Priors for Random Count Matrices with Random or Fixed Row Sums

Mingyuan Zhou

Joint work with Oscar Madrid and James Scott

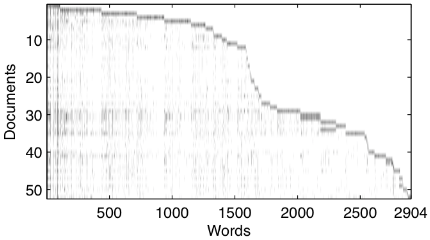
IROM Department, McCombs School of Business
Department of Statistics and Data Sciences
The University of Texas at Austin

10th Conference on Bayesian Nonparametrics
Raleigh, NC, June 23, 2015

Table of Contents

Where do random count matrices appear?

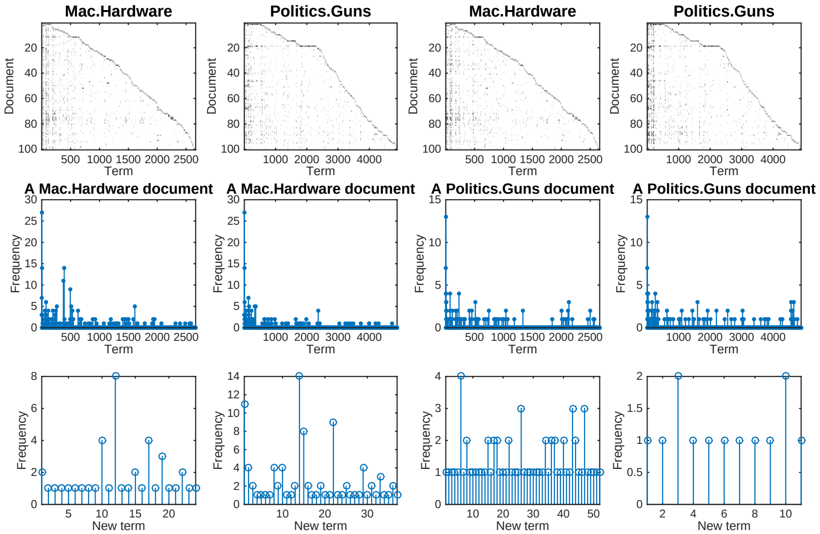
- Directly observable random count matrices:
  - Text analysis: document-word count matrix
  - DNA-sequencing: sample-gene count matrix
  - Social network analysis: user-venue check-in count matrix
  - Consumer behavior: consumer-product count matrix
- Latent random count matrices:
  - Topic models: document-topic count matrix (the sum of each row is the length of the corresponding document)
  - Hidden Markov models: state-state transition count matrix



Motivations to Study Random Count Matrices

- Lack of priors to describe random count matrices with a potentially infinite number of rows/columns.
- A naive Bayes classifier often requires a predetermined vocabulary shared across all categories, and has to ignore previously unseen features/terms.
- How to calculate the predictive distribution of a new count vector that brings previously unseen terms?
- Priors for random count matrices can be used to construct priors for mixed-membership modeling.

Representation of a cout vector under a count matrix



Infinite random count matrices to be studied

- No natural upper bound on the number of rows or columns
- Conditionally independent rows, i.i.d. columns
- Parallel column-wise construction
- Sequential row-wise constructions
- Predictive distribution of a new row count vector that brings new features
- Random count matrices with fixed row sums for mixed-membership modeling

Related prior distributions

- Prior distributions for a univariate count:
  - Poisson distribution
  - Logarithmic distribution
  - Negative binomial distribution
  - Beta-negative binomial distribution
- Generating a random count vector:
  - Chinese restaurant process, Pitman-Yor process
  - Normalized random measures with independent increments
  - (Sample-size dependent) exchangeable partition probability functions
- Generating an infinite random binary matrix:
  - Indian buffet (beta-Bernoulli) process
- Generating an infinite random count matrix:
  - How?

Steps to construct an infinite random count matrix

- Choose a completely random measure G, a draw from which consists of countably infinite atoms $G = \sum_{k=1}^{\infty} r_k \delta_{\omega_k}$.
- For $X_j := \sum_{k=1}^{\infty} n_{jk} \delta_{\omega_k}$, draw counts $n_{jk} \sim f(r_k, \theta_j)$, where f denotes a count distribution parameterized by r_k and θ_j.
- Denote $\boldsymbol{n}_{:k} = (n_{1k}, \ldots, n_{Jk})^T$ and $n_{\cdot k} = \sum_{j=1}^{J} n_{jk}$.
- The count matrix $\mathbf{N}_J$ is constructed by organizing all the nonzero column count vectors, $\{\boldsymbol{n}_{:k}\}_{k:n_{\cdot k}>0}$, in an arbitrary order into a random count matrix.

In practice, we cannot instantiate all the atoms of G. Thus we will have to marginalize G out from $\{X_j\}_{1,J}$ to construct $\mathbf{N}_J$.

