BML Lecture #6

Bayesian nonparametrics

Julyan Arbel

Outline

Bayesian nonparametrics

- Pitman-Yor process, Gibbs-type process
- Hierarchical Dirichlet process
- Indian buffet process
- Practical: Dirichlet process mixture models in Pyro

Outline

Bayesian nonparametrics

- Pitman-Yor process, Gibbs-type process
- Hierarchical Dirichlet process
- Indian buffet process
- Practical: Dirichlet mixture models in Pyro

Bayesian deep learning

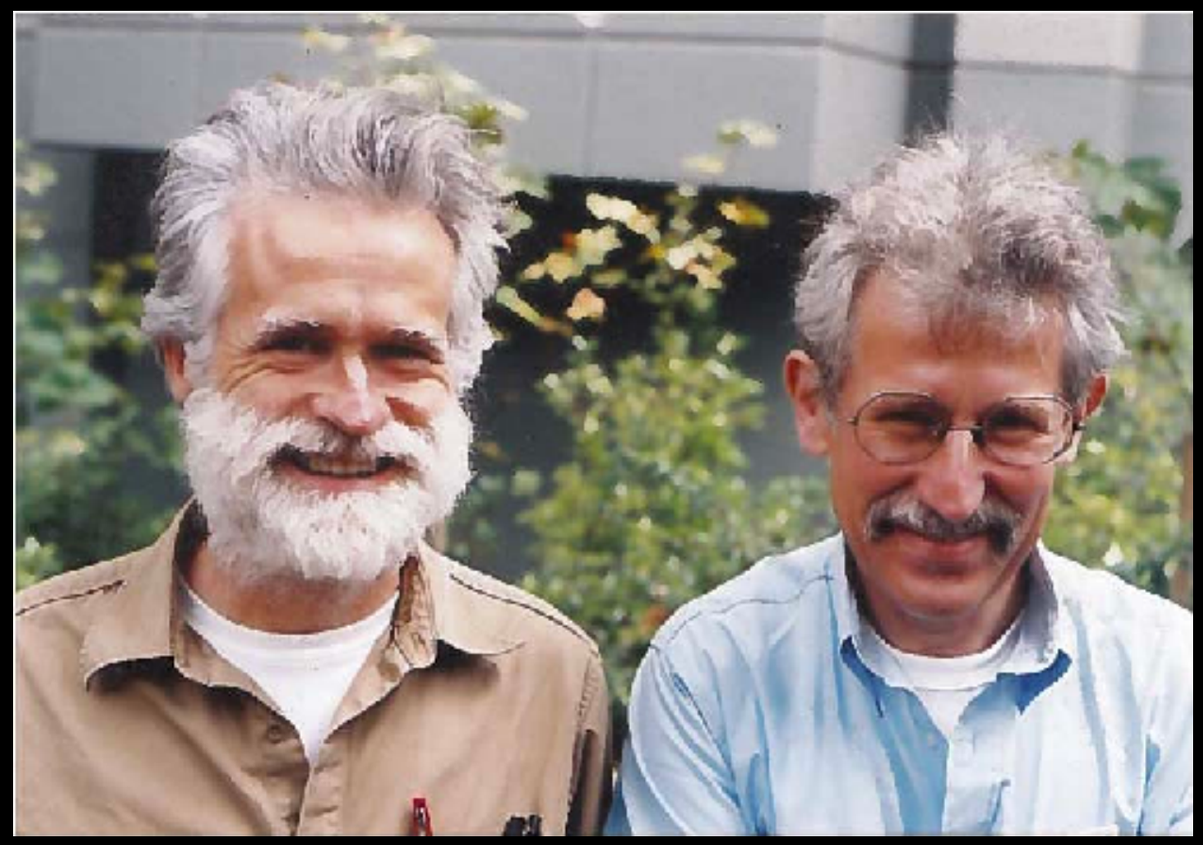
- Maximum a posteriori = Regularized maximum likehood
- Laplace approximation (MacKay, 1992, Neur. Comp.)
- Variational inference (Hinton and van Kamp, 1993, Barber & Bishop, 1998, NIPS)
- Monte Carlo dropout (Gal & Ghahramani, 2016, ICML)
- Practical: Bayesian neural networks in Pyro

