Streaming Inference for Infinite Feature Models

Rylan Schaeffer [1 2] **Yilun Du** [3] **Gabrielle Kaili-May Liu** [2] **Ila Rani Fiete** [2 4]

Abstract

Unsupervised learning from a continuous stream of data is arguably one of the most common and most challenging problems facing intelligent agents. One class of unsupervised models, collectively termed *feature models*, attempts unsupervised discovery of latent features underlying the data and includes common models such as PCA, ICA, and NMF. However, if the data arrives in a continuous stream, determining the number of features is a significant challenge, and the number may grow with time. In this work, we make feature models significantly more applicable to streaming data by imbuing them with the ability to create new features, online, in a probabilistic and principled manner. To achieve this, we derive a novel recursive form of the Indian Buffet Process, which we term the *Recursive IBP* (R-IBP). We demonstrate that R-IBP can be be used as a prior for feature models to efficiently infer a posterior over an unbounded number of latent features, with quasilinear average time complexity and logarithmic average space complexity. We compare R-IBP to existing sampling and variational baselines in two feature models (Linear Gaussian and Factor Analysis) and demonstrate on synthetic and real data that R-IBP achieves comparable or better performance in significantly less time.

1. Introduction

Feature models are a broad class of unsupervised probabilistic models that aim to decompose data into an unknown number of unknown features under certain assumptions, a class which includes principal component analysis, factor analysis, independent component analysis, non-negative matrix factorization, matching pursuit, and more.

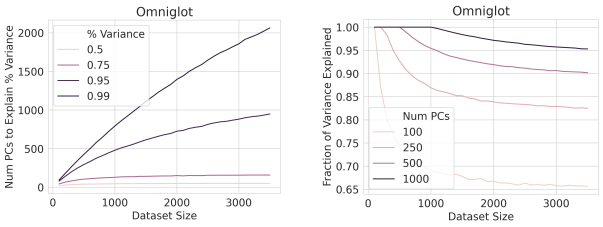

Figure 1. **Motivation for Infinite Feature Models.** As more data are observed, a feature model (here: PCA) requires increasingly more features to explain the data (Omniglot handwritten characters) (left) or else becomes increasingly unable to do so (right).

A fundamental problem in feature modeling – analogous to the problem in mixture modeling of choosing the number of clusters – is choosing the number of features. Users typically employ one of two approaches: Either (1) prespecifying a fixed number of features or (2) retroactively choosing a number of features after seeing all the data based on some criterion (e.g., selecting the number of principal components necessary to explain 95% of the variance). In a streaming setting, however, where data are received over time, neither approach suffices. For instance, representing handwritten characters with a fixed number of principal components becomes inadequate as more characters are encountered (Fig. 1). Thus the number of features should flexibly adapt to the data in the streaming context.

Such flexibility is a goal not only because the streaming setting for feature models is important in its own right, but also because feature models are a pervasive approach taken in neuroscience and cognitive science to explain how intelligent agents model the world as they move through it (Olshausen & Field, 1997; Hyvärinen, 2010; Pehlevan et al., 2015). Intelligent agents, from mice to humans to mobile devices, must deal with streaming data since these agents operate with limited memory that renders storage of and computation on all previously seen data prohibitive.

This raises the question of how to perform efficient streaming inference for "infinite" feature models, a question we answer here. Following the approach of Schaeffer et al. (2021) to efficient streaming inference for "infinite" mixture models, we first show that the Indian Buffet Process

[*]Equal contribution [1]Computer Science, Stanford University [2]Brain and Cognitive Sciences, MIT [3]Electrical Engineering and Computer Science, MIT [4]McGovern Institute for Brain Research, MIT. Correspondence to: Rylan Schaeffer <rschaef@cs.stanford.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

(Griffiths & Ghahramani, 2005), a stochastic process frequently used for Bayesian nonparametric feature models, can be rewritten in a novel form designed for streaming inference with expected quasilinear time and expected logarithmic space complexity. We then demonstrate on both synthetic and real (tabular & non-tabular) data that R-IBP matches or exceeds the performance of five streaming and non-streaming baseline inference algorithms in less time.

2. Background

2.1. Generative Model

We consider observing a sequence of D-dimensional variables $o_{1:N}$ ($o_n \in \mathbb{R}^D$) based on a sequence of N K-dimensional binary latent variables $z_{1:N}$, with $z_n \in \{0,1\}^K$, K unknown, and $\cdot_{1:N}$ denoting the sequence $(\cdot_1, \cdot_2, ..., \cdot_N)$. Each z_{nk} in the $(N \times K)$-dimensional latent variable matrix Z denotes the presence or absence of the kth feature in the nth observation. Each feature is some unknown vector $A_k \in \mathbb{R}^D$ drawn i.i.d. from some distribution $p(A_\cdot)$. Because the number of latent features K is unknown, the Indian Buffet Process (IBP) serves as a flexible prior over the latent indicators:

$$\begin{aligned} z_{1:N} &\sim IBP(\alpha, \beta) \\ A_k &\sim_{i.i.d} p(A_\cdot) \\ o_n | z_n, \{A_k\} &\sim p(o | z_n, \{A_k\}) \end{aligned} \tag{1}$$

This encompasses many feature models including Principal Component Analysis, Factor Analysis, Independent Component Analysis, and Non-Negative Matrix Factorization.

2.2. Indian Buffet Process

The Indian Buffet Process (IBP) (Griffiths & Ghahramani, 2011) is a two-parameter[1] ($\alpha > 0, \beta > 0$) stochastic process that defines a discrete distribution over binary matrices with finitely many rows (observations) and an unbounded number of columns (features). The name IBP arises from imagining customers (rows/observations) arriving sequentially at a buffet that has an infinite number of dishes (columns/features) and selecting which dishes to eat: the nth customer selects an integer number of new dishes $\lambda_n \sim Poisson(\alpha\beta/(\beta + n - 1))$ and then selects previous dishes with probability proportional to the number of previous customers who selected those dishes. Denoting the total number of dishes after the first n customers $\Lambda_n = \sum_{n'=1}^{n'=n} \lambda_{n'}$, the IBP defines a conditional distribution for the nth row and kth column's binary variable z_{nk}:

$$p(z_{n,k} = 1 | z_{<n,k}, \Lambda_{n-1}, \lambda_n, \alpha, \beta) = \begin{cases} \frac{1}{\beta+n-1} \sum_{n'<n} z_{n'k} & \text{if } 1 \le k \le \Lambda_{n-1} \\ 1 & \text{if } \Lambda_{n-1} < k \le \Lambda_{n-1} + \lambda_n \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The IBP is a useful stochastic process for defining a prior over the number of features as well as the presence/absence of features in any particular observation because it allows for the number of features to grow as more data are observed while independently controlling the features' sparsity. Because each λ_n is an independent Poisson with rate $\alpha\beta/(\beta + n - 1)$ and because the sum of independent Poisson random variables is itself Poisson, we know that $\Lambda_n \sim Poisson(\sum_{n'=1}^{n} \alpha\beta/(\beta + n' - 1))$. This implies the expected number of dishes grows logarithmically with n because $\mathbb{E}[\Lambda_n] = \sum_{n'=1}^{n} \alpha\beta/(\beta + n' - 1)) \approx \alpha\beta \int_{n'=1}^{n} dn'/(\beta+n'-1) = \alpha\beta(\log(\beta+n-1) - \log(\beta)) \approx \alpha\beta \log(1 + n/\beta)$; this detail becomes important in our later complexity analysis. Ahead, we often omit α, β for brevity.

3. Efficient Streaming Inference

3.1. Objective

Our goal is to infer a posterior distribution over the current observation's binary latent variables $z_n \overset{\text{def}}{=} \{z_{nk}\}_{k=1}^{k=\infty}$ and the latent features $\{A_k\}_{k=1}^{\infty}$, given the entire history of observations $o_{\le n}$, subject to two constraints:

1. Inference must be performed online, i.e. the nth observation is discarded before proceeding to the $(n+1)$th observation.

2. Inference must be efficient in the large sample limit.

Inferring the latent posterior $p(z_n, \{A_k\} | o_{\le n})$ is often called filtering (e.g., Kalman filter, particle filter). We slightly abuse terminology by calling $p(z_n, \{A_k\} | o_{<n})$ the *filtering prior* and $p(z_n, \{A_k\} | o_{\le n})$ the *filtering posterior*, to indicate whether the observation o_n is conditioned upon.

3.2. Challenges with Streaming Inference

Filtering with an IBP prior requires solving several emergent problems. For concreteness, we illustrate these problems on the commonly used linear-Gaussian model (Griffiths & Ghahramani, 2005; Teh et al., 2007; Doshi-Velez et al., 2009; Paisley & Carin, 2009; Doshi-Velez & Ghahramani, 2009), although our experiments will also showcase Factor Analysis. In the linear-Gaussian model, $O \in \mathbb{R}^{N \times D}$ are the observed data, $Z \in \{0,1\}^{N \times K}$ are the binary indicators,

[1] The IBP originally had a single parameter (Griffiths & Ghahramani, 2005) but was extended to two (Ghahramani et al., 2007) and later three (Teh & Görür, 2009). Our paper applies equally to all, but since our focus is on efficient streaming inference and not particular properties of an IBP variant, we chose the two parameter IBP to balance expositional simplicity against model flexibility.