Gamma-Poisson process

- $X_j \sim \mathrm{PP}(G), \; G \sim \Gamma\mathrm{P}(G_0, 1/c)$
- Conditional likelihood:

$$p(\{X_j\}_{1,J} \mid G) = \prod_{k=1}^{\infty} \frac{r_k^{n_{\cdot k}}}{\prod_{j=1}^{J} n_{jk}!} e^{-Jr_k} = e^{-JG(\Omega \setminus \mathcal{D})} \prod_{k=1}^{K_J} \frac{r_k^{n_{\cdot k}} e^{-Jr_k}}{\prod_{j=1}^{J} n_{jk}!}$$

- To marginalize G out, one may separate Ω to the absolution continuous space and points of discontinuity, and then apply the characteristic function to $G(\Omega \setminus \mathcal{D})$ and the Lévy measure of G to each point of discontinuity.
- The $\{X_j\}_{1,J}$ to $\mathbf{N}_J$ is a one-to-$(K_J!)$ mapping, thus

$$f(\mathbf{N}_J \mid \gamma_0, c) = \frac{\mathbb{E}_G[p(\{X_j\}_{1,J} \mid G)]}{K_J!}$$

Exchangeable rows and i.i.d. columns

- Distribution for the count matrix:

$$f(\mathbf{N}_J \mid \gamma_0, c) = \frac{\gamma_0^{K_J} \exp\left[-\gamma_0 \ln(\frac{J+c}{c})\right]}{K_J!} \prod_{k=1}^{K_J} \frac{\frac{\Gamma(n_{\cdot k})}{(J+c)^{n_{\cdot k}}}}{\prod_{j=1}^{J} n_{jk}!}$$

- Row exchangeable, column i.i.d:

$$\boldsymbol{n}_{:k} \sim \text{Multinomial}(n_{\cdot k}, 1/J, \ldots, 1/J),$$
$$n_{\cdot k} \sim \text{Log}[J/(J+c)],$$
$$K_J \sim \text{Pois}\left\{\gamma_0 \left[\ln(J+c) - \ln(c)\right]\right\}.$$

- Closed-form Gibbs sampling update equations for model parameters

Exchangeable rows and i.i.d. columns

- Distribution for the count matrix:

$$f(\mathbf{N}_J \mid \gamma_0, c) = \frac{\gamma_0^{K_J} \exp\left[-\gamma_0 \ln(\frac{J+c}{c})\right]}{K_J!} \prod_{k=1}^{K_J} \frac{\frac{\Gamma(n_{\cdot k})}{(J+c)^{n_{\cdot k}}}}{\prod_{j=1}^{J} n_{jk}!}$$

- Row exchangeable, column i.i.d:

$$\boldsymbol{n}_{:k} \sim \text{Multinomial}(n_{\cdot k}, 1/J, \ldots, 1/J),$$
$$n_{\cdot k} \sim \text{Log}[J/(J+c)],$$
$$K_J \sim \text{Pois}\left\{\gamma_0 \left[\ln(J+c) - \ln(c)\right]\right\}.$$

- Closed-form Gibbs sampling update equations for model parameters

Sequential row-wise construction

- Sequential row-wise construction:

$$p(\mathbf{N}_{J+1}^{+} \mid \mathbf{N}_J, \boldsymbol{\theta}) = \frac{f(\mathbf{N}_{J+1} \mid \boldsymbol{\theta})}{f(\mathbf{N}_J \mid \boldsymbol{\theta})} = \frac{K_J! K_{J+1}^{+}!}{K_{J+1}!} \prod_{k=1}^{K_J} \mathrm{NB}\left(n_{(J+1)k}; n_{\cdot k}, \frac{1}{J+c+1}\right)$$
$$\times \prod_{k=K_J+1}^{K_{J+1}} \mathrm{Log}\left(n_{(J+1)k}; \frac{1}{J+c+1}\right)$$
$$\times \mathrm{Pois}\left\{K_{J+1}^{+}; \gamma_0\left[\ln(J+c+1) - \ln(J+c)\right]\right\}.$$

- To add a new row to $\mathbf{N}_J \in \mathbb{Z}^{J \times K_J}$:
  - First, draw count $\mathrm{NB}(n_{\cdot k}, p_{J+1})$ at each existing column
  - Second, draw $K_{J+1}^{+} \sim \mathrm{Pois}\left\{\gamma_0\left[\ln(J+c+1) - \ln(J+c)\right]\right\}$ number of new columns
  - Third, draw $\mathrm{Log}(p_{J+1})$ random count at each new column
- The combinatorial coefficient arises as the newly added columns are inserted into the original ones at random locations, with their relative orders preserved.

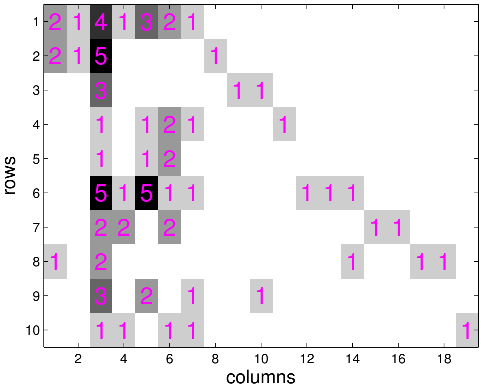


Figure: A sequentially constructed negative binomial process random count matrix $\mathbf{N}_J \sim \mathrm{NBPM}(\gamma_0, c)$.