Bayesian nonparametrics

Pitman-Yor process

- Dirichlet process
  - Ferguson, 1973

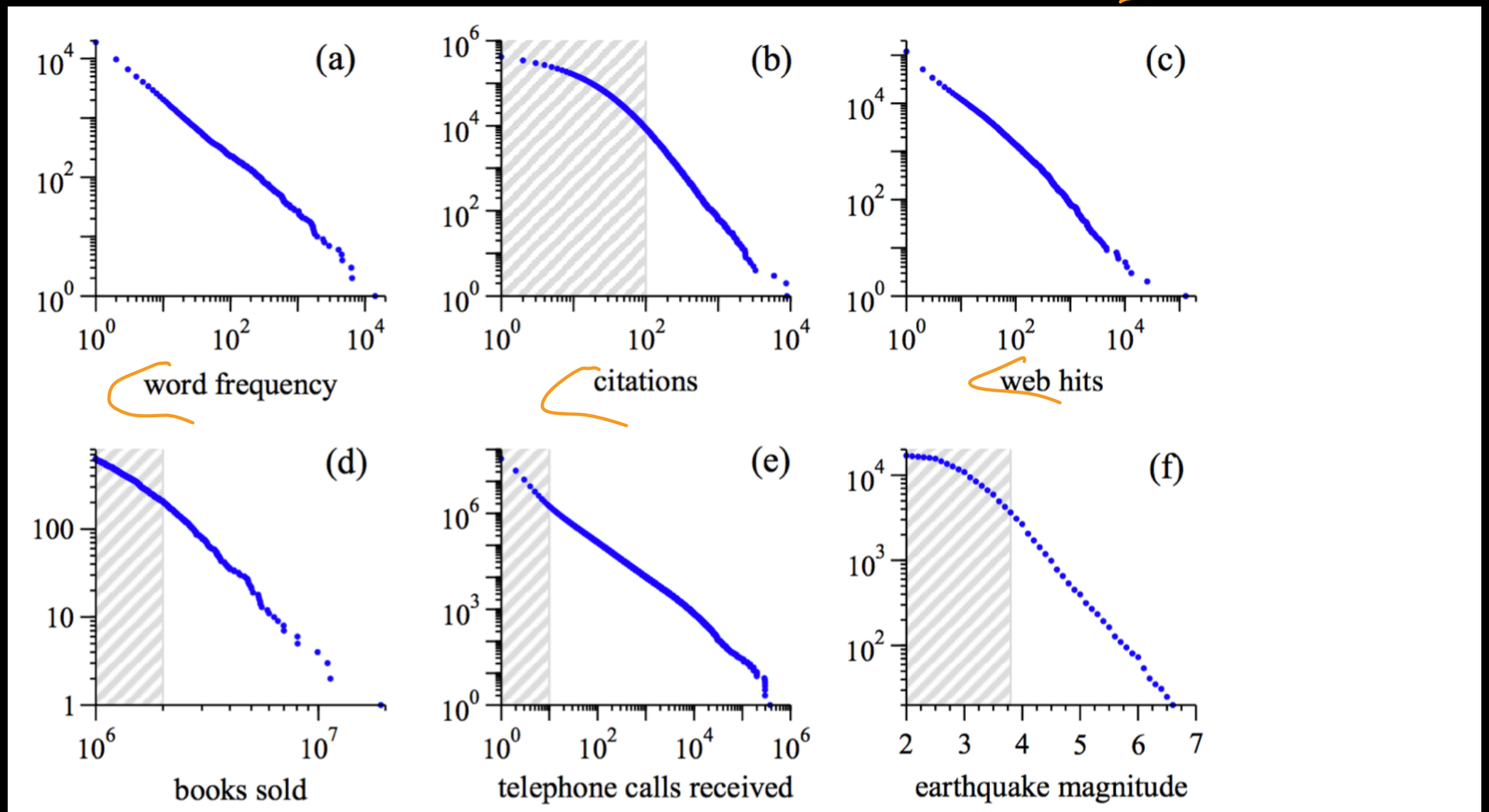
- Pitman-Yor process
  - Perman, Pitman & Yor, 1992
  - Pitman & Yor, 1997



Bayesian nonparametrics

Pitman-Yor process

- Depart from log #clusters to power-law #clusters



[Image from Newman [2005]]

Dirichlet process

predictive distr: $X_{n+1} | X_{1:n} \sim \frac{\alpha}{\alpha+n} P_0 + \frac{1}{\alpha+n} \sum_{i=1}^{n} \delta_{X_i}$

clusters: $K_n \approx \alpha \log n$

Pitman-Yor process

$$X_{n+1} | X_{1:n} \sim \frac{\alpha + \sigma K_n}{\alpha+n} P_0 + \frac{1}{\alpha+n} \sum_{j=1}^{K_n} (n_j - \sigma) \delta_{X_j^*}$$

X_j^*, $j = 1 \dots K_n$, nb of distinct obs.

n_j : # of obs. = to X_j^*

$\sigma \in [0, 1)$ → discount parameter

clusters: $K_n \approx S_{\alpha,\sigma} n^{\sigma}$

$S_{\alpha,\sigma}$: diversity of PY

→ Ex 2 : $\mathbb{E} K_n$, $\mathrm{Var}\, K_n$.

Bayesian nonparametrics

Pitman-Yor process: diversity

Power Law and σ-diversity

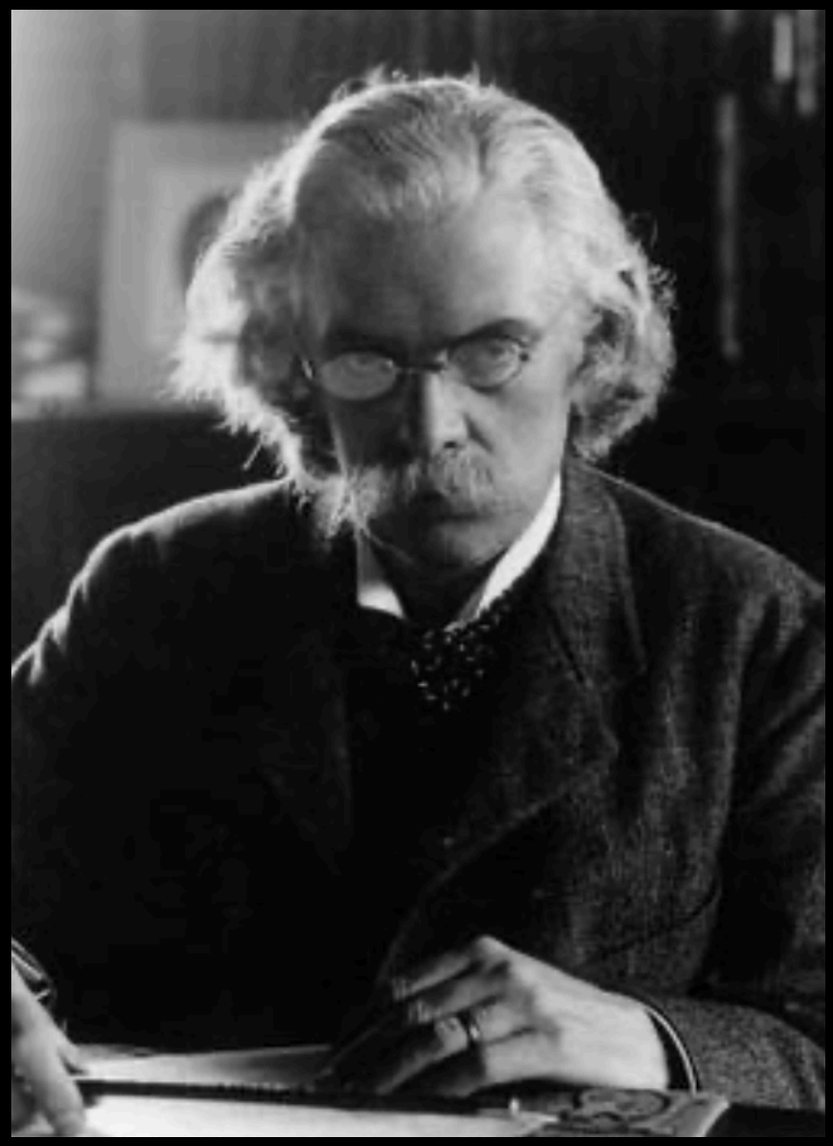
For $\sigma > 0$ we have the almost sure convergence

$$n^{-\sigma} K_n \to S_{\sigma,\alpha},$$

where $S_{\sigma,\alpha}$ is called σ-diversity of the PY, whose density is a polynomially tilted **Mittag–Leffler density** *(ML):*