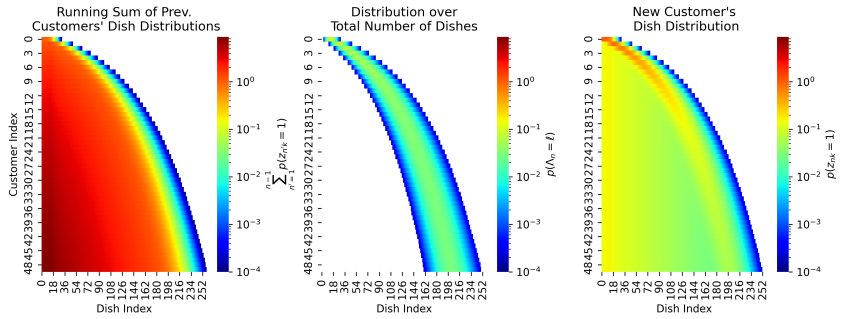

Figure 2. **Visualization of the Recursive IBP**. To make streaming inference possible, we break the IBP's dependence on the entire history $z_{<n}$ by converting the conditional $p(z_n|z_{<n}, \alpha, \beta)$ into a sequence of marginals $p(z_n|\alpha, \beta)$. The running sum of the previous marginal distributions $\sum_{n'<n} p(z_{n'k} = 1)$ (left) and the distribution over the number of dishes $p(\Lambda_n = k)$ (middle) together determine the next marginal distribution $p(z_{nk} = 1)$ (right).

$A \in \mathbb{R}^{K\times D}$ are the features, and $E \in \mathbb{R}^{N\times D}$ are noise.

$$\begin{aligned} Z &\sim IBP(\alpha, \beta) \\ A_k &\sim \mathcal{N}(\mu_A, \Sigma_A) \\ E_n &\sim \mathcal{N}(0, \sigma_o^2 I_{D\times D}) \\ O &= ZA + E \end{aligned} \tag{3}$$

On streaming data, each o_n (i.e. row of O) is observed, then discarded. What are the challenges for inference?

1. **Dependence on Entire History**: The IBP's conditional distribution $p(z_{nk}|z_{<nk}, \Lambda_{n-1}, \lambda_n)$ renders the current indicators z_n dependent on *all* previous indicators $z_{<n}$, implying any inference algorithm must remember the entire history of indicators.

2. **Exponentially Many Evaluations of Likelihood**: The likelihood $p(o_n|z_n; A)$ looks benign, but recall that z_n is the set of binary variables $\{z_{nk}\}_{k=1}^{k=\Lambda_n}$. This means computing a posterior requires evaluating the likelihood for 2^{Λ_n} possible configurations at each step n.

3. **History Dependence and Non-Factorized Posterior**: In the prior, the indicators are independent, i.e., $p(z_n|z_{<n}, \Lambda_{n-1}, \lambda_n) = \prod_{k=1}^{k=\Lambda_n} p(z_{nk}|z_{<nk}, \Lambda_{n-1}, \lambda_n)$, although this independence depends on knowing the entire history $z_{<n}$. Upon conditioning on the observations, the indicators are no longer independent, i.e., $p(z_n|o_{\leq n}) \neq \prod_{k=1}^{k=\Lambda_n} p(z_{nk}|o_{\leq n})$, because features are not required to be orthogonal and the presence/absence of one feature can "explain away" the presence/absence of another feature.

4. **Unknown Posterior over Number of Features**: In the IBP prior, the number of new indicators per observation λ_n and the total number of indicators after n observations $\Lambda_n = \sum_{n'=1}^{n} \lambda_{n'}$ are both Poisson with known rates. But what are the posterior distributions, and are they efficiently computable?

3.3. Recursive Expression for IBP Marginals

Our approach is to recast the IBP in a novel form that breaks the IBP conditional distribution's dependence on the entire history. We achieve this by converting the conditional distribution $p(z_{nk} = 1|z_{<n}, \Lambda_{n-1}, \lambda_n, \alpha, \beta)$, which depends on the entire history, into a sequence of marginal distributions $p(z_{nk} = 1|\alpha, \beta)$ that can be efficiently computed recursively. The marginal distribution $p(z_{nk} = 1)$ is exactly equal to the IBP's conditional distribution averaged over all sample paths:

$$\begin{aligned} p(z_{nk} = 1) &= \mathbb{E}_{p(z_{nk})}[\mathbb{I}(z_{nk} = 1)] \\ &= \mathbb{E}_{p(z_{<n}, \Lambda_{n-1}, \lambda_n)}\Big[\mathbb{E}_{p(z_{nk}|z_{<n}, \Lambda_{n-1}, \lambda_n)}[\mathbb{I}(z_{nk} = 1)]\Big] \\ &= \mathbb{E}_{p(z_{<n}, \Lambda_{n-1}, \lambda_n)}\Big[p(z_{nk}|z_{<n}, \Lambda_{n-1}, \lambda_n)\Big] \end{aligned}$$

Substituting Eqn. (2) and simplifying yields a recursive expression for the marginal distribution:

$$\begin{aligned} p(z_{nk} = 1) = \frac{1}{\beta + n - 1} \sum_{n'<n} p(z_{n'k} = 1) \\ + p(\Lambda_{n-1} \leq k - 1) - p(\Lambda_{n-1} + \lambda_n \leq k - 1) \end{aligned} \tag{4}$$

We term Eqn. (4) the recursive form of the IBP, or the **Recursive IBP** for short. Intuitively, the Recursive IBP tells us that the probability that the kth feature is present in the nth observation is given by the running sum of how probable the kth feature's presence was in each previous observation, plus the difference of two terms; this difference is between two Poisson CDFs, which drives new observations to add new features. Fig. 2 offers a visual intuition for Eqn. (4), showing how the accumulating probability mass of previous dishes competes with the addition of new dishes to determine the next customer's likely dishes. This recursive form

of the IBP preserves two qualities of the IBP: (1) if a feature is frequently present in observations, then the next observation is also likely to possess that feature, and (2) new features can be created to explain new data.

3.4. Performing Inference with the Recursive IBP

For the IBP as a stochastic process, Eqn. (4) is exact. However, to use the R-IBP for inference, we use one approximation. To see why, suppose we want a prior for the next observation and so condition on the sequence of observations up to but excluding the current index:

$$p(z_{nk}=1|o_{<n}) = \frac{1}{\beta+n-1}\sum_{n'<n} p(z_{n'k}=1|o_{<n}) + p(\Lambda_{n-1}\leq k-1|o_{<n}) - p(\Lambda_{n-1}+\lambda_n \leq k-1|o_{<n})$$

Each term $p(z_{n'k}=1|o_{<n})$ in the sum requires that, for each observation, all previous posteriors must be retroactively revised; these revisions would require $O(n)$ operations at each step n, and would also require remembering all n observations. To avoid this, we turn to approximate inference by approximating the true IBP prior $p(z_n|o_{<n})$ with an approximate IBP-like prior $q(z_n|o_{<n})$:

$$q(z_{nk}|o_{<n}) \stackrel{\text{def}}{=} \frac{1}{\beta+n-1}\sum_{n'<n} q(z_{n'k}=1|o_{\leq n'}) + q(\Lambda_{n-1}\leq k-1|o_{<n}) - q(\Lambda_{n-1}+\lambda_n \leq k-1|o_{<n}) \tag{5}$$

This approximate prior is akin to the true prior, with the key difference that the former prohibits revising previous posteriors based on later observations. For the linear-Gaussian model, the variational family in which we optimize is:

$$q(z_n|o_{\leq n};\theta_n) \stackrel{\text{def}}{=} \prod_k q(z_{nk}|o_{\leq n};b_{nk})q(A_k|o_{\leq n};\mu_{nk},\Sigma_{nk})$$
$$q(z_{nk}|o_{\leq n};b_{nk}) \stackrel{\text{def}}{=} \text{Bern}(b_{nk})$$
$$q(A_k|o_{\leq n};\mu_{nk},\Sigma_{nk}) \stackrel{\text{def}}{=} \mathcal{N}(\mu_{nk},\Sigma_{nk})$$

where $\theta_n \stackrel{\text{def}}{=} \{b_{nk}\}_k \cup \{\mu_{nk}\}_k \cup \{\Sigma_{nk}\}_k$ are the variational parameters and the optimization problem is to maximize the approximate lower bound:

$$\begin{aligned}\mathcal{L}(\theta_n) \stackrel{\text{def}}{=} &\mathbb{E}_{q(z_n,A|o_{\leq n})}[\log p(o_n|z_n,A)] \\ &+ \mathbb{E}_{q(A|o_{\leq n})}[\log q(A|o_{<n})] \\ &+ \mathbb{E}_{q(z_n|o_{\leq n})}[\log q(z_n|o_{<n}) \\ &+ H[q(z_n,A|o_{\leq n})]\end{aligned} \tag{6}$$

The variational parameters must be solved self-consistently, and we derive the necessary equations in closed form in the Supplement. At the risk of overloading terminology, we also call this inference algorithm *R-IBP* based on its origin. R-IBP operates by performing a single iteration of message passing on the IBP's directed graph. Two advantages of our approximation are that it solves the second challenge (exponentially many likelihood evaluations) and the third (non-factorized posterior), but at the cost of using an objective function that is no longer a guaranteed lower bound on the log evidence. We do not necessarily see this as a problem since prior work shows that tighter log evidence bounds do not necessarily produce better models (Rainforth et al., 2019). Yet the fourth issue remains: what is the filtered posterior $q(\Lambda_n|o_{\leq n})$ over the total number of features?

