i = k | z_{-i}, y_i, \theta) \propto \begin{cases} n_{-i,k}\, F(y_i|\theta_k) & k \text{ existing}, \\ \alpha \int F(y_i|\theta)\, dH(\theta) & k \text{ new}. \end{cases}$$

   - If k is new, draw its $\theta \sim p(\theta|y_i) \propto F(y_i|\theta)H(\theta)$.
2. Sample $\theta_k | \{y_j : z_j = k\}$ for each table k.

This is a Gibbs sampler: z_i and θ_k are drawn exactly from their full conditionals (possible because H is conjugate to F), so each update leaves the posterior invariant.

Non-conjugate H: θ_k can be updated by MH (Metropolis-within-Gibbs), but $\int F(y_i|\theta)dH(\theta)$ is intractable, so z_i needs Neal's Algorithms 5–8.

Collapsed Gibbs sampling

If H and F are conjugate, then we do not need to explicitly represent θ_k, we can marginalize it out to obtain the predictive distribution:

$$p(y_i|z_i = k, z_{-i}, y_{-i}) = \int F(y_i|\theta')p(\theta'|\{y_j : z_j = k, j \neq i\})d\theta' \quad (6)$$

where $p(\theta'|\{y_j : z_j = k, j \neq i\})$ is the posterior of θ_k given the other data points at table k.

Collapsed Gibbs sampling

If H and F are conjugate, then we do not need to explicitly represent θ_k, we can marginalize it out to obtain the predictive distribution:

$$p(y_i|z_i = k, z_{-i}, y_{-i}) = \int F(y_i|\theta')p(\theta'|\{y_j : z_j = k, j \neq i\})d\theta' \quad (6)$$

where $p(\theta'|\{y_j : z_j = k, j \neq i\})$ is the posterior of θ_k given the other data points at table k.

For a new table, integrate against the prior instead: $\int F(y_i|\theta')dH(\theta')$ (prior predictive).

Collapsed Gibbs sampling

Why collapse?

- Fewer variables to sample: only $z_{1:N}$, no θ_k.
- Each table is summarized by its sufficient statistics, updated as points move in and out.

Collapsed Gibbs sampling

Why collapse?

- Fewer variables to sample: only $z_{1:N}$, no θ_k.
- Each table is summarized by its sufficient statistics, updated as points move in and out.

Conjugate examples:

- F Normal, H Normal-Inverse-Gamma $\Rightarrow$ posterior Normal-Inverse-Gamma, predictive Student-t (next slide).
- σ^2 known, H Normal on $\mu \Rightarrow$ posterior Normal, predictive Normal.

Worked example: Normal-Inverse-Gamma predictive

Prior H: $\sigma^2 \sim \text{Inv-Gamma}(a_0, b_0)$, $\mu|\sigma^2 \sim N(m_0, \sigma^2/\kappa_0)$.

Table k holds n points (excluding y_i) with mean $\bar{y}$ and $S = \sum_j (y_j - \bar{y})^2$.
Posterior:

$$\kappa_n = \kappa_0 + n, \qquad m_n = \frac{\kappa_0 m_0 + n\bar{y}}{\kappa_n},$$

$$a_n = a_0 + \frac{n}{2}, \qquad b_n = b_0 + \frac{S}{2} + \frac{\kappa_0 n(\bar{y} - m_0)^2}{2\kappa_n}.$$

Worked example: Normal-Inverse-Gamma predictive

Prior H: $\sigma^2 \sim \text{Inv-Gamma}(a_0, b_0)$, $\mu|\sigma^2 \sim N(m_0, \sigma^2/\kappa_0)$.

Table k holds n points (excluding y_i) with mean $\bar{y}$ and $S = \sum_j (y_j - \bar{y})^2$.
Posterior:

$$\kappa_n = \kappa_0 + n, \qquad m_n = \frac{\kappa_0 m_0 + n\bar{y}}{\kappa_n},$$

$$a_n = a_0 + \frac{n}{2}, \qquad b_n = b_0 + \frac{S}{2} + \frac{\kappa_0 n(\bar{y} - m_0)^2}{2\kappa_n}.$$

Predictive (Student-t with $2a_n$ degrees of freedom):

$$p(y_i|z_i = k, z_{-i}, y_{-i}) = t_{2a_n}\left(y_i \,\middle|\, m_n, \frac{b_n(\kappa_n + 1)}{a_n \kappa_n}\right).$$

Worked example: Normal-Inverse-Gamma predictive

Prior H: $\sigma^2 \sim \text{Inv-Gamma}(a_0, b_0)$, $\mu|\sigma^2 \sim N(m_0, \sigma^2/\kappa_0)$.