Gamma-negative binomial process

- Gamma-negative binomial process:

$$X_j \sim \mathrm{NBP}(G, p_j), \;\; G \sim \Gamma\mathrm{P}(G_0, 1/c)$$

- Conditional likelihood:

$$p(\{X_j\}_{1,J} | G, \boldsymbol{p}) = \prod_{k=1}^{\infty} \prod_{j=1}^{J} \frac{\Gamma(n_{jk} + r_k)}{n_{jk}!\Gamma(r_k)} p_j^{n_{jk}} (1 - p_j)^{r_k}$$

- Augmented likelihood:

$$p(\{X_j, L_j\}_{1,J} \mid G, \boldsymbol{p}) = e^{-q_{\cdot} G(\Omega \setminus \mathcal{D})} \prod_{k=1}^{K_J} r_k^{l_{\cdot k}} e^{-q_{\cdot} r_k} \left(\prod_{j=1}^{J} \frac{|s(n_{jk}, l_{jk})| p_j^{n_{jk}}}{n_{jk}!} \right),$$

where $q_j = -\ln(1 - p_j)$ and $q_{\cdot} = \sum_{j=1}^{J} q_j$.

- Distribution for the (augmented) count matrix:

$$f(\mathbf{N}_J, \mathbf{L}_J \mid \boldsymbol{\theta}) = \frac{\gamma_0^{K_J} \exp\left[-\gamma_0 \ln(\frac{c+q_\cdot}{c})\right]}{K_J!} \prod_{k=1}^{K_J} \frac{\Gamma(l_{\cdot k})}{(c+q_\cdot)^{l_{\cdot k}}} \left(\prod_{j=1}^{J} \frac{|s(n_{jk}, l_{jk})| p_j^{n_{jk}}}{n_{jk}!} \right)$$

- Row heterogeneity, column i.i.d.:

$$n_{jk} = \sum_{t=1}^{l_{jk}} n_{jkt}, \;\; n_{jkt} \sim \mathrm{Log}(p_j),$$
$$(l_{1k}, \ldots, l_{Jk}) \sim \mathrm{Mult}(l_{\cdot k}, q_1/q_\cdot, \ldots, q_J/q_\cdot),$$
$$l_{\cdot k} \sim \mathrm{Log}[q_\cdot/(c+q_\cdot)],$$
$$K_J \sim \mathrm{Pois}\{\gamma_0[\ln(c+q_\cdot) - \ln(c)]\}.$$

- Closed-form Gibbs sampling update equations for model parameters.

- Predictive distribution of a new row:

$$
\begin{aligned}
p(\mathbf{N}_{J+1}^{+}, \mathbf{L}_{J+1}^{+} \mid \mathbf{N}_J, \mathbf{L}_J, \boldsymbol{\theta}) = \tfrac{K_J! K_{J+1}^{+}!}{K_{J+1}!} \textstyle\prod_{k=1}^{K_{J+1}} \mathrm{SumLog}\left(l_{(J+1)k}, p_{J+1}\right) \\
\times \textstyle\prod_{k=1}^{K_J} \mathrm{NB}\left(l_{(J+1)k}; l_{\cdot k}, \tfrac{q_{J+1}}{c+q_\cdot+q_{J+1}}\right) \\
\times \textstyle\prod_{k=K_J+1}^{K_{J+1}} \mathrm{Log}\left(l_{(J+1)k}; \tfrac{q_{J+1}}{c+q_\cdot+q_{J+1}}\right) \\
\times \mathrm{Pois}\left\{K_{J+1}^{+}; \gamma_0 \left[\ln(c+q_\cdot+q_{J+1}) - \ln(c+q_\cdot)\right]\right\}.
\end{aligned}
$$

- To add a new row:
  - Draw $\mathrm{NB}(l_{\cdot k}, \frac{q_{J+1}}{c+q_\cdot+q_{J+1}})$ tables at existing columns (dishes)
  - Draw $K_{J+1}^{+} \sim \mathrm{Pois}\left\{\gamma_0 \left[\ln(c+q_\cdot+q_{J+1}) - \ln(c+q_\cdot)\right]\right\}$ new dishes
  - Draw $\mathrm{Log}(\frac{q_{J+1}}{c+q_\cdot+q_{J+1}})$ tables at each new dish
  - Draw $\mathrm{Log}(p_{J+1})$ customers at each table and aggregate the counts across the tables of the same dish as $n_{(J+1)k} = \sum_{t=1}^{l_{(J+1)k}} n_{(J+1)kt}$

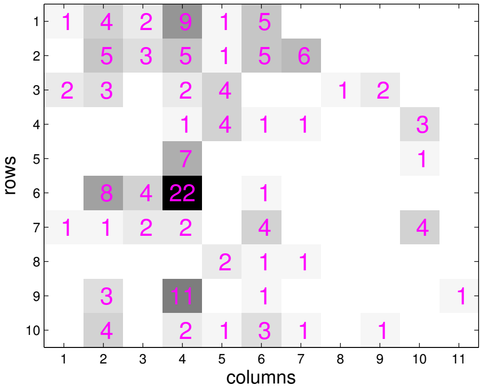


Figure: A sequentially constructed gamma-negative binomial process random count matrix $\mathbf{N}_J \sim \text{GNBPM}(\gamma_0, c, p_1, \cdots, p_J)$.