3.5. Distribution over Number of Features

Perhaps surprisingly, under the same assumption and regardless of the particular feature model, the filtered posterior over the number of features is Poisson with a calculable rate. The total number of dishes after the nth customer is defined:

$$\Lambda_n \stackrel{\text{def}}{=} \sum_{k=1}^{k=\infty} \min\left(1, \sum_{n'=1}^{n'=n} z_{n'k}\right)$$

Each term in the sum counts whether the kth feature was present in at least one of the first n observations, and the sum is therefore a Bernoulli random variable with success probability $1-\prod_{n'=1}^{n'=n} p(z_{n'k}=0|o_{\leq n'})$ because, in order for the k-th feature to not exist, the feature cannot have been present in any of the first n observations. Le Cam's Theorem (Le Cam, 1960) tells us that the sum of independent Bernoullis is closely approximated by a Poisson:

$$q(\Lambda_n|o_{\leq n}) \approx \text{Poisson}\left(\sum_{k=1}^{k=\infty}\left(1-\prod_{n'=1}^{n'=n} q(z_{n'k}=0|o_{\leq n'})\right)\right)$$

Additionally, because the prior over the number of new features added by the nth observation is independent from the preceding total number of features, the second Poisson in Eqn. (5) is distributed:

$$q(\Lambda_n|o_{<n}) \approx \text{Poisson}\left(\frac{\alpha\beta}{\beta+n-1} + \sum_{k=1}^{k=\infty}\left(1-\prod_{n'=1}^{n'=n-1} q(z_{n'k}=0|o_{\leq n'})\right)\right)$$

Although the equation looks daunting, all terms are available through R-IBP. For detailed derivation, see the Supplement.

3.6. Complexity Analysis

The time and space complexity of the Recursive IBP is determined by the number of latent features Λ_n, which is

unbounded and neither converges nor concentrates. If we instead use the expected number of latent features in the prior $\mathbb{E}[\Lambda_n] = \alpha\beta\log(1+\beta/n)$, and assume that we take at most S coordinate ascent steps per observation, the average-case time complexity per observation is $O(S\mathbb{E}[\Lambda_n])$, making the total average-case time complexity for N observations quasi-linear $O(NS\log(1+N))$. The total average-case space complexity is logarithmic $O(\mathbb{E}[\Lambda_N]) = O(\log(1+N))$ to store the variational parameters, the running sum of probability masses and the running product of probability masses. Empirically, we find that R-IBP follows these asymptotic trends on both synthetic and real data (Fig. 8).

4. Analytical Results

4.1. R-IBP in Zero-Noise Limit

In general, given some generative model and an inference algorithm, one often wants to know whether the algorithm converges, what it converges to, and how quickly it converges. Feature models are notoriously difficult to characterize analytically for several reasons including degeneracies and combinatoric complications. To our knowledge, there is one setting in which IBP theory was achieved: the linear-Gaussian model in the zero noise limit, i.e. $\sigma_o^2 \to 0$ (Broderick et al., 2013b). By considering R-IBP in the same limit, and similarly reparameterizing the model with $\alpha \overset{\text{def}}{=} \exp(-\gamma^2/2\sigma_o^2)$ and setting $\beta = 1$, we show:

Proposition 4.1. *Consider a linear-Gaussian model $O = ZA + E$ with an $IBP(\alpha, \beta)$ prior on Z. In the limit $\sigma_o^2 \to 0$, R-IBP fits the data using Z and A with a regularization term penalizing the number of features Λ_N by minimizing the objective function:*

$$\text{Tr}\left[(O - ZA)^T(O - ZA)\right] + \gamma^2\Lambda_N \tag{7}$$

This objective function tells us R-IBP will seek to minimize the squared error between the observations and the subset of infinite features thought to be present (or equivalently, maximize the log likelihood of the data), with a regularization term that penalizes the number of used features. This objective function is akin to the Bayesian Information Criterion (Schwarz, 1978), in that it maximizes the log likelihood while penalizing the number of parameters. However, R-IBP does not necessarily converge because it performs only a single pass through the data and multiple passes may be necessary for convergence. Our proof works by showing that in the $\sigma_o^2 \to 0$ limit, R-IBP becomes Broderick et al.'s BP-Means algorithm (Broderick et al., 2013b) and thus minimizes the same objective; see the Supplement for details.

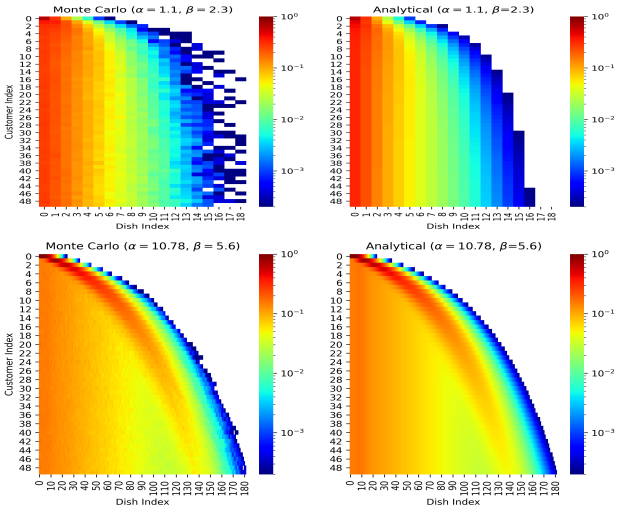

Figure 3. **Monte Carlo vs. Analytical Expression.** Over a wide range of (α, β) parameter pairs, we find excellent visual match between Monte Carlo estimates of the marginal probabilities drawn from the conditional $p(z_n|z_{<n}, \alpha, \beta)$ (left) and the Recursive IBP's marginal probabilities $p(z_n|\alpha, \beta)$ (right).

5. Experimental Results

5.1. Exactness of Recursive IBP for the IBP

Setting inference aside temporarily and considering solely the IBP stochastic process, the Recursive IBP should exactly give the IBP indicators' marginal distributions. We confirm this by comparing Eqn. (4)'s analytical expression to 5000 Monte Carlo samples drawn from the IBP's conditional distribution over $\alpha \in \{1.1, 10.78, 15.37\} \times \beta \in \{2.3, 5.6, 12.9\}$. Visually, the analytical and Monte Carlo plots display excellent agreement (Fig. 3). Quantitatively, the mean squared error between the analytical expression for all $p(z_{nk}|\alpha, \beta)$ and the Monte Carlo estimates falls approximately as a power law in the number of Monte Carlo samples (Fig. 4) for all (α, β) values. This supports our claim that R-IBP is exact for the IBP as a stochastic process.

5.2. Infinite Linear Gaussian on Synthetic Data

We next turned to performing inference in the linear-Gaussian (LG) feature model given in Eqn. (3):

$$O = ZA + E$$

where the indicators Z are drawn from an IBP and the features $A = \{A_k\}$ from a matrix Normal distribution. We used synthetically generated data to have access to ground truth features. We compared R-IBP against five baseline algorithms. The first two baselines are streaming algorithms, whereas the last three baselines are non-streaming algorithms that have unfettered access to all observations and

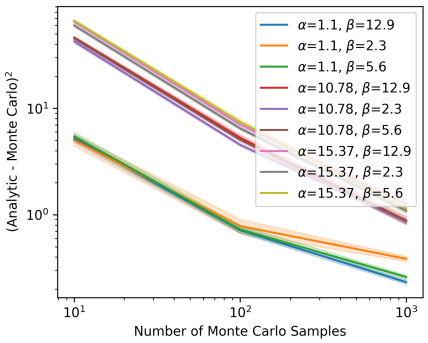

Figure 4. **Mean-Squared Error between analytical expression for the marginal and a Monte Carlo marginal estimate.** Over a wide range of (α, β) pairs, the mean-squared error between our analytical expression and Monte Carlo estimates falls approximately as a power law, showing the exactness of Eqn. (4).

therefore serve as upper bounds on performance; any comparison against these last three baselines maximally disfavors our method. The baseline algorithms are:

- Streaming Variational Inference (Widjaja & Doshi-Velez, 2017), both "finite" and "infinite" variants.
- Variational Inference (Doshi-Velez et al., 2009), both "finite" and "infinite" variants.
- Hamiltonian Monte Carlo-Gibbs Sampling (Duane et al., 1987), implemented in Pyro (Bingham et al., 2019)

We included Widjaja & Doshi-Velez (2017)'s method despite being less well known because it is the only streaming variational inference algorithm for the IBP that we are aware of. At a high level, the algorithm works via a Beta Process stick-breaking construction. Specifically, each presence/absence indicator z_{nk} for the kth feature is sampled i.i.d. from $Bernoulli(\pi_k)$, where π_k is defined as a product of i.i.d. Beta variables $\pi_k \stackrel{\text{def}}{=} \prod_{k' \leq k} v_{k'}$ and $v_{k'} \sim_{i.i.d.} Beta(\alpha, 1)$; the variational distribution for each $v_{k'}$ is then defined as $Beta(\tau_{k'1}, \tau_{k'2})$ for variational parameters $\tau_{k'1}, \tau_{k'2}$.