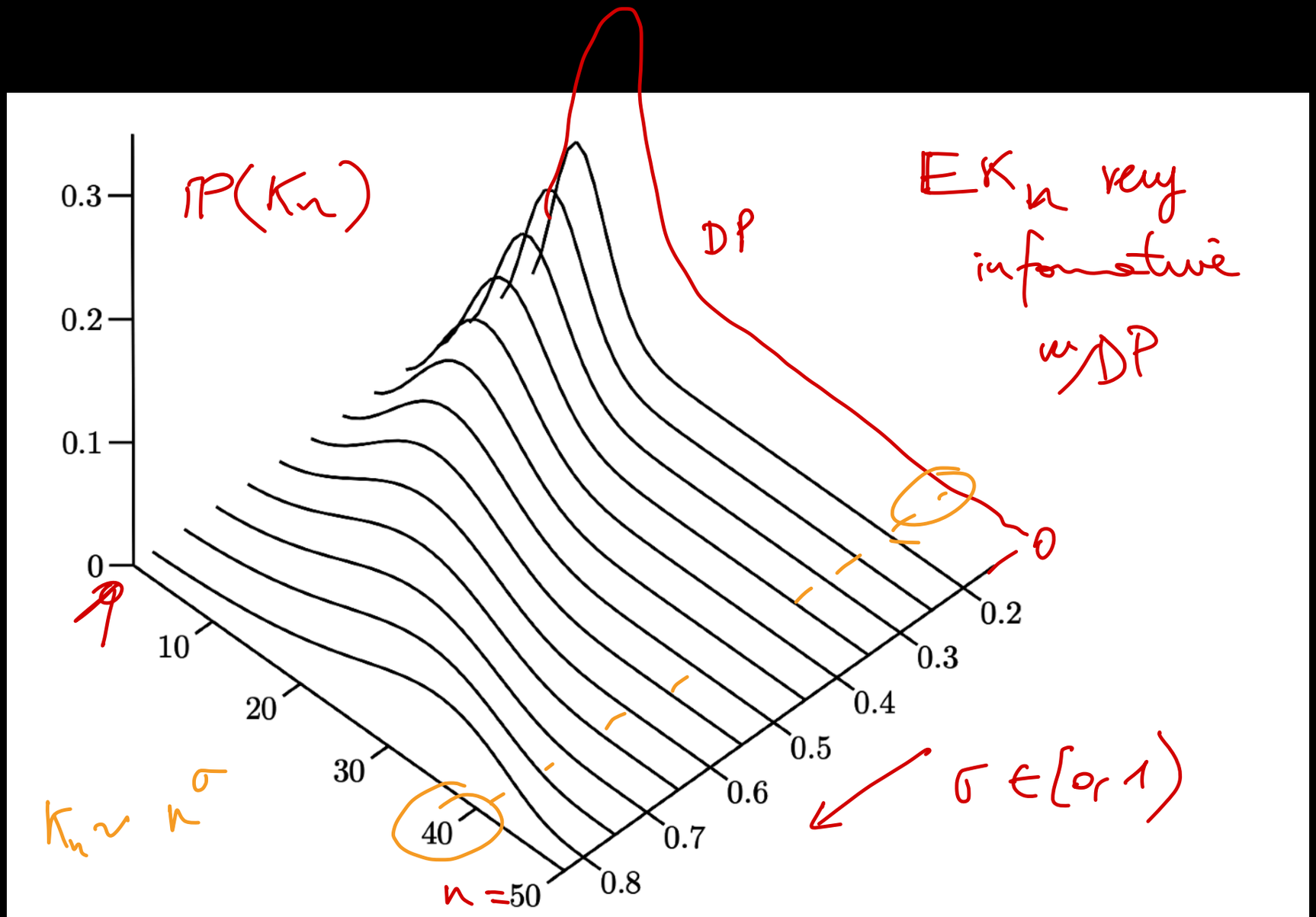
$$g_{\sigma,\alpha}(x) \propto x^{\alpha/\sigma} g_\alpha(x),$$

and g_α is ML density.



Bayesian nonparametrics

Pitman-Yor process: number of clusters



[Image by De Blasi et al. [2015]]

Notation $P \sim PY(P_0, \alpha, \sigma)$

CRP: Chinese Restaurant process = clustering distribution

aka EPPF: Exchangeable partition proba. function

$n_1, \dots, n_k$ k for K_n

CRP: $p(n_1, \dots, n_K)$

Cust. seated at new table wp: $\frac{\alpha + \sigma K_n}{\alpha + n}$ discovery probability

——— occupied table wp: $\frac{(n_j - \sigma}{\alpha + n}$ ...

$$p(n_1, \dots, n_k) = \sigma^K \frac{(1 + \frac{\alpha}{\sigma})_{k-1}}{(1+\alpha)_{n-1}} \prod_{j=1}^{k} (1-\sigma)_{n_j - 1}$$