Beta-negative binomial process

- Beta-negative binomial process:

$$X_j \sim \mathrm{NBP}(r_j, B),\ \ B \sim \mathrm{BP}(c, B_0)$$

- Conditional likelihood:

$$p(\{X_j\}_{1,J}|B, \boldsymbol{r}) = e^{-p_* r_\cdot} \prod_{k=1}^{K_J} p_k^{n_{\cdot k}} (1-p_k)^{r_\cdot} \prod_{j=1}^{J} \frac{\Gamma(n_{jk}+r_j)}{n_{jk}!\Gamma(r_j)}$$

where

$$p_* = -\sum_{k=K_J+1}^{\infty} \ln(1-p_k)$$

- Distribution for the count matrix:

$$f(\mathbf{N}_J \mid \gamma_0, c, \boldsymbol{r}) = \frac{\gamma_0^{K_J} e^{-\gamma_0[\psi(c+r.)-\psi(c)]}}{K_J!} \times \prod_{k=1}^{K_J} \frac{\Gamma(n_{\cdot k})\Gamma(c+r.)}{\Gamma(c+n_{\cdot k}+r.)} \prod_{j=1}^{J} \frac{\Gamma(n_{jk}+r_j)}{n_{jk}!\Gamma(r_j)}$$

- Row heterogeneity, column i.i.d.:

$$\begin{aligned} \boldsymbol{n}_{:k} &\sim \mathrm{DirMult}(n_{\cdot k}, r_1, \cdots, r_J) \\ n_{\cdot k} &\sim \mathrm{Digam}(r., c) \\ K_J &\sim \mathrm{Pois}\{\gamma_0\,[\psi(c+r.) - \psi(c)]\,\} \end{aligned}$$

where $\mathrm{Digam}(n|r,c) = \frac{1}{\psi(c+r)-\psi(c)} \frac{\Gamma(r+n)\Gamma(c+r)}{n\Gamma(c+n+r)\Gamma(r)}$

- Closed-form Gibbs sampling update equations for model parameters

Ice cream buffet process (a.k.a., multi-scoop IBP [Zhou et al., 2012] and negative binomial IBP [Heaukulani & Roy, 2013])

- Sequential row-wise construction:

$$p(\mathbf{N}_{J+1}^{+} \mid \mathbf{N}_J) = \frac{K_J! K_{J+1}^{+}!}{K_{J+1}!} \prod_{k=1}^{K_J} \mathrm{BNB}(n_{(J+1)k}; r_{J+1}, n_{\cdot k}, c + r_\cdot) \times \prod_{k=K_J+1}^{K_{J+1}} \mathrm{Digam}(n_{(J+1)k}; r_{J+1}, c + r_\cdot) \times \mathrm{Pois}\left\{K_{J+1}^{+}; \gamma_0 \left[\psi(c + r_\cdot + r_{J+1}) - \psi(c + r_\cdot)\right]\right\}.$$

- To add a new row:
  - Customer $J+1$ takes $n_{(J+1)k} \sim \mathrm{BNB}(r_{J+1}, n_{\cdot k}, c + r_\cdot)$ number of scoops at an existing ice cream (column).
  - The customer further selects $K_{J+1}^{+} \sim \mathrm{Pois}\{\gamma_0 [\psi(c + r_\cdot + r_{J+1}) - \psi(c + r_\cdot)]\}$ new ice creams out of the buffet line.
  - The customer takes $n_{(J+1)k} \sim \mathrm{Digam}(r_{J+1}, c + r_\cdot)$ number of scoops at each new ice cream.

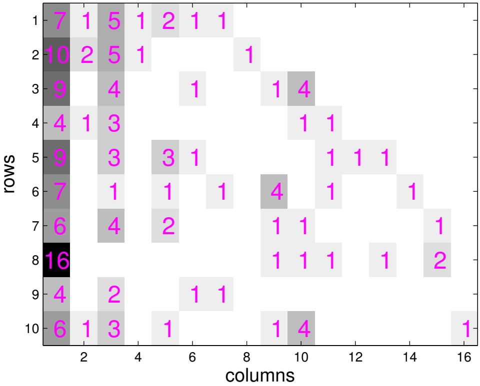


Figure: A sequentially constructed beta-negative binomial process random count matrix $\mathbf{N}_J \sim \mathrm{BNBPM}(\gamma_0, c, r_1, \cdots, r_J)$.

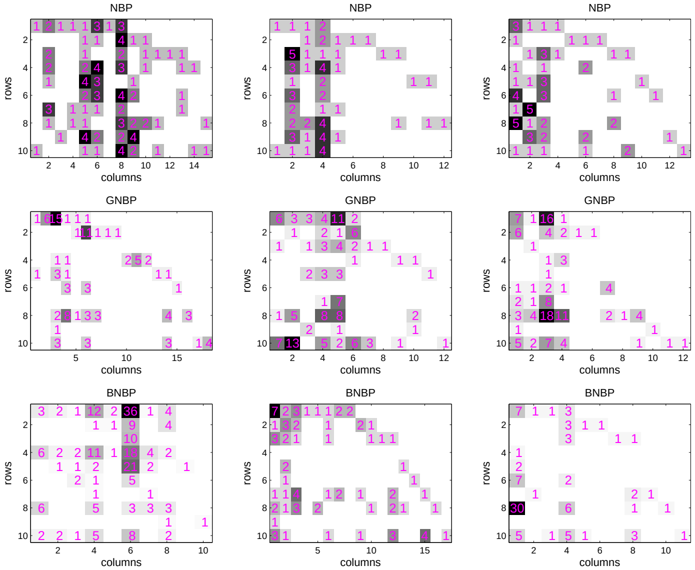
Comparison of different priors

Model	Number of new columns K_{J+1}^{+}	Counts in existing columns	Counts in new columns
NBP	$\mathrm{Pois}\{\gamma_0[\ln(J+c+1)-\ln(J+c)]\}$	$\mathrm{NB}[n_{\cdot k}, 1/(J+c+1)]$	$\mathrm{Log}[1/(J+c+1)]$
GNBP	$\mathrm{Pois}\{\gamma_0[\ln(c+q_{\cdot}+q_{J+1})-\ln(c+q_{\cdot})]\}$	$\mathrm{GNB}(l_{\cdot k}, c+q_{\cdot}, p_{J+1})$	$\mathrm{LogLog}(c+q_{\cdot}, p_{J+1})$
BNBP	$\mathrm{Pois}\{\gamma_0[\psi(c+r_{\cdot}+r_{J+1})-\psi(c+r_{\cdot})]\}$	$\mathrm{BNB}(r_{J+1}, n_{\cdot k}, c+r_{\cdot})$	$\mathrm{Digam}(r_{J+1}, c+r_{\cdot})$