Quantitatively comparing inference algorithms for feature models is notoriously difficult, and many papers skip attempting to do so altogether, e.g., (Griffiths & Ghahramani, 2005; Teh et al., 2007; Miller et al., 2009; Paisley & Carin, 2009; Paisley et al., 2012; 2010). The most appropriate metric we found was the (negative log) posterior predictive probability (Widjaja & Doshi-Velez, 2017; Paisley et al., 2011), as the metric may be be computed for any inference algorithm, regardless of underlying parametric assumptions. The posterior predictive distribution quantifies, in a parameter-free manner, how probable new observations O_{test} drawn from the same generative process are, after seeing the original data O_{train}, marginalizing over all possible parameters:

$$
\begin{aligned}
&p(O_{test}|O_{train}) \\
&= \int p(O_{test}|Z_{test}, A) p(Z_{test}, A|O_{train}) d(Z_{test}, A) \\
&= \mathbb{E}_{p(Z_{test}, A|O_{train})}[p(O_{test}|Z_{test}, A)] \\
&\approx \frac{1}{S} \sum_{s=1}^{s=S} \mathcal{N}(O_{test}|Z_{test}^{(s)}, A^{(s)}, \sigma_o^2)
\end{aligned}
$$

where S is a pre-specified number of samples (we arbitrarily use 100) and $Z_{test}^{(s)}, A^{(s)} \sim p(Z_{test}, A|O_{train})$.

Over different (α, β) pairs and averaging over 10 synthetically generated datasets, we find that R-IBP achieves lower (better) negative log posterior predictive values than all other inference algorithms except for Doshi-Velez's (non-streaming) finite algorithm (Doshi-Velez et al., 2009) (Fig. 5), outperforming even Doshi-Velez's (non-streaming) infinite algorithm. We also find that R-IBP is significantly faster than almost all other inference algorithms (Fig. 5) except for Widjaja's (streaming) finite algorithm (Widjaja & Doshi-Velez, 2017) which achieves significantly worse performance. These results demonstrate that R-IBP provides a good tradeoff between performance and speed and is a competitive inference algorithm for infinite feature modeling on streaming and on non-streaming data.

One surprise was that R-IBP sometimes performs as well as, or even better than, non-streaming baselines when the model is properly specified. For both of Widjaja et al.'s algorithms and both of Doshi-Velez et al.'s algorithms, we used author-published code to minimize the possibility of implementation errors. Our hypothesis (see Discussion for details and supporting evidence) is that because most (if not all) IBP-inference algorithms rely on stick-breaking constructions that *chain-multiply* inferred beta variables, errors amplify in a multiplicative way, whereas R-IBP *adds* inferred beta variables to cumulative sums of sufficient statistics, washing out errors as more data are observed.

We also tested whether R-IBP recovers the true number of features when the model is properly specified. We found that as R-IBP receives more observations, it converges to the true number of inferred features (Fig. 6, left), over a range of different data dimensions. Those features are incrementally added with more observations (Fig. 6, right).

5.3. Infinite Linear Gaussian on MNIST Digits

We next tested how well R-IBP performs on real data, following the example set by (Paisley & Carin, 2009): we took the odd digits from MNIST (Lecun et al., 1998) and measured how (dis)similar the features inferred for each class

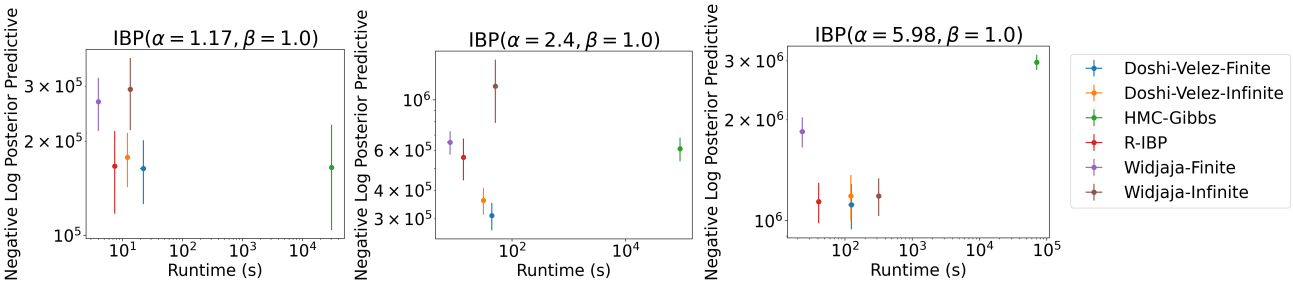

Figure 5. **Comparison of Linear-Gaussian Inference Algorithms.** Over a range of α values, R-IBP is significantly faster than baseline inference algorithms and has better (lower) negative log posterior predictive values than the streaming baselines and even some non-streaming baselines, averaged over 10 synthetic datasets. We fix $\beta = 1.0$ because baseline algorithms are only defined for $\beta = 1.0$. In all panels, the correct α, β values are given to each inference algorithm.

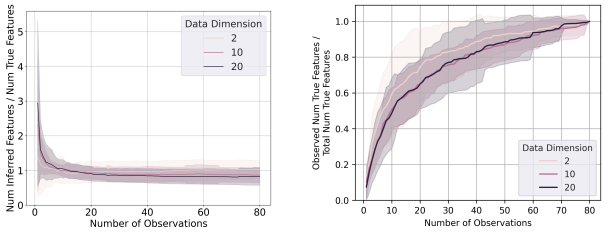

Figure 6. **R-IBP Feature Recovery on Streaming Data.** R-IBP recovers the correct order of magnitude of number of features (left), adding features as more observations are encountered (right).

are. To quantify similarity between MNIST digit classes, we compute the fraction of features shared between two data drawn from the same digit vs. two data drawn from different digits. One might predict that 3 and 5 are similar, 7 and 9 are similar, and perhaps 1 is on its own. This is precisely what R-IBP recovers, in an unsupervised manner, qualitatively matching the confusion matrix of a separately trained supervised convolutional neural network classifier (Fig. 7), and matching the results of Paisley & Carin (2009).

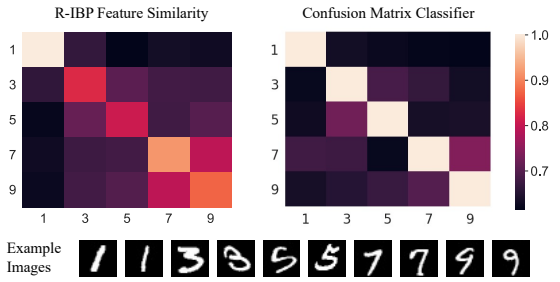

Figure 7. **R-IBP Recovers Intuitive Features for MNIST Classes.** Feature similarity between images of MNIST digits drawn from same and different classes. Feature similarity matches the confusion matrix of an independently-trained convolutional neural network classifier on MNIST images. R-IBP infers more similar features for digit classes 3 and 5, and for 7 and 9, with the digit class 1 largely isolated.

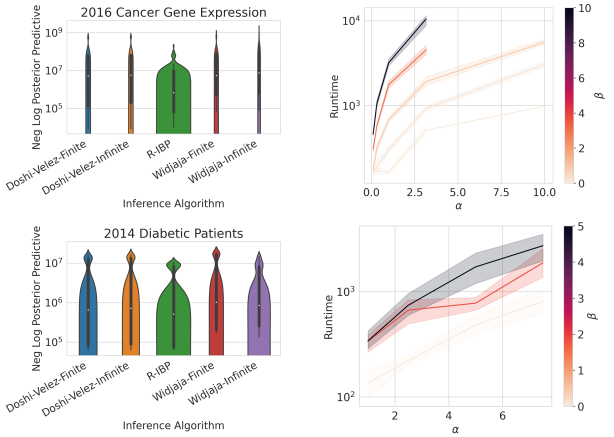

Figure 8. **R-IBP performance on cancer gene expression and diabetic patient data.** On cancer gene expression (top) and diabetic patient (bottom) data, R-IBP matches or outperforms baseline algorithms across hyperparameter configurations (left). R-IBP runtime scales linearly with α and quasilinearly with β (right), qualitatively matching our complexity analysis.

5.4. Infinite Linear Gaussian on Tabular Data

We additionally tested R-IBP on tabular data, using two datasets from the UCI Machine Learning Repository (Dua & Graff, 2017): gene expression of cancer patients (801 samples, 20k features), and diabetic patient profiles (100k samples, 55 features) (Strack et al., 2014). Because the hyperparameters $\alpha, \beta, \sigma_A, \sigma_o$ are unknown, we swept these for each algorithm. The distribution of negative log posterior predictive scores shows that on both datasets, R-IBP performs well (Fig. 8, left); however, if one considers only the best configuration of hyperparameters for each algorithm, the two Doshi-Velez algorithms outperform R-IBP. We also tested whether our complexity analysis holds qualitatively by plotting how R-IBP's runtime varies as a function of α, β, with the expectation that the runtime should scale linearly with α and quasilinearly with β i.e. $\beta \log(1 + N/\beta)$; this is precisely what we found in both datasets (Fig. 8, right).