$\sigma = 0 \rightarrow$ DP CRP

Ewens Sampling Formula $p(m_1, \dots, m_n)$

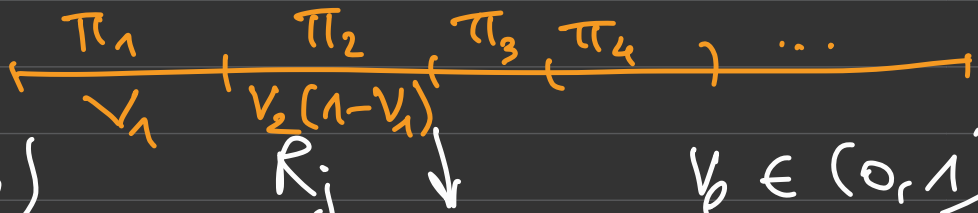
Stick-breaking representation for PY, P_0 is non atomic

$\theta_j \overset{iid}{\sim} P_0$ ⫫ $V_j \overset{ind}{\sim} \text{Beta}(1-\sigma, \alpha + j\sigma)$ (DP w/ $\sigma = 0$)

$\pi_1 = V_1, \quad j > 1, \quad \boxed{\pi_j = V_j \prod_{\ell=1}^{j-1} (1 - V_\ell)}$ $\quad \mathbb{E} V_j = \frac{1-\sigma}{1+\alpha+(j-1)\sigma}$

Proposition: $P := \sum_{j=1}^{\infty} \pi_j \delta_{\theta_j} \sim PY(P_0, \alpha, \sigma)$

→ discrete random probability measure. $\mathbb{E} R_j = (1-\dots)^j$



Left-over $R_j = \prod_{\ell=1}^{j} (1 - V_\ell)$ $\quad V_\ell \in (0,1)$

$$\mathbb{E} R_i = \mathbb{E} \prod_{j=1}^{i} (1 - V_\ell) = \prod \mathbb{E}(1 - V_\ell) = \prod \frac{\alpha + \sigma j}{1 + \alpha + (j-1)\sigma}$$

$$= \prod \left(1 - \frac{1-\sigma}{1+\alpha+(j-1)\sigma}\right) \qquad \log(1+x) \leq x$$

$$\log \mathbb{E} R_i = \sum \log\left(\frac{\quad}{\quad}\right) \leq -\sum \frac{1-\sigma}{1+\alpha+(j-1)\sigma} \longrightarrow -\infty$$

$\Rightarrow \mathbb{E} R_i \longrightarrow 0$

$R_i \longrightarrow 0$ a.s.

A, B measurable subsets of Θ. {Remember for DP: used $P(A) \sim Be(\dots,\dots)$

$$\mathbb{E}[P(A)] = \mathbb{E}\left[\sum_{i=1}^{\infty} \pi_i \delta_{\theta_i}(A)\right]$$

$$= \sum \mathbb{E}\,\pi_i \, \mathbb{E}[\delta_{\theta_i}(A)] = \sum \mathbb{E}[\pi_i] P_0(A) \quad \text{because } \theta_i \sim P_0$$

$$= P_0(A) \sum \mathbb{E}[\pi_i] = P_0(A) \, \mathbb{E}\Big[\underbrace{\textstyle\sum \pi_i}_{1}\Big] = P_0(A).$$

P_0 serves as a base measure.

$$\mathrm{Cov}(P(A), P(B)) = \underbrace{\mathbb{E}[P(A)P(B)]}_{(*)} - \underbrace{\mathbb{E}[P(A)]}_{P_0(A)} \underbrace{\mathbb{E}(P(B))}_{P_0(B)}$$

$X_1, X_2 \mid P \overset{iid}{\sim} P$ $\quad (*) = \mathbb{P}(X_1 \in A, X_2 \in B)$

$$= \underbrace{\mathbb{P}(X_1 \in A)}_{\mathbb{E}[P(A)] = P_0(A)} \mathbb{P}(X_2 \in B \mid X_1 \in A) \overset{(**)}{} \qquad K = 1, \; n = 1$$

predict. $X_2 \mid X_1 \sim \frac{\alpha+\sigma}{\alpha+1} P_0 + \frac{1-\sigma}{\alpha+1} \delta_{X_1}$

$$(**) = \frac{\alpha+\sigma}{\alpha+1} P_0(B) + \frac{1-\sigma}{\alpha+1} P_0(B \mid A) \to \frac{P_0(A \cap B)}{P_0(A)}$$

$$\mathrm{Var}(P(A)) = \frac{1-\sigma}{\alpha+1} P_0(A)(1 - P_0(A))$$

$$\mathrm{Cov}(P(A), P(B)) = -\frac{1-\sigma}{\alpha+1}\left[P_0(A)P_0(B) - P_0(A \cap B)\right]$$

Posterior characterization for PY

Assume $\begin{cases} X_1, \dots, X_n \mid P \overset{iid}{\sim} P \\ P \sim PY(P_0, \alpha, \sigma) \end{cases}$

$$P \mid X_1, \dots, X_n \overset{d}{=} (1-q_n)\, \underbrace{P_n}_{PY} + q_n \underbrace{\sum_{j=1}^{k_n} p_j^* \delta_{x_j^*}}_{\text{dist. obs.}}$$

$$\begin{cases} \blacksquare\ q_n \sim \text{Beta}(n - \sigma k_n, \alpha + \sigma k_n) \in (0,1) \\ \blacksquare\ P_n \sim PY(P_0, \sigma, \alpha + \sigma k_n) \\ \blacksquare\ (p_1^*, p_2^*, \dots, p_{k_n}^*) \sim \text{Dir}(n_1 - \sigma, \dots, n_{k_n} - \sigma) \end{cases}$$