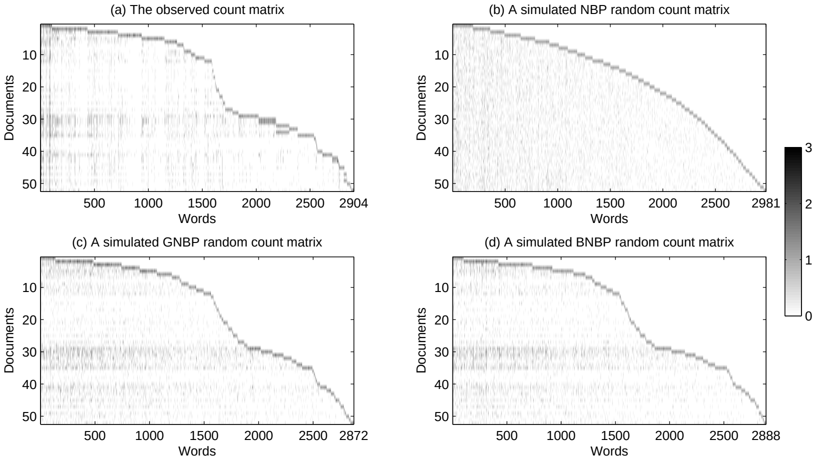
$$\text{NBP:}\quad \mathrm{Var}[n_{(J+1)k}] = \mathbb{E}[n_{(J+1)k}] + \frac{\mathbb{E}^2[n_{(J+1)k}]}{n_{\cdot k}}$$

$$\text{GNBP:}\quad \mathrm{Var}[n_{(J+1)k}] = \frac{\mathbb{E}[n_{(J+1)k}]}{1-p_{J+1}} + \frac{\mathbb{E}^2[n_{(J+1)k}]}{l_{\cdot k}}$$

$$\text{BNBP:}\quad \mathrm{Var}[n_{(J+1)k}] = \frac{\mathbb{E}[n_{(J+1)k}]}{\frac{c+r_{\cdot}}{n_{\cdot k}+c+r_{\cdot}-1}} + \frac{\mathbb{E}^2[n_{(J+1)k}]}{\frac{n_{\cdot k}(c+r_{\cdot}-2)}{n_{\cdot k}+c+r_{\cdot}-1}}$$


NBP
GNBP
BNBP
rows
columns

Training and posterior predictive checking



Predictive distribution of a new row vector

- The predictive distribution of a row vector $\boldsymbol{n}_{J+1}$ is

$$p(\boldsymbol{n}_{J+1} \mid \mathbf{N}_J, \boldsymbol{\theta}) = \frac{p(\mathbf{N}_{J+1}^{+} \mid \mathbf{N}_J, \boldsymbol{\theta})}{K_{J+1}^{+}!} \tag{1}$$

$$= \frac{K_J!}{K_{J+1}!} \frac{\frac{K_{J+1}!}{K_J! K_{J+1}^{+}!} f(\mathbf{N}_{J+1} \mid \boldsymbol{\theta})}{f(\mathbf{N}_J \mid \boldsymbol{\theta})} . \tag{2}$$

- The normalizing constant $1/K_{J+1}^{+}!$ in (1) arises because a realization of $\mathbf{N}_{J+1}^{+}$ to $\boldsymbol{n}_{J+1}$ is one-to-many, with $K_{J+1}^{+}!$ distinct orderings of these new columns.
- The normalizing constant $K_J!/K_{J+1}!$ in (2) arises because there are $\prod_{i=1}^{K_{J+1}^{+}}(K_J + i)! = K_{J+1}!/K_J!$ ways to insert the K_{J+1}^{+} new columns into the original ordered K_J columns, which is again a one-to-many mapping.

- Each category is summarized as a random count matrix $\mathbf{N}_J$; columns with all zeros are excluded.
- Gibbs sampling is used to infer the parameters $\boldsymbol{\theta}$ that generate $\mathbf{N}_J$; to represent the posterior of $\boldsymbol{\theta}$, S MCMC samples $\{\boldsymbol{\theta}^{[s]}\}_{1,S}$ are collected.
- For a testing row count vector $\boldsymbol{n}_{J+1}$, its predictive likelihood given $\mathbf{N}_J$ is calculated via Monte Carlo integration using