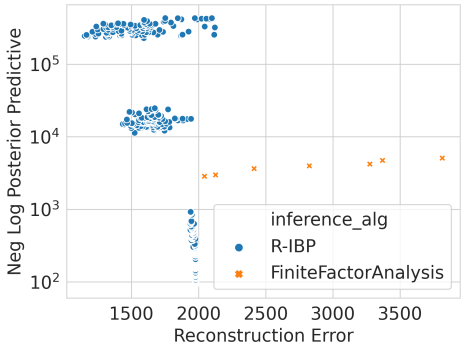

Figure 9. **R-IBP vs Finite Factor Analysis on Omniglot.** R-IBP overfits for low noise parameters (upper left), but outperforms Finite Factor Analysis for higher noise parameters (lower center).

5.5. Infinite Factor Analysis on Omniglot

We conclude by demonstrating R-IBP's general applicability using a different feature model, Factor Analysis (FA), which generalizes linear-Gaussian and (probabilistic) PCA. FA introduces $W_n \in \mathbb{R}^K \sim \mathcal{N}(0, \Sigma_w)$ (with Σ_w diagonal) to capture the degree to which a feature is expressed. The FA generative model is:

$$\begin{aligned} Z &\sim IBP(\alpha, \beta) \\ A_k &\sim \mathcal{N}(\mu_A, \Sigma_A) \\ w_n &\sim \mathcal{N}(0, \Sigma_w) \\ e_n &\sim \mathcal{N}(0, \sigma_o^2 I_{D\times D}) \\ O &= (Z \circ W)A + E \end{aligned} \tag{8}$$

To showcase the utility of R-IBP on non-tabular data, we took a pretrained variational autoencoder (VAE) (Kingma & Welling, 2014) with a Gaussian latent prior[2], fed it Omniglot handwritten character images (Lake et al., 2015), and used its latent posterior means as observations. As a baseline, we used Finite Factor Analysis (FFA), implemented in scikit-learn (Pedregosa et al., 2011), sweeping the number of finite components. We found that low-noise parameters significantly overfit (Fig. 9) compared to FFA baselines, but for higher-noise parameters, achieved lower reconstruction error and negative log posterior predictive values.

6. Related Work

There is significant prior work on streaming inference as well as Bayesian nonparametric modeling. At the intersection of the two, early papers focused on mixture modeling (also known as clustering) (Lin, 2013; Tank et al., 2015; Campbell et al., 2013), but later papers considered more general nonparametric models (Campbell et al., 2015; Broderick et al., 2013a).

[2]The VAE was acquired from (Tomczak & Welling, 2018)'s publicly available code at `https://github.com/jmtomczak/vae_vampprior`.

R-IBP is similar to the Collapsed Gibbs sampler (CGS) proposed in the original IBP paper (Griffiths & Ghahramani, 2005), but differs in four critical ways. First, CGS is based on the IBP's conditional distribution, whereas R-IBP is based on the IBP's marginal distribution. Second, R-IBP never forces the indicators $z_{n,k}$ to take values in $\{0, 1\}$; rather, R-IBP's indicators exist in a superposition defined by the average over all sample paths. Third, unlike CGS, R-IBP does not marginalize out the features. Fourth and finally, CGS cannot be used on streaming data because the marginalization requires the features to follow a matrix normal distribution, yet once any data are observed, the features no longer follow a matrix normal distribution since some features shift to explain the data while other features do not. Two related IBP streaming inference papers are (Widjaja & Doshi-Velez, 2017) and (Wood & Griffiths, 2007).

7. Discussion

In this paper, we demonstrate how intelligent agents receiving streaming data can make use of infinite feature models that create new features online, as demanded by the data, in a probabilistic and principled manner. This was possible due to our novel recursive form of the Indian Buffet Process, which we termed the *Recursive IBP*. We showed that the Recursive IBP can be combined with different feature models, and that inference based on the Recursive IBP displays performance and speed close to or sometimes surpassing baseline algorithms (including some offline baseline algorithms, which have a significant advantage).

One curiosity is why Recursive-IBP performs so well. We used published code for the two Widjaja et al. and two Doshi-Velez et al. baselines, so implementation error is unlikely. Our hypothesis stems from the observation that the baselines do not use the IBP but rather its De Finetti mixing-distribution: the Beta Process, e.g. (Teh et al., 2007; Thibaux & Jordan, 2007; Doshi-Velez et al., 2009; Paisley & Carin, 2009). The consequence of using the Beta Process is that its stick-breaking constructions *chain multiply* inferred quantities. We hypothesize this multiplication causes impre-

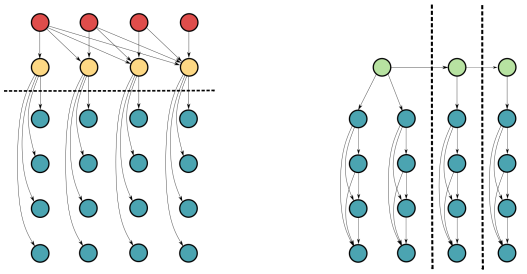

Figure 10. **Beta Process vs. Indian Buffet Process.** The Beta Process (left) chain multiplies terms (red) to compute each feature's probability (yellow) for Z (aqua), whereas the IBP (right) creates columns (green) then adds terms within columns of Z (aqua).

cisions to quickly compound. In contrast, the Recursive IBP marginalizes over feature probabilities, and instead *adds* the inferred quantities to running sums, causing errors to become less damaging with large n (Fig. 10).

This hypothesis is similar to the claim that collapsed Gibbs sampling is often superior to Gibbs sampling (Liu, 1994) because variables have been marginalized out. Although we are currently unable to prove our hypothesis, Widjaja & Doshi-Velez's "infinite" algorithm provides supporting evidence. That algorithm and R-IBP are both infinite (i.e., non-truncated) and both streaming; the *only* difference is that Widjaja et al. use a stick-breaking prior for z_{nk}, whereas we use the approximate R-IBP prior. In our experiments, R-IBP consistently outperforms Widjaja et al.'s infinite algorithm.

To emphasize one point, our particular choice of the filtering prior $q(z_n|o_{<n})$ drops dependence on future observations. Consequently, R-IBP will suffer if the smoothing and filtering distributions differ significantly. However, characterizing this difference analytically or empirically proved difficult. The challenge is that other inference algorithms we are familiar with use the stick-breaking construction of the IBP, and we couldn't think of how to disentangle the effect of assuming a different graphical structure from the effect of not revisiting past filtered distributions. Our paper is not the first to use this restriction for tractability (e.g., Marino et al. (2018)), and we attempted to remove it by adapting Campbell et al. (2021), but found their approach relies on assumptions inapplicable to the IBP. We view this as important, non-trivial future work.

Looking forward, Bayesian nonparametric models are a growing topic of interest in cognitive science and neuroscience, in studies ranging from human sensorimotor learning (Heald et al., 2021) to mouse spatial navigation (Sanders et al., 2020). We are keen to study whether R-IBP and similar streaming inference algorithms, e.g., (Schaeffer et al., 2021), can better explain behavioral and neural data.

References

Beal, M. J. Variational Algorithms for Approximate Bayesian Inference. *PhD thesis, Gatsby Computational Neuroscience Unit, UCL*, pp. 281, 2003.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*, 20:6, 2019.

Broderick, T., Boyd, N., Wibisono, A., Wilson, A. C., and Jordan, M. I. Streaming Variational Bayes. *Neural Information Processing Systems*, pp. 9, 2013a.

Broderick, T., Kulis, B., and Jordan, M. I. MAD-Bayes: MAP-based Asymptotic Derivations from Bayes. *International Conference on Machine Learning*, pp. 9, 2013b.

Campbell, A., Shi, Y., Rainforth, T., and Doucet, A. Online Variational Filtering and Parameter Learning. *arXiv:2110.13549 [cs, stat]*, October 2021. URL http://arxiv.org/abs/2110.13549. arXiv: 2110.13549.

Campbell, T., Liu, M., Kulis, B., How, J. P., and Carin, L. Dynamic Clustering via Asymptotics of the Dependent Dirichlet Process Mixture. *arXiv:1305.6659 [cs, stat]*, November 2013. URL http://arxiv.org/abs/1305.6659. arXiv: 1305.6659.

Campbell, T., Straub, J., Iii, J. W. F., and How, J. P. Streaming, Distributed Variational Inference for Bayesian Nonparametrics. *Neural Information Processing Systems*, pp. 9, 2015.

Doshi-Velez, F. and Ghahramani, Z. Accelerated sampling for the Indian Buffet Process. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, pp. 1–8, Montreal, Quebec, Canada, 2009. ACM Press. ISBN 978-1-60558-516-1. doi: 10.1145/1553374.1553409. URL http://portal.acm.org/citation.cfm?doid=1553374.1553409.

Doshi-Velez, F., Miller, K. T., Gael, J. V., and Teh, Y. W. Variational Inference for the Indian Buffet Process. *Artificial Intelligence and Statistics*, pp. 8, 2009.

Dua, D. and Graff, C. UCI Machine Learning Repository, 2017. URL http://archive.ics.uci.edu/ml.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, September 1987. ISSN 0370-2693. doi: 10.1016/0370-2693(87)91197-X. URL https://www.sciencedirect.com/science/article/pii/037026938791197X.