Bayesian nonparametrics

Gibbs-type process

- Pitman, 2003

Define via EPPF (CRP):

$$p(n_1, \dots, n_k) = V_{n,k} \prod_{j=1}^{k} (1-\sigma)_{n_j}$$

$n \geq 1$

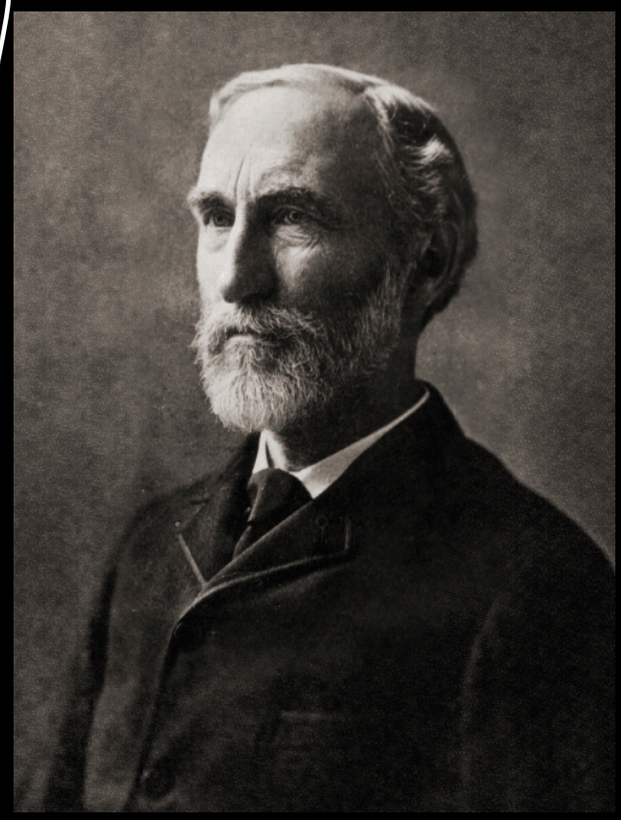
$1 \leq k \leq n$ $\qquad \sigma \in (0,1)$

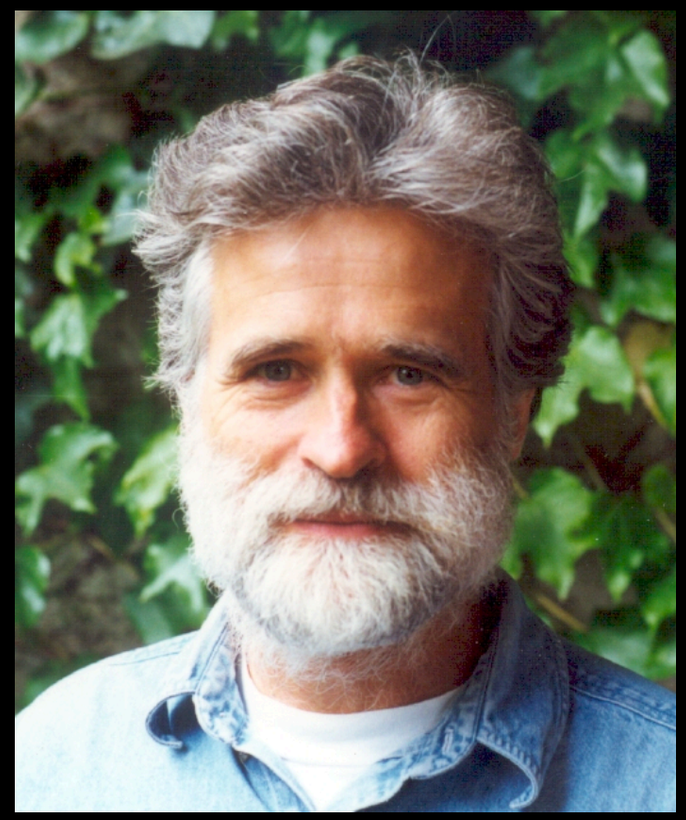
$$\begin{cases} V_{1,1} = 1 \\ V_{n,k} = (n - \sigma k) V_{n+1,k} + V_{n+1,k+1} \end{cases} \iff$$

Recover PY: $\quad V_{n,k} = \sigma^k \dfrac{\left(1 + \frac{\alpha}{\sigma}\right)_{k-1}}{(1+\alpha)_{n-1}}$

DP: $\sigma \Leftarrow 0$

Predictive: $\quad X_{n+1} \mid X_{1:n} \sim \dfrac{V_{n+1,k+1}}{V_{n,k}} P_0 + \dfrac{V_{n+1,k}}{V_{n,k}} \sum_{j=1}^{k_n} (n_j - \sigma) \delta_{x_j^*}$





Jim Pitman

discovery proba. depends (n, K_n)

old quantity (n, k_n)