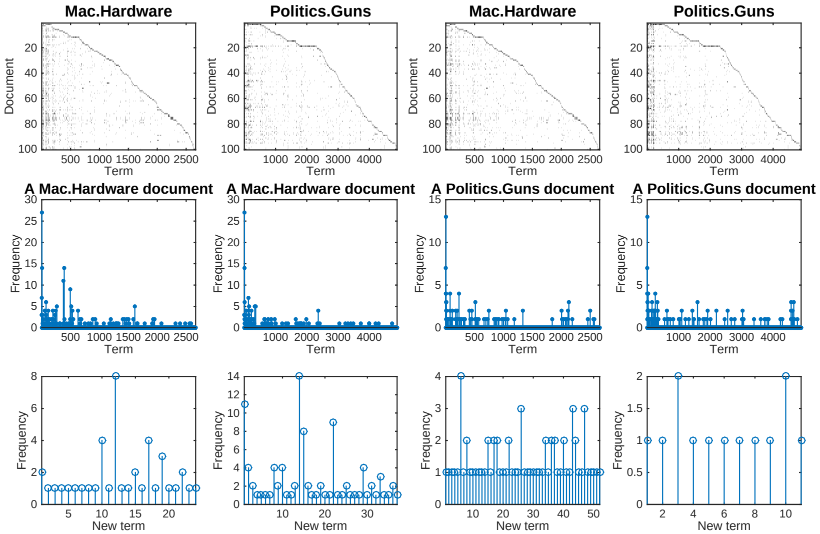
$$p(\boldsymbol{n}_{J+1} \mid \mathbf{N}_J) = \frac{1}{S}\sum_{s=1}^{S} \frac{p(\mathbf{N}_{J+1}^{+} \mid \mathbf{N}_J, \boldsymbol{\theta}^{[s]})}{K_{J+1}^{+}!}$$

for both the NBP and BNBP, and using

$$p(\boldsymbol{n}_{J+1} \mid \mathbf{N}_J) = \frac{1}{S}\sum_{s=1}^{S} \frac{p(\mathbf{N}_{J+1}^{+} \mid \mathbf{N}_J, \mathbf{L}_J^{[s]}, \boldsymbol{\theta}^{[s]})}{K_{J+1}^{+}!}$$

for the GNBP.

Infinite vocabulary naive Bayes classifiers



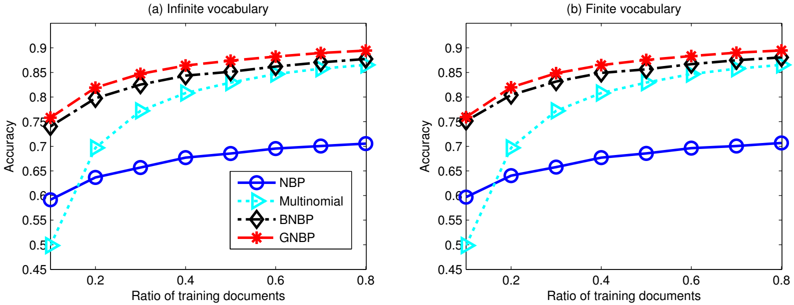


Figure: Document categorization results on the 20 Newsgroup dataset with (a) an unconstrained vocabulary that can grow to infinite, and (b) an predetermined finite vocabulary of size $V = 61{,}188$, using the negative binomial process (NBP), gamma-negative binomial process (GNBP), and beta-negative binomial process (BNBP). The results of the multinomial naive Bayes classifier using Laplace smoothing are included for comparison.

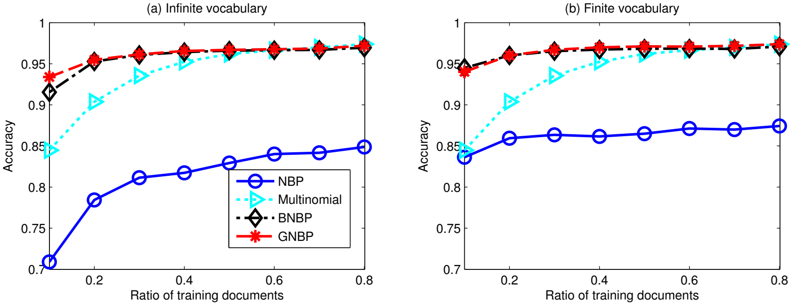


Figure: Analogous plots to the plots in the previous Figure for the TDT2 dataset. The predetermined finite vocabulary has the size of $V = 36{,}771$.

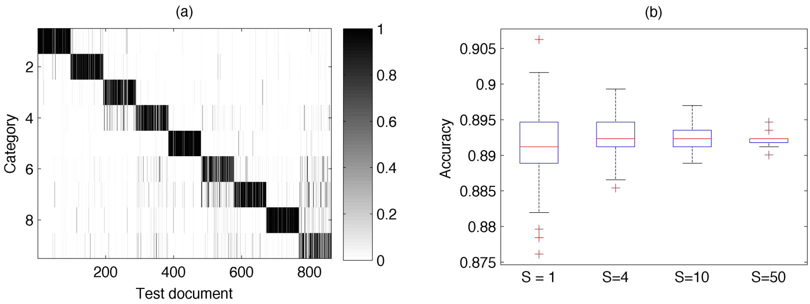


Figure: (a) The predicted probabilities of the test documents under different categories for the CNAE-9 dataset, using the GNBP nonparametric Bayesian naive Bayes classifier with 20% of the documents of each of the nine categories used for training. (b) Boxplots of the categorization accuracies; each accuracy is computed with $S = 1$, $S = 5$, $S = 40$, or $S = 100$ MCMC samples.