Ghahramani, Z., Griffiths, T. L., and Sollich, P. Bayesian Nonparametric Latent Feature Models. *Bayesian Statistics*, 8:25, 2007.

Griffiths, T. L. and Ghahramani, Z. Infinite latent feature models and the Indian buffet process. *Neural Information Processing Systems*, pp. 8, 2005.

Griffiths, T. L. and Ghahramani, Z. The Indian Buffet Process: An Introduction and Review. *Journal of Machine Learning Research*, 12(32):1185–1224, 2011. ISSN 1533-7928. URL http://jmlr.org/papers/v12/griffiths11a.html.

Heald, J. B., Lengyel, M., and Wolpert, D. M. Contextual inference underlies the learning of sensorimotor repertoires. *Nature*, 600(7889):489–493, December 2021. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-021-04129-3. URL https://www.nature.com/articles/s41586-021-04129-3.

Hyvärinen, A. Statistical Models of Natural Images and Cortical Visual Representation. *Topics in Cognitive Science*, 2(2):251–264, April 2010. ISSN 17568757, 17568765. doi: 10.1111/j.1756-8765.2009.01057.x. URL https://onlinelibrary.wiley.com/doi/10.1111/j.1756-8765.2009.01057.x.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. *International Conference on Learning Representations*, 2014. URL https://dare.uva.nl/search?identifier=cf65ba0f-d88f-4a49-8ebd-3a7fce86edd7. Publisher: Ithaca, NYarXiv.org.

Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, December 2015. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aab3050. URL https://science.sciencemag.org/content/350/6266/1332. Publisher: American Association for the Advancement of Science Section: Research Article.

Le Cam, L. An approximation theorem for the Poisson binomial distribution. *Pacific Journal of Mathematics*, 10(4):1181–1197, 1960. ISSN 0030-8730. URL https://projecteuclid.org/euclid.pjm/1103038058. Publisher: Pacific Journal of Mathematics.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998. ISSN 1558-2256. doi: 10.1109/5.726791. Conference Name: Proceedings of the IEEE.

Lin, D. Online Learning of Nonparametric Mixture Models via Sequential Variational Approximation. *Neural Information Processing Systems*, pp. 9, 2013.

Liu, J. S. The Collapsed Gibbs Sampler in Bayesian Computations with Applications to a Gene Regulation Problem. *Journal of the American Statistical Association*, 89(427):958–966, September 1994. ISSN 0162-1459. doi: 10.1080/01621459.1994.10476829. URL https://doi.org/10.1080/01621459.1994.10476829. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01621459.1994.10476829.

Marino, J., Cvitkovic, M., and Yue, Y. A General Method for Amortizing Variational Filtering. *arXiv:1811.05090 [cs, stat]*, November 2018. URL http://arxiv.org/abs/1811.05090. arXiv: 1811.05090.

Miller, K. T., Griffiths, T. L., and Jordan, M. I. Nonparametric Latent Feature Models for Link Prediction. January 2009. URL https://openreview.net/forum?id=B1b4F_bd-r.

Olshausen, B. A. and Field, D. J. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research*, 37(23):3311–3325, 1997. Publisher: Elsevier.

Paisley, J. and Carin, L. Nonparametric factor analysis with beta process priors. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, pp. 1–8, Montreal, Quebec, Canada, 2009. ACM Press. ISBN 978-1-60558-516-1. doi: 10.1145/1553374.1553474. URL http://portal.acm.org/citation.cfm?doid=1553374.1553474.

Paisley, J., Zaas, A., Woods, C. W., Ginsburg, G. S., and Carin, L. A Stick-Breaking Construction of the Beta Process. *International Conference on Machine Learning*, pp. 8, 2010.

Paisley, J., Carin, L., and Blei, D. Variational Inference for Stick-Breaking Beta Process Priors. *International Conference on Machine Learning*, pp. 8, 2011.

Paisley, J., Blei, D. M., and Jordan, M. I. Stick-Breaking Beta Processes and the Poisson Process. *AISTATS*, pp. 9, 2012.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011. URL http://jmlr.org/papers/v12/pedregosa11a.html.

Pehlevan, C., Hu, T., and Chklovskii, D. B. A Hebbian/Anti-Hebbian Neural Network for Linear Subspace Learning: A Derivation from Multidimensional Scaling of Streaming Data. *Neural Computation*, 27(7):1461–1495, July 2015. ISSN 0899-7667, 1530-888X. doi: 10.1162/NECO_a_00745. URL https://direct.mit.edu/neco/article/27/7/1461-1495/8104.

Rainforth, T., Kosiorek, A. R., Le, T. A., Maddison, C. J., Igl, M., Wood, F., and Teh, Y. W. Tighter Variational Bounds are Not Necessarily Better. *arXiv:1802.04537 [cs, stat]*, March 2019. URL http://arxiv.org/abs/1802.04537. arXiv: 1802.04537.

Sanders, H., Wilson, M. A., and Gershman, S. J. Hippocampal remapping as hidden state inference. *eLife*, 9:e51140, June 2020. ISSN 2050-084X. doi: 10.7554/eLife.51140. URL https://doi.org/10.7554/eLife.51140. Publisher: eLife Sciences Publications, Ltd.

Schaeffer, R., Bordelon, B., Khona, M., Pan, W., and Fiete, I. R. Efficient Online Inference for Nonparametric Mixture Models. *Uncertainty in Artificial Intelligence*, pp. 10, 2021.

Schwarz, G. Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461–464, 1978. ISSN 0090-5364. URL https://www.jstor.org/stable/2958889. Publisher: Institute of Mathematical Statistics.

Strack, B., DeShazo, J. P., Gennings, C., Olmo, J. L., Ventura, S., Cios, K. J., and Clore, J. N. Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records. *BioMed Research International*, 2014:e781670, April 2014. ISSN 2314-6133. doi: 10.1155/2014/781670. URL https://www.hindawi.com/journals/bmri/2014/781670/. Publisher: Hindawi.

Tank, A., Foti, N. J., and Fox, E. B. Streaming Variational Inference for Bayesian Nonparametric Mixture Models. *AISTATS*, pp. 9, 2015.

Teh, Y. W. and Görür, D. Indian buffet processes with power-law behavior. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, NIPS'09, pp. 1838–1846, Red Hook, NY, USA, December 2009. Curran Associates Inc. ISBN 978-1-61567-911-9.

Teh, Y. W., Grür, D., and Ghahramani, Z. Stick-breaking Construction for the Indian Buffet Process. In *Artificial Intelligence and Statistics*, pp. 556–563. PMLR, March 2007. URL http://proceedings.mlr.press/v2/teh07a.html. ISSN: 1938-7228.

Thibaux, R. and Jordan, M. I. Hierarchical Beta Processes and the Indian Buffet Process. In *Artificial Intelligence and Statistics*, pp. 564–571. PMLR, March 2007. URL http://proceedings.mlr.press/v2/thibaux07a.html. ISSN: 1938-7228.

Tomczak, J. M. and Welling, M. VAE with a VampPrior. *arXiv:1705.07120 [cs, stat]*, February 2018. URL http://arxiv.org/abs/1705.07120. arXiv: 1705.07120.

Wainwright, M. J. and Jordan, M. I. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc, 2008. ISBN 978-1-60198-184-4. Google-Books-ID: zp5Mo3VsJbgC.

Widjaja, F. and Doshi-Velez, F. Streaming Variational Inference for the Indian Buffet Process. *Harvard University Senior Theses*, July 2017. URL https://dash.harvard.edu/handle/1/38811474. Accepted: 2019-03-26T10:57:02Z.

Wood, F. and Griffiths, T. Particle Filtering for Nonparametric Bayesian Matrix Factorization. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2007.

A. Posterior Distribution Over Total Number of Dishes

As before, let Λ_n denote the total number of dishes after the nth customer:

$$\Lambda_n \overset{\text{def}}{=} \sum_{k=1}^{k=\infty} \min\left(1, \sum_{n'=1}^{n'=n} z_{n',k}\right)$$

Each term in the sum represents whether the kth dish (feature) exists after n customers (observations). Let's consider one term in the sum: $m_{nk} \overset{\text{def}}{=} \min(1, \sum_{n' \leq n} z_{n',k})$. We can use the following proposition to determine the distribution of m_{nk}:

Proposition A.1. *Let X be a random variable with CDF $F_X(x) = p(X \leq x)$ and let $c \in \mathbb{R}$ be a constant. Then the random variable $Y \overset{def}{=} \min(c, X)$ has a CDF $F_Y(y) = p(Y \leq y)$ given by*

$$F_Y(y) = \begin{cases} F_X(y) & \text{if } y < c \\ 1 & \text{if } y \geq c \end{cases}.$$