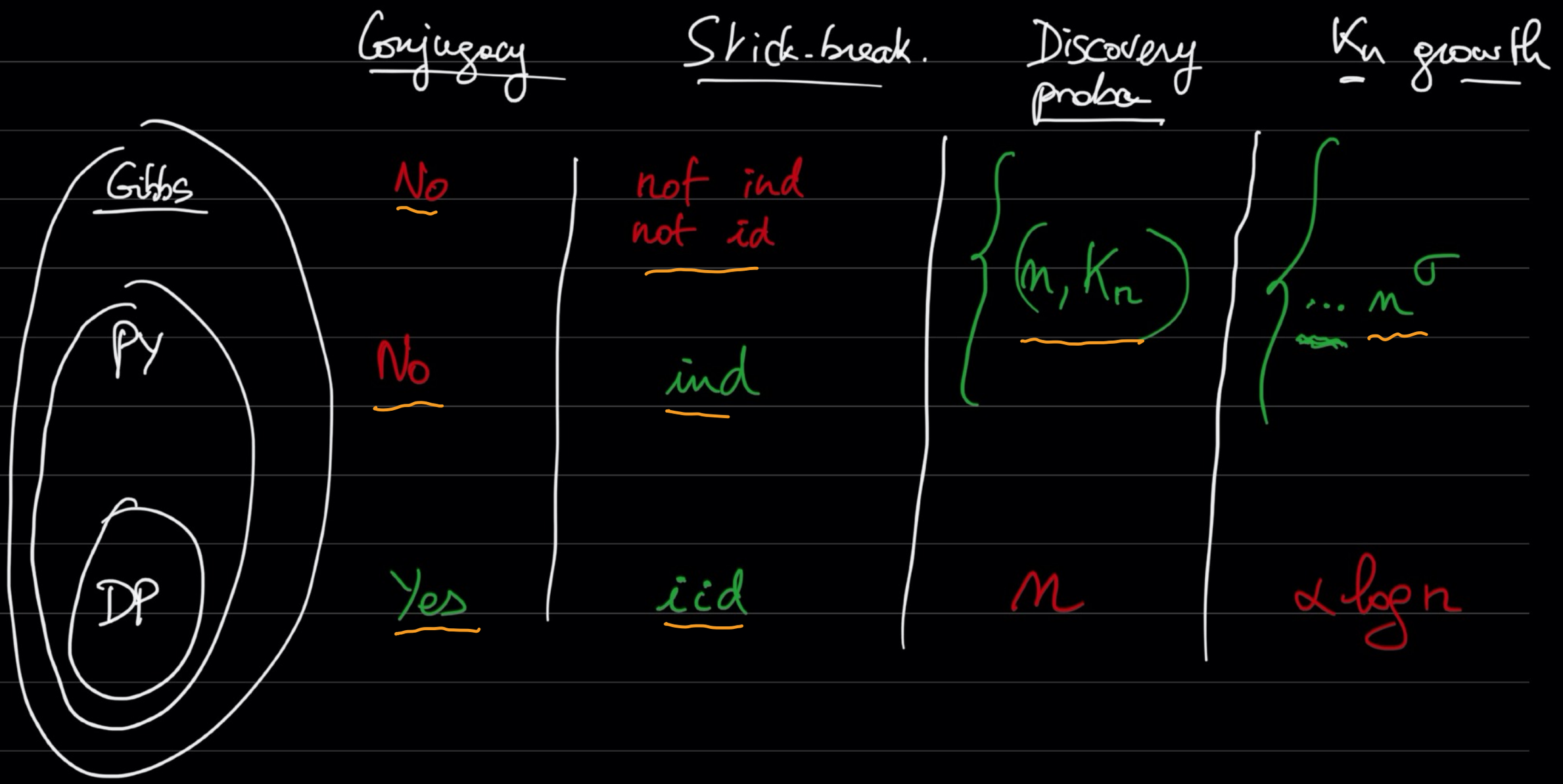
Stick-breaking : $V_j \mid V_{j-1}, \dots, V_1 \sim g(v_j \mid v_{j-1}, \dots, v_1)$

One class of Gibbs-type : normalized generalized gamma

NGG $\neq$ PY

Bayesian nonparametrics

Dirichlet ⊂ Pitman-Yor ⊂ Gibbs-type



Bayesian nonparametrics

Hierarchical Dirichlet process

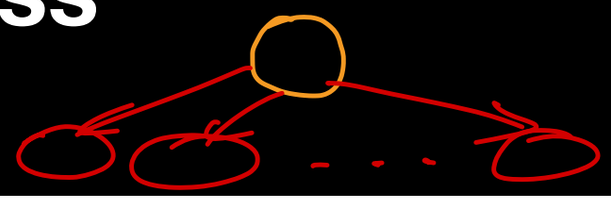
- Extension of Latent Dirichlet Allocation (Blei, Ng, and Jordan, 2003, JMLR)
- to infinite dimensional parameter space (Teh, Jordan, Beal, and Blei, JASA, 2006)




Bayesian nonparametrics

Hierarchical Dirichlet process

Ghosal & van der Vaart (2017)