Table k holds n points (excluding y_i) with mean $\bar{y}$ and $S = \sum_j (y_j - \bar{y})^2$.
Posterior:

$$\kappa_n = \kappa_0 + n, \qquad m_n = \frac{\kappa_0 m_0 + n\bar{y}}{\kappa_n},$$

$$a_n = a_0 + \frac{n}{2}, \qquad b_n = b_0 + \frac{S}{2} + \frac{\kappa_0 n(\bar{y} - m_0)^2}{2\kappa_n}.$$

Predictive (Student-t with $2a_n$ degrees of freedom):

$$p(y_i|z_i = k, z_{-i}, y_{-i}) = t_{2a_n}\left(y_i \,\middle|\, m_n, \frac{b_n(\kappa_n + 1)}{a_n \kappa_n}\right).$$

- New table: same formula with $n = 0$ (prior predictive).
- Small table: heavy tails, wide; large table: $\to N(\bar{y}, S/n)$.
- Only $(n, \sum_j y_j, \sum_j y_j^2)$ per table are needed.

Collapsed Gibbs algorithm for DPMM (Algorithm 3)

For $t = 1, ..., T$ (MCMC chain length):

Collapsed Gibbs algorithm for DPMM (Algorithm 3)

For $t = 1, ..., T$ (MCMC chain length):

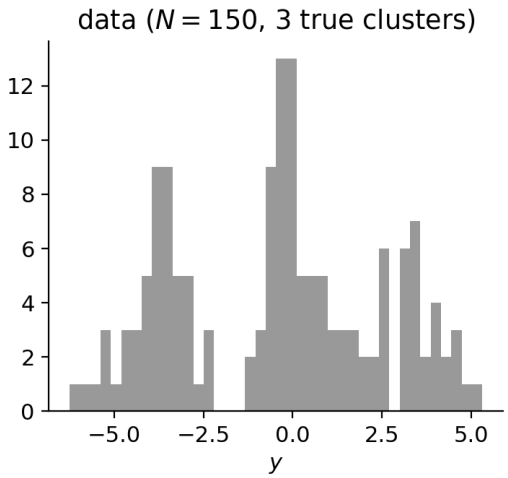
- For each datum $i = 1, ..., N$:
  - Remove y_i from its current table (update that table's sufficient statistics); delete the table if it becomes empty.
  - Reassign z_i with probability ($y_{k,-i} = \{y_j : z_j = k, j \neq i\}$)

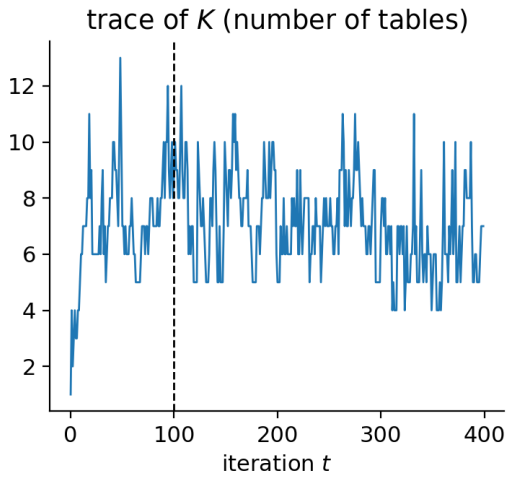
$$p(z_i = k | z_{-i}, y) \propto \begin{cases} n_{-i,k} \int F(y_i|\theta)\, p(\theta|y_{k,-i})\, d\theta & k \text{ existing,} \\ \alpha \int F(y_i|\theta)\, dH(\theta) & k \text{ new.} \end{cases}$$