Substituting m_{nk} for Y, $\sum_{n' \leq n} z_{n'k}$ for X and 1 for c, it follows that

$$F_{m_{nk}|o_{\leq n}}(0) = F_{\sum z_{n'k}|o_{\leq n}}(0)$$

and

$$F_{m_{nk}|o_{\leq n}}(1) = 1.$$

We can now determine the probability mass function (PMF) of m_{nk}:

$$\begin{aligned} q(m_{nk} = 0|o_{\leq n}) &= q(m_{nk} \leq 0|o_{\leq n}) \\ &= F_{m_{nk}|o_{\leq n}}(0) \\ &= F_{\sum z_{n'k}|o_{\leq n}}(0) \\ &= q\Big(\sum_{n' \leq n} z_{n'k} \leq 0 \Big| o_{\leq n}\Big) \\ &= q\Big(\sum_{n' \leq n} z_{n'k} = 0 \Big| o_{\leq n}\Big) \end{aligned}$$

where the first and last steps follow because m_{nk} and $\sum_{n' \leq n} z_{n',k}$ can only take values in $\{0, 1, 2, ..., n\}$. Each $z_{n'k}$ is a Bernoulli random variable with distribution given by $p(z_{n'k}|o_{\leq n'})$. The sum can only be 0 if all $z_{n'k} = 0$, which occurs with probability $\prod_{n' \leq n} p(z_{n'k} = 0|o_{\leq n'})$. The PMF of m_{nk} is therefore

$$\begin{aligned} p(m_{nk} = 0|o_{\leq n}) &= \prod_{n' \leq n} p(z_{n'k} = 0|o_{\leq n}) \\ p(m_{nk} = 1|o_{\leq n}) &= F_{m_{nk}|o_{\leq n}}(1) - F_{m_{nk}|o_{\leq n}}(0) \\ &= 1 - \prod_{n' \leq n} p(z_{n'k} = 0|o_{\leq n'}) \end{aligned}$$

and $p(M_k = n) = 0$ for $n = 2, 3, ..., t$. This tells us that $M_k \sim Bernoulli(1 - \prod_{n' \leq n} p(z_{n'k} = 0|o_{\leq n}))$, which matters for two reasons. First, as $t \to \infty$, the product approaches 0 and thus the probability that the kth feature exists goes to 1, which is what we expect: given infinite data, the IBP should fill the entire feature space. Second, because Λ_n is the sum of independent but non-identically distributed Bernoullis, Le Cam's Theorem (Le Cam, 1960) again tells us Λ_n closely follows a Poisson distribution:

$$\begin{aligned} p(\Lambda_n|o_{\leq n}) &= \text{Poisson}\left(\sum_{k=1}^{k=\infty} m_{nk}\right) \\ &= \text{Poisson}\left(\sum_{k=1}^{k=\infty} \left(1 - \prod_{n'=1}^{n'=n} p(z_{n'k} = 0|o_{\leq n'})\right)\right) \end{aligned}$$

B. Variational Parameter Updates

B.1. Closed Form Expression for Variational Parameters in the Exponential Family

In the following three subsections, we use the following fact (Beal, 2003; Wainwright & Jordan, 2008): if a distribution p and its variational approximation q are both in the exponential family, then the optimal variational parameters ζ_i that correspond to the variational distribution over variable W_i are the solution to

$$\log q(W_i; \zeta_i) = \mathbb{E}_{q(W_{-i})}[\log p(W, X|\theta)] \tag{9}$$

This simply means that when optimizing the variational parameters for a variable, we can replace other variables with their expectations under the variational distribution and then solve for that one variable's parameters.

B.2. Closed Form Solutions for Linear-Gaussian Variational Parameters

We provide closed form solutions for the variational parameters for the linear Gaussian model. The model is

$$o_n = A^T z_n + \epsilon_n$$

where $A \in \mathbb{R}^{K \times D}$, $z_n \in \{0,1\}^K$, $\epsilon_n \in \mathbb{R}^D$, with Gaussian priors on A and ϵ. We posit the variational family:

$$\begin{aligned} q(z_n, A|o_{\leq n}; \theta_n) &\stackrel{\text{def}}{=} \prod_{k=1}^{k=\Lambda_n} q(z_{nk}|o_{\leq n}; b_{nk}) q(A_k|o_{\leq n}; \mu_{nk}, \Sigma_{nk}) \\ q(z_{nk}|o_{\leq n}; b_{nk}) &\stackrel{\text{def}}{=} \text{Bern}(b_{nk}) \\ q(A_k|o_{\leq n}; \mu_{nk}, \Sigma_{nk}) &\stackrel{\text{def}}{=} \mathcal{N}(\mu_{nk}, \Sigma_{nk}) \end{aligned}$$

where $\theta_n \stackrel{\text{def}}{=} \{b_{nk}\}_k \cup \{\mu_{nk}\}_k \cup \{\Sigma_{nk}\}_k$ are our variational parameters for the n observation. Our optimization problem is to maximize the approximate lower bound with respect to θ_n:

$$\mathbb{E}_{q(z_n, A|o_{\leq n}; \theta_n)}\Big[\log q(z_n|o_{<n}) + \log q(A|o_{<n}) + \log p(o_n|z_n, A)\Big] + H[q(z_n, A|o_{\leq n})]$$

where $q(A|o_{<n}) \stackrel{\text{def}}{=} q(A|o_{\leq n-1})$ and $q(z_n|o_{<n})$ is given by Eqn. 5. To find the variational parameters for the indicators z_{nl} and features A_{nl}, we will use the closed form solutions. Dropping irrelevant terms from line to line, for the binary indicators, we have:

$$\begin{aligned} \log q(z_{nl}|o_{\leq n}; b_{nl}) &= \mathbb{E}_{q(z_{n-l}, A|o_{\leq n}; \theta_n)}[\log p(o_n, z_n, A)] \\ &= \mathbb{E}_{q(z_{n-l}, A|o_{\leq n}; \theta_n)}[\log q(z_{nl}|o_{<n}) + \log p(o_n|z_n, A)] \\ \mathbb{E}_{q(z_{n-l}, A; \theta_n)}[\log q(z_{nl}|o_{<n})] &= \log q(z_{nl}|o_{<n}) \\ &= z_{nl} \log \frac{q(z_{nl}|o_{<n})}{1 - q(z_{nl}|o_{<n})} \\ \mathbb{E}_{q(z_{n-l}, A|o_{\leq n}; \theta_n)}[\log p(o_n|z_n, A)] &= -\frac{1}{2\sigma_o^2} \mathbb{E}_q[(o_n^T o_n - 2 o_n^T A^T z_n + z_n^T A A^T z_n)] \\ &= -\frac{1}{2\sigma_o^2} \mathbb{E}_q\left[\left(-2 \sum_k z_{nk} A_k^T o_n + z_n^T A A^T z_n\right)\right] \\ &= -\frac{1}{2\sigma_o^2}\left[-2 z_{nl} \mu_l^T o_n + z_{nl} \,\text{Tr}[\Sigma_{nl} + \mu_l \mu_l^T]) + 2 z_{nl} \mu_l^T \Big(\sum_{k:k \neq l} b_{nk} \mu_k\Big)\right] \end{aligned}$$

Grouping the z_{nl} terms, setting equal, substituting the canonical parameterization of the Bernoulli and solving, we have:

$$\log \frac{b_{nl}}{1-b_{nl}} = \underbrace{\log \frac{q(z_{nl}|o_{<n})}{1-q(z_{nl}|o_{<n})} - \frac{1}{2\sigma_o^2}\Big[-2\mu_l^T o_n + \mathrm{Tr}[\Sigma_{nl} + \mu_l\mu_l^T] + 2\mu_l^T\Big(\sum_{k:k\neq l} b_{nk}\mu_k\Big)\Big]}_{\stackrel{\text{def}}{=}\vartheta}$$

$$b_{nl} = \frac{1}{1+e^{-\vartheta}}$$

For the linear Gaussian parameters, we want to solve

$$\log q(A_l|o_{\leq n};\mu_{nl},\Sigma_{nl}) = \mathbb{E}_{q(z_n,A_{-l}|o_{\leq n};\theta_n)}[\log p(o_n,z_n,A)]$$

We take only the terms that depend on A_l. On the left hand side, the terms that depend on A_l are:

$$\log q(A_l|o_{\leq n};\mu_{nl},\Sigma_{nl}) \propto -\frac{1}{2}(A_l^T\Sigma_{nl}^{-1}A_l - 2A_l^T\Sigma_{nl}^{-1}\mu_{nl})$$

On the right hand side, the terms that depend on A_k are:

$$\begin{aligned}
\mathbb{E}_{q(z_n,A_{-l}|o_{\leq n};\theta_n)}[\log p(o_n,z_n,A)] &= \mathbb{E}_{q(z_n,A_{-l}|o_{\leq n};\theta_n)}[\log q(A_l|o_{<n}) + \log p(o_n|z_n,A)] \\
\mathbb{E}_{q(z_n,A_{-l}|o_{\leq n};\theta_n)}[\log q(A_l|o_{<n})] &= -\frac{1}{2}(A_l^T\Sigma_{n-1,l}^{-1}A_l - 2A_l^T\Sigma_{n-1,l}^{-1}\mu_{n-1,l}) \\
\mathbb{E}_{q(z_n,A_{-l}|o_{\leq n};\theta_n)}\log p(o_n|z_n,A)] &= -\frac{1}{2\sigma_o^2}\mathbb{E}_{q(z_n,A_{-l}|o_{\leq n};\theta_n)}[(o_n - A^Tz_n)^T(o_n - A^Tz_n)] \\
&= -\frac{1}{2\sigma_o^2}\mathbb{E}_q\left[\Big(o_n - \sum_k z_{nk}A_k\Big)^T\Big(o_n - \sum_{k'} z_{nk'}A_{k'}\Big)\right] \\
&= -\frac{1}{2\sigma_o^2}\left[-2o_n^Tb_{nl}A_l + b_{nl}A_l^TA_l + 2\Big(\sum_{k:k\neq l} b_{nk}\mu_{nk}\Big)^Tb_{nl}A_{nl}\right] \\
&= -\frac{1}{2\sigma_o^2}\left[b_{nl}A_l^TA_l + 2\Big(\sum_{k:k\neq l} b_{nk}\mu_{nk} - o_n\Big)^Tb_{nl}A_{nl}\right]
\end{aligned}$$