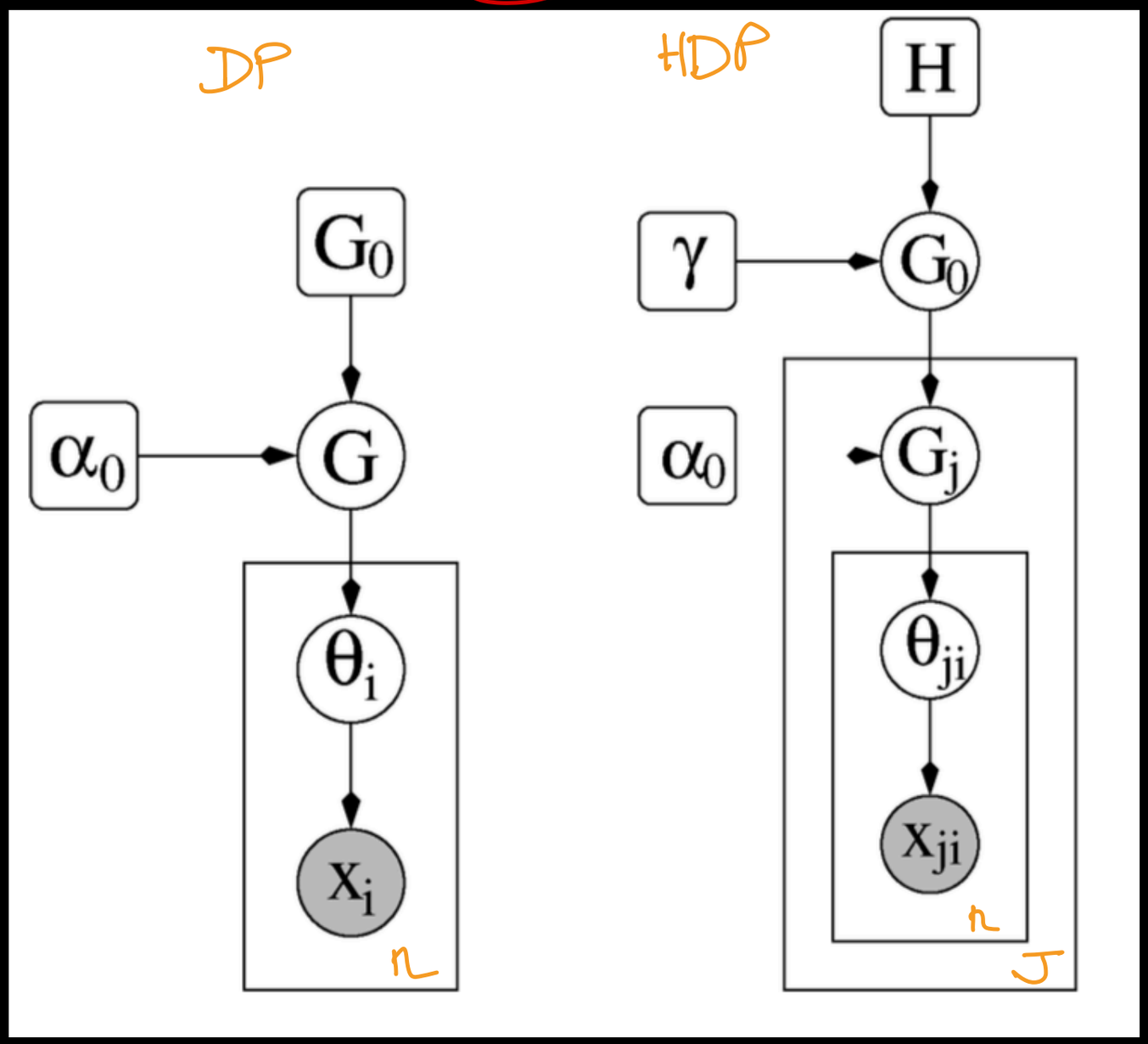


Chinese Restaurant Franchise

Stick-breaking

DP: $\begin{cases} G \mid \alpha_0, G_0 \sim DP(\alpha_0, G_0) \\ \theta_i \mid G \overset{iid}{\sim} G \\ X_i \mid \theta_i \overset{ind}{\sim} F(x_i \mid \theta_i) \end{cases}$

HDP: $\begin{cases} G_0 \mid \gamma, H \sim DP(\gamma, H) \\ G_j \mid \alpha_0, G_0 \overset{iid}{\sim} DP(\alpha_0, G_0) \quad * \\ \theta_{ji} \mid G_j \sim G_j, j \dots \\ X_{ji} \mid \theta_{ji} \overset{ind}{\sim} F(x_{ji} \mid \theta_{ji}) \end{cases}$



Something new here:

base measure G_0 : atomic.

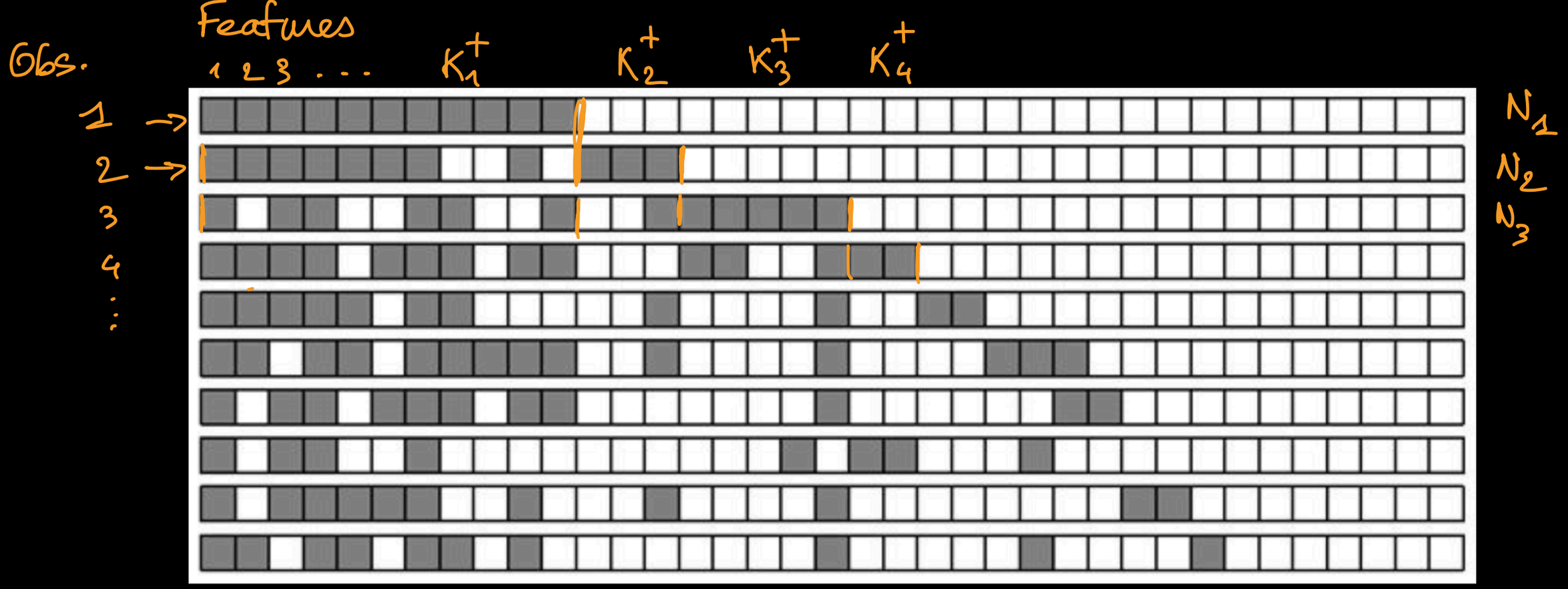
Bayesian nonparametrics

Indian buffet process

- Extension of Dirichlet process (Ferguson, 2003, AOS) to multiple classes/allocation models
- Ghahramani & Griffiths, 2005.