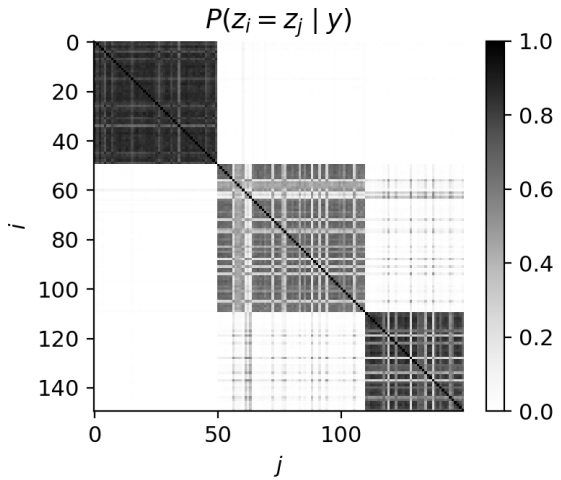
  - Add y_i to its new table and update the posterior $p(\theta_k | \{y_j : z_j = k\})$.

What does the sampler return?

Algorithm 3 on $N = 150$ points from 3 Gaussians ($\alpha = 1$):







- The output is a sequence of *partitions*. Labels switch between iterations, so summarize with label-free quantities such as the co-clustering probability $P(z_i = z_j | y)$.
- The co-clustering matrix recovers the 3 groups, but K hovers around 5–10 because of tiny transient clusters: do not read K naively as "the number of clusters".

Limitations and extensions

- **Slow mixing**: one-at-a-time updates struggle to merge two large clusters or split one, because it requires passing through low-probability intermediate states. Split-merge moves (Jain & Neal 2004) propose these changes in one step.

Limitations and extensions

- **Slow mixing**: one-at-a-time updates struggle to merge two large clusters or split one, because it requires passing through low-probability intermediate states. Split-merge moves (Jain & Neal 2004) propose these changes in one step.
- **Non-conjugate** H: auxiliary-variable Gibbs (Neal's Algorithm 8) represents the new-table option with m fresh draws from H, avoiding $\int F \, dH$.

Limitations and extensions

- **Slow mixing**: one-at-a-time updates struggle to merge two large clusters or split one, because it requires passing through low-probability intermediate states. Split-merge moves (Jain & Neal 2004) propose these changes in one step.
- **Non-conjugate** H: auxiliary-variable Gibbs (Neal's Algorithm 8) represents the new-table option with m fresh draws from H, avoiding $\int F \, dH$.
- **Choosing** α: α controls how many clusters we expect. Instead of fixing it, place a prior $\alpha \sim \text{Gamma}(a, b)$ and sample it within the Gibbs sweep (Escobar & West 1995).

Additional slides

- CRP as predictive distribution.
- Exchangeability.
- De Finetti's theorem.

Predictive distribution (CRP)

The predictive distribution, with G marginalized:

$$P(\theta_{N+1} \in A | \theta_{1:N}) = \mathbb{E}[1[\theta_{N+1} \in A] | \theta_{1:N}].$$

Since $\theta_{N+1}|G \sim G$, $\mathbb{E}[1[\theta_{N+1} \in A]|G] = G(A)$. By the tower property,

$$\mathbb{E}[1[\theta_{N+1} \in A]|\theta_{1:N}] = \int G(A)\, p(G|\theta_{1:N}) dG = \mathbb{E}[G(A)|\theta_{1:N}].$$

$$\mathbb{E}[G(A)|\theta_{1:N}] = H'(A) = \frac{\alpha}{\alpha + N} H(A) + \frac{1}{\alpha + N} \sum_{i=1}^{N} \delta_{\theta_i}(A).$$

Exchangeability

A sequence of random variables $Y_1, ..., Y_N$ is exchangeable if for every permutation σ:

$$p(y_1, ..., y_N) = p(y_{\sigma(1)}, ..., y_{\sigma(N)})$$

De Finetti's theorem

Any infinite exchangeable sequence of random variables can be viewed as conditionally i.i.d. draws given a random latent distribution G:

$$p(y_1, ..., y_N) = \int \prod_{i=1}^{N} p(y_i|G)\, dP(G) \quad \text{for all } N.$$

For the CRP, the latent G is exactly the DP draw.

De Finetti's theorem

Any infinite exchangeable sequence of random variables can be viewed as conditionally i.i.d. draws given a random latent distribution G:

$$p(y_1, ..., y_N) = \int \prod_{i=1}^{N} p(y_i|G)\, dP(G) \quad \text{for all } N.$$

For the CRP, the latent G is exactly the DP draw.

- i.i.d $\Rightarrow$ Exchangeability but reverse is not true.