Construct EPPFs for mixture modeling using priors for random count vectors [Zhou and Walker, 2014]

- One way to generate a random count vector $(n_1, \ldots, n_\ell)$:
  - Draw ℓ, the length of the vector, and then draw independent positive random counts $\{n_k\}_{1,\ell}$.
- Another way to generate such a random count vector:
  - Draw a total count n, and partition it using an EPPF, resulting in a set of exchangeable categorical variables $\boldsymbol{z} = (z_1, \ldots, z_n)$.
  - Map $\boldsymbol{z}$ to a random positive count vector $(n_1, \ldots, n_\ell)$, where $n_k := \sum_{i=1}^n \delta(z_i = k) > 0$.
- Both ways lead to the same distributed $(n_1, \ldots, n_\ell)$ if and only if $P(n_1, \ldots, n_\ell, n) = \frac{1}{\ell!} \frac{n!}{\prod_{k=1}^\ell n_k!} P(\boldsymbol{z}, n)$
- (Sample size dependent) EPPF for Mixture modeling:

$$P(\boldsymbol{z}|n) = \frac{P(\boldsymbol{z}, n)}{P(n)} = \left[\frac{1}{\ell!} \frac{n!}{\prod_{k=1}^\ell n_k!}\right]^{-1} \frac{P(n_1, \ldots, n_\ell, n)}{P(n)}$$

Construct EPPFs for mixed-membership modeling using priors for random count matrices [Zhou 2014]

- BNBP random count matrix prior

$$f(\mathbf{N}_J|\boldsymbol{r},\gamma_0,c) = \frac{\gamma_0^{K_J} e^{-\gamma_0[\psi(c+r_\cdot)-\psi(c)]}}{K_J!} \prod_{k=1}^{K_J} \frac{\Gamma(n_{\cdot k})\Gamma(c+r_\cdot)}{\Gamma(c+n_{\cdot k}+r_\cdot)} \prod_{j=1}^{J} \frac{\Gamma(n_{jk}+r_j)}{n_{jk}!\Gamma(r_j)}$$

- With $\boldsymbol{z} = (z_{11},\ldots,z_{Jm_J})$ and $n_{jk} = \sum_{i=1}^{m_j} \delta(z_{ji}=k)$, the joint distribution of a column count vector $\boldsymbol{m} = (m_1,\ldots,m_J)^T$ and its partition into a column exchangeable latent random count matrix with K_J nonempty columns can be expressed as

$$f(\boldsymbol{z},\boldsymbol{m}|\boldsymbol{r},\gamma_0,c) = \left[\frac{1}{K_J!}\prod_{j=1}^{J}\frac{m_j!}{\prod_{k=1}^{K_J} n_{jk}!}\right]^{-1} f(\mathbf{N}_J|\boldsymbol{r},\gamma_0,c)$$

$$= \frac{\gamma_0^{K_J} e^{-\gamma_0[\psi(c+r_\cdot)-\psi(c)]}}{\prod_{j=1}^{J} m_j!} \prod_{k=1}^{K_J}\left[\frac{\Gamma(n_{\cdot k})\Gamma(c+r_\cdot)}{\Gamma(c+n_{\cdot k}+r_\cdot)}\prod_{j=1}^{J}\frac{\Gamma(n_{jk}+r_j)}{\Gamma(r_j)}\right]$$

- The BNBP's EPPF for mixed-membership modeling:

$$f(\boldsymbol{z}|\boldsymbol{m},\boldsymbol{r},\gamma_0,c) = \frac{f(\boldsymbol{z},\boldsymbol{m}|\boldsymbol{r},\gamma_0,c)}{f(\boldsymbol{m}|\boldsymbol{r},\gamma_0,c)} = \frac{1}{\frac{1}{K_J!}\prod_{j=1}^{J}\frac{m_j!}{\prod_{k=1}^{K_J} n_{jk}!}}\frac{f(\mathbf{N}_J|\boldsymbol{r},\gamma_0,c)}{f(\boldsymbol{m}|\boldsymbol{r},\gamma_0,c)}$$

- The prediction rule is simple:

$$P(z_{ji}|\boldsymbol{z}^{-ji},\boldsymbol{m},\boldsymbol{r},\gamma_0,c) = \frac{f(z_{ji},\boldsymbol{z}^{-ji},\boldsymbol{m}|\boldsymbol{r},\gamma_0,c)}{\sum_{k=1}^{K_J^{-ji}+1} f(z_{ji}=k,\boldsymbol{z}^{-ji},\boldsymbol{m}|\boldsymbol{r},\gamma_0,c)}.$$

$$\propto \begin{cases} \dfrac{n_{\cdot k}^{-ji}}{c+n_{\cdot k}^{-ji}+r_{\cdot}}(n_{jk}^{-ji}+r_j), & \text{for } k=1,\cdots,K_J^{-ji}; \\ \dfrac{\gamma_0}{c+r_{\cdot}}r_j, & \text{if } k=K_J^{-ji}+1. \end{cases}$$

Random count matrices with fixed row sums

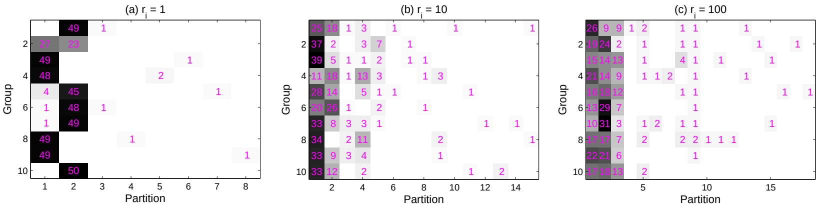


Figure: Random draws from the EPPF that governs the BNBP's exchangeable random partitions of 10 groups (rows), each of which has 50 data points.
The jth row of each matrix, which sums to 50, represents the partition of the $m_j = 50$ data points of the jth group over a random number of exchangeable clusters.
The kth column of each matrix represents the kth nonempty cluster in order of appearance in Gibbs sampling (the empty clusters are deleted).