Bayesian nonparametrics

Indian buffet process



Poisson lemmas :

- additivity $N_1 \sim \text{Pois}(\gamma_1)$, $N_2 \sim \text{Pois}(\gamma_2)$ $\perp\!\!\!\perp$ $\Longrightarrow N_1 + N_2 \sim \text{Pois}(\gamma_1 + \gamma_2)$

- thinning : $N \sim \text{Pois}(\gamma)$, $X_1, \dots, X_N \overset{iid}{\sim} \text{Ber}(p) \Longrightarrow \sum_{i=1}^{N} X_i \sim \text{Pois}(\gamma p)$

Generative model for $IBP(\gamma)$, $\gamma > 0$

- Customer 1 : choose $N_1 \sim Pois(\gamma)$ dishes $^{(C1)}$
- Customer 2 : - choose some dishes of C1 wp $\frac{1}{2}$
  - ——— new dishes $K_2^+ \sim Pois(\frac{\gamma}{2})$

$\Rightarrow N_2 \sim Pois(\frac{\gamma}{2}) + Pois(\frac{\gamma}{2}) = Pois(\gamma)$

⋮

- Customer i : denote $1, \dots, K$ all sampled dishes
  $m_1, \dots, m_K$ their numbers
  chooses previous dishes wp $\frac{1}{i}$, ie dish j wp $\frac{m_j}{i}$
  - new dishes $Pois(\frac{\gamma}{i})$

$$N_i \sim \underbrace{Pois\left(\frac{i-1}{i}\gamma\right)}_{\text{old}} + Pois\left(\frac{\gamma}{i}\right) = Pois(\gamma).$$

Total nb of dishes : $K_n = K_1^+ + \dots + K_n^+ = Pois(\gamma) + Pois(\frac{\gamma}{2}) \dots + Pois(\frac{\gamma}{n})$ $\sim Pois(\gamma \log n)$

$= Pois\left(\gamma\left(1 + \frac{1}{2} + \dots + \frac{1}{n}\right)\right) \xrightarrow[n \to \infty]{} \infty$

Bayesian nonparametrics

Practical: Pyro, pyro.ai

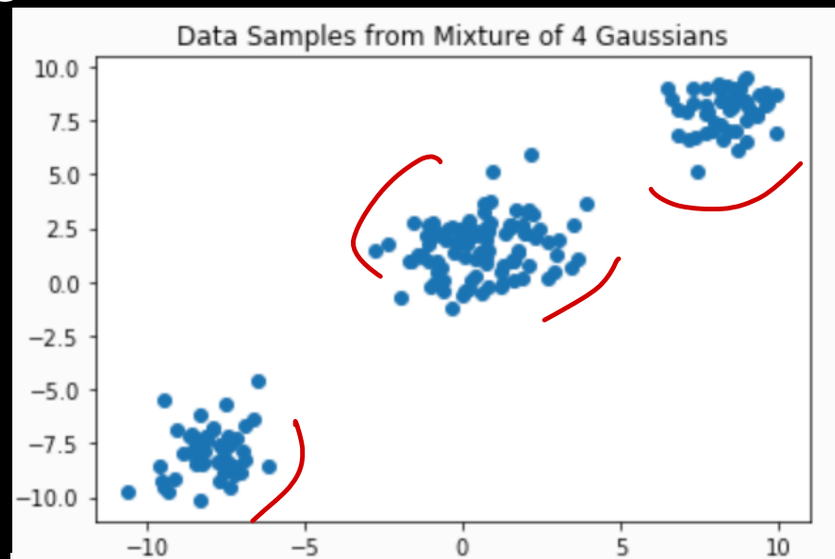
Pyro is a universal probabilistic programming language (PPL) written in Python and supported by PyTorch on the backend. Pyro enables flexible and expressive deep probabilistic modeling, unifying the best of modern deep learning and Bayesian modeling. It was designed with these key principles:

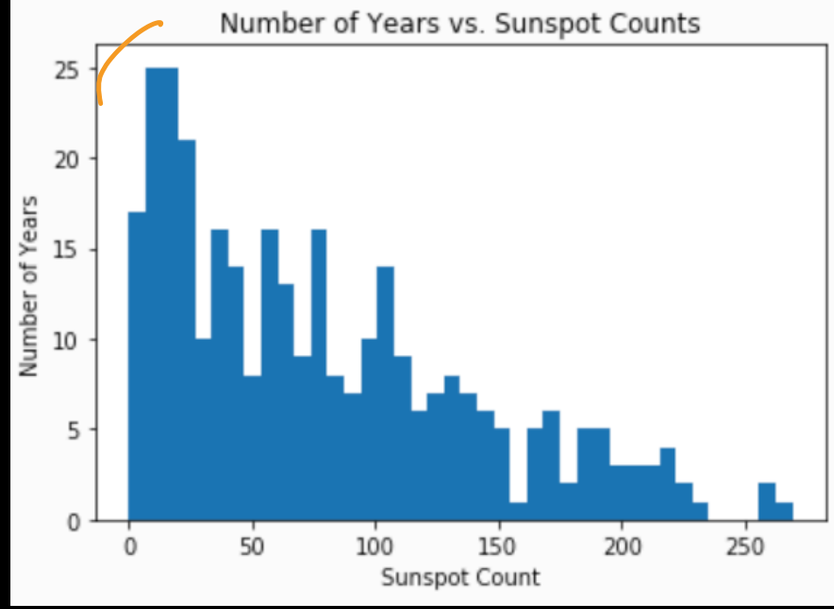
Universal: Pyro can represent any computable probability distribution.
Scalable: Pyro scales to large data sets with little overhead.
Minimal: Pyro is implemented with a small core of powerful, composable abstractions.
Flexible: Pyro aims for automation when you want it, control when you need it.

Bayesian nonparametrics

Practical: Dirichlet process mixture models in Pyro

- https://pyro.ai/examples/dirichlet_process_mixture.html
- Implements Variational inference based on stick-breaking
  - On Synthetic Mixture of Gaussians
  - On Long Term Solar Observations
- Objectives:
  - Install Pyro & PyTorch (or work on Colab)
  - Run the code
  - Write your own code for Pitman-Yor mixture models





Bayesian deep learning

Bayesian neural networks

- Maximum a posteriori = Regularized maximum likehood
- Laplace approximation (MacKay, 1992, Neur. Comp.)
- Variational inference (Hinton and van Kamp, 1993, Barber & Bishop, 1998, NIPS)
- Monte Carlo dropout (Gal & Ghahramani, 2016, ICML)
- Implementation: Pyro & PyTorch

Bayesian deep learning

Pyro example on Bayesian neural networks

- Demonstrates how to use NUTS to do inference on a simple (small) Bayesian neural network with two hidden layers.