The GNBP's EPPF for mixed-membership modeling

- GNBP random count matrix prior

$$f(\mathbf{N}_J, \mathbf{L}_J \mid \gamma_0, c, \boldsymbol{p}) = \frac{\gamma_0^{K_J} \exp\left[-\gamma_0 \ln\left(\frac{c+q_\cdot}{c}\right)\right]}{K_J!} \prod_{k=1}^{K_J} \frac{\Gamma(l_{\cdot k})}{(c+q_\cdot)^{l_{\cdot k}}} \left(\prod_{j=1}^{J} \frac{|s(n_{jk}, l_{jk})| p_j^{n_{jk}}}{n_{jk}!} \right)$$

- With $\boldsymbol{z} = (z_{11}, \ldots, z_{Jm_J})$, $\boldsymbol{b} = (b_{11}, \ldots, b_{Jm_J})$, and $n_{jkt} = \sum_{i=1}^{m_j} \delta(z_{ji} = k, b_{ji} = t)$, the joint distribution of a column count vector $\boldsymbol{m} = (m_1, \ldots, m_J)^T$, its partition into a column exchangeable latent random count matrix with K_J nonempty columns, and an auxiliary categorical random vector can be expressed as

$$f(\boldsymbol{b}, \boldsymbol{z}, \boldsymbol{m} | \gamma_0, c, \boldsymbol{p}) = \gamma_0^{K_J} e^{-\gamma_0 \ln\left(\frac{c+q_\cdot}{c}\right)} \times \left(\prod_{j=1}^{J} \frac{p_j^{m_j}}{m_j!} \right) \prod_{k=1}^{K_J} \left[\frac{\Gamma(l_{\cdot k})}{(c+q_\cdot)^{l_{\cdot k}}} \prod_{j=1}^{J} \prod_{t=1}^{l_{jk}} \Gamma(n_{jkt}) \right]$$

- The GNBP's EPPF for mixed-membership modeling:

$$f(\boldsymbol{z}, \boldsymbol{b} | \boldsymbol{m}, \gamma_0, c, \boldsymbol{p}) = \frac{f(\boldsymbol{z}, \boldsymbol{b}, \boldsymbol{m} | \gamma_0, c, \boldsymbol{p})}{f(\boldsymbol{m} | \gamma_0, c, \boldsymbol{p})}$$

- The prediction rule is simple:

$$\begin{aligned} & P(z_{ji} = k, b_{ji} = t | \boldsymbol{b}^{-ji}, \boldsymbol{z}^{-ji}, \boldsymbol{m}, \boldsymbol{p}, c) \\ & = \frac{f(z_{ji} = k, b_{ji} = t, \boldsymbol{b}^{-ji}, \boldsymbol{z}^{-ji}, \boldsymbol{m} | \boldsymbol{p}, c)}{\sum_{z_{ji}, b_{ji}} f(z_{ji}, b_{ji}, \boldsymbol{b}^{-ji}, \boldsymbol{z}^{-ji}, \boldsymbol{m} | \boldsymbol{p}, c)} \\ & \propto \begin{cases} n_{jkt}^{-ji}, & \text{if } k \leq K_J^{-ji}, t \leq l_{jk}^{-ji}; \\ l_{\cdot k}^{-ji}/(c + q_\cdot), & \text{if } k \leq K_J^{-ji}, t = l_{jk}^{-ji} + 1; \\ \gamma_0/(c + q_\cdot), & \text{if } k = K_J^{-ji} + 1, t = 1. \end{cases} \end{aligned}$$

- If we let z_{ji} be the dish index and b_{ji} be the table index for customer i in restaurant j, then the collapsed Gibbs sampler can be related to the Chinese restaurant franchise sampler of the hierarchical Dirichlet process (Teh et al., 2005).

Conclusions

- A family of probability mass functions for random count matrices.
- The proposed random count matrices have a random number of i.i.d. columns and could also be constructed by adding one row at a time.
- Their parameters can be inferred with closed-form Gibbs sampling update equations.
- Infinite vocabulary naive Bayes classifiers.
- Priors for random count matrices can be used to construct (group size dependent) EPPFs for mixed-membership modeling, with simple prediction rules for collapsed Gibbs sampling.

Main References

M. Zhou, O. H. M. Padilla and J. G. Scott.
Priors for random count matrices derived from a family of negative binomial processes.
arXiv:1404.3331, 2014.

M. Zhou.
Beta-negative binomial process and exchangeable random partitions for mixed-membership modeling.
NIPS, 2014.

M. Zhou and S. G. Walker.
Sample size dependent species models.
arXiv:1410.3155, 2014.

C. Heaukulani and D. M. Roy.
The combinatorial structure of beta negative binomial processes.
arXiv:1401.0062, 2013.

T. Broderick, L. Mackey, J. Paisley, and M. I. Jordan.
Combinatorial clustering and the beta negative binomial process.
IEEE Trans. Pattern Analysis and Machine Intelligence, 2015.

M. Zhou and L. Carin.
Negative binomial process count and mixture modeling.
IEEE Trans. Pattern Analysis and Machine Intelligence, 2015.

M. Zhou and L. Carin.
Augment-and-conquer negative binomial processes.
In *NIPS*, 2012.

M. Zhou, L. Hannah, D. Dunson, and L. Carin.
Beta-negative binomial process and Poisson factor analysis.
In *AISTATS*, 2012.