Setting equal, removing the $-1/2$ prefactor and completing the square gives us

$$A_l^T\Sigma_{nl}^{-1}A_l - 2A_l^T\Sigma_{nl}^{-1}\mu_{nl} = A_l^T\Sigma_{n-1,l}^{-1}A_l - 2A_l^T\Sigma_{n-1,l}^{-1}\mu_{n-1,l} + \frac{1}{\sigma_o^2}\left[b_{nl}A_l^TA_l + 2\Big(\sum_{k:k\neq l} b_{nk}\mu_{nk} - o_n\Big)^Tb_{nl}A_{nl}\right]$$

Considering terms with the form $A_l^T(\cdot)A_l$ allows us to solve for the covariance Σ_{nl}:

$$A_l^T\Sigma_{nl}^{-1}A_l = A_l^T\Big(\Sigma_{n-1,l}^{-1} + \frac{b_{nl}}{\sigma_o^2}I\Big)A_l$$

which gives us the final expression:

$$\Sigma_{nl} = \Big(\Sigma_{n-1,l}^{-1} + \frac{b_{nl}}{\sigma_o^2}I\Big)^{-1} \tag{10}$$

To find the mean μ_{nl}, we consider terms of the form $A_l^T(\cdot)\mu_{nl}$:

$$-2A_l^T\Sigma_{nl}^{-1}\mu_{nl} = -2A_l^T\Sigma_{n-1,l}^{-1}\mu_{n-1,l} + 2\frac{1}{\sigma_o^2}A^Tb_{nl}\left(\sum_{k:k\neq l} b_{nk}\mu_{nk} - o_n\right)$$

which gives us the final expression for the mean:

$$\mu_{nl} = \Sigma_{nl}\left(\Sigma_{n-1,l}^{-1}\mu_{n-1,l} + \frac{b_{nl}}{\sigma_o^2}\left(o_n - \sum_{k:k\neq l} b_{nk}\mu_{nk}\right)\right) \tag{11}$$

We add one heuristic, based on the intuition that the features should ossify as evidence accumulates to support their existence. Let μ_{nl}^* and Σ_{nl}^* denote the solutions to the previous equations. Note that the that optimization problem doesn't take into account how many observations were used to infer those parameters; the previous parameters $\mu_{n-1,l}$ and $\Sigma_{n-1,l}$ carry just as much weight regardless of wheter $n = 2$ or $n = 10^{10}$. Consequently, instead of accepting outright the solutions $\mu_{nl}^*, \Sigma_{nl}^*$, we take a number-of-observations weighted average:

$$\mu_{nl} \propto q(z_{nl} = 1|o_{\leq n})\mu_{nl}^* + \left(\sum_{n'<n} q(z_{n'k} = 1|o_{\leq n'})\right)\mu_{n-1,k}$$
$$\Sigma_{nl} \propto q(z_{nl} = 1|o_{\leq n})\Sigma_{nl}^* + \left(\sum_{n'<n} q(z_{n'k} = 1|o_{\leq n'})\right)\Sigma_{n-1,k}$$

These running sums are already available from the recursion and thus require no additional time or space.

B.3. Closed Form Solutions for Factor Analysis Variational Parameters

We provide closed form solutions for the variational parameters for the Factor Analysis model. The model is

$$o_n = A^T(z_n \circ w_n) + \epsilon_n$$

where $w_n \in \mathbb{R}^K \sim \mathcal{N}(0, \Sigma_w)$ and $\circ$ denotes element-wise multiplication. We posit the variational family:

$$q(z_n, w_n, A|o_{\leq n}; \theta_n) \stackrel{\text{def}}{=} \prod_{k=1}^{k=\Lambda_n} q(z_{nk}|o_{\leq n}; b_{nk})q(w_n|o_{\leq n}; \phi_n, \Phi_n)q(A_k|o_{\leq n}; \mu_{nk}, \Sigma_{nk})$$
$$q(z_{nk}|o_{\leq n}; b_{nk}) \stackrel{\text{def}}{=} \text{Bern}(b_{nk})$$
$$q(w_n|o_{\leq n}; \phi_n, \Phi_n) \stackrel{\text{def}}{=} \mathcal{N}(\phi_n, \Phi_n)$$
$$q(A_k|o_{\leq n}; \mu_{nk}, \Sigma_{nk}) \stackrel{\text{def}}{=} \mathcal{N}(\mu_{nk}, \Sigma_{nk})$$

where $\theta_n \stackrel{\text{def}}{=} \{b_{nk}\}_k \cup \{\phi_n, \Phi_n\} \cup \{\mu_{nk}\}_k \cup \{\Sigma_{nk}\}_k$ are our variational parameters for the n observation. Our optimization problem is to maximize the approximate lower bound with respect to θ_n:

$$\mathbb{E}_{q(z_n,w_n,A|o_{\leq n};\theta_n)}\Big[\log q(z_n|o_{<n}) + \log p(w_n) + \log q(A|o_{<n}) + \log p(o_n|z_n, w_n, A)\Big] + H[q(z_n, w_n, A|o_{\leq n})]$$

where $q(A|o_{<n}) \stackrel{\text{def}}{=} q(A|o_{\leq n-1}; \mu_{n-1,k}, \Sigma_{n-1,k})$ and $q(z_n|o_{<n})$ is given in the main text as:

$$q(z_{nk}|o_{<n}) \stackrel{\text{def}}{=} \frac{1}{\beta + n - 1}\sum_{n'<n} q(z_{n'k} = 1|o_{\leq n'}) + q(\Lambda_{n-1} \leq k - 1|o_{<n}) - q(\Lambda_{n-1} + \lambda_n \leq k - 1|o_{<n})$$

We use the same approach as for the linear Gaussian model. Starting with the binary indicator variables z_{nk}, we want to

solve:

$$
\begin{aligned}
\log q(z_{nl}; b_{nl}) &= \mathbb{E}_{q(z_{n\,-l}, w_n, A | o_{\leq n}; \theta_n)}[\log p(z_n, w_n, o_n, A)] \\
z_{nl} \log \frac{b_{nl}}{1 - b_{nl}} &= \mathbb{E}_{q(z_{n-l}, w_n, A | o_{\leq n}; \theta_n)}[\log p(z_n, w_n, o_n, A)] \\
&= \mathbb{E}_{q(z_{n-l}, A | o_{\leq n}; \theta_n)}[\log q(z_{nl} | o_{<n}) + \log p(o_n | z_n, w_n, A)] \\
&= z_{nl} \log \frac{q(z_{nl} | o_{<n})}{1 - q(z_{nl} | o_{<n})} \\
&\quad - \frac{1}{2\sigma_o^2} \mathbb{E}_q[(o_n^T o_n - 2 o_n^T A^T (z_n \circ w_n) + (z_n \circ w_n)^T A A^T (z_n \circ w_n))] \\
&= z_{nl} \log \frac{q(z_{nl} | o_{<n})}{1 - q(z_{nl} | o_{<n})} \\
&\quad - \frac{1}{2\sigma_o^2} \Big(- 2 o_n^T \mu_{nl} z_{nl} \phi_{nl} + \mathbb{E}_q \left[\sum_k z_{nk}^2 w_{nk}^2 A_k^T A_k + \sum_{k,k' : k \neq k'} z_{nk} w_{nk} z_{nk'} w_{nk'} A_k^T A_{k'} \right] \Big) \\
&= z_{nl} \log \frac{q(z_{nl} | o_{<n})}{1 - q(z_{nl} | o_{<n})} \\
&\quad - \frac{1}{2\sigma_o^2} \left(- 2 o_n^T \mu_{nl} z_{nl} \phi_{nl} + z_{nl} [\phi_{nl}^2 + \Phi_{nll}] \operatorname{Tr}[\Sigma_{nl} + \mu_{nl} \mu_{nl}^T] + 2 z_{nl} \phi_{nl} \mu_{nl}^T \Big(\sum_{k : l \neq k} b_{nk} \phi_{nk} \mu_k \Big) \right)
\end{aligned}
$$

Grouping the z_{nl} terms, setting equal, substituting the canonical parameterization of the Bernoulli and solving, we have:

$$
\log \frac{b_{nl}}{1 - b_{nl}} = \underbrace{\log \frac{q(z_{nl} | o_{<n})}{1 - q(z_{nl} | o_{<n})} - \frac{1}{2\sigma_o^2} \left[- 2 o_n^T \mu_{nl} \phi_{nl} + [\phi_{nl}^2 + \Phi_{nll}] \operatorname{Tr}[\Sigma_{nl} + \mu_{nl} \mu_{nl}^T] + 2 \phi_{nl} \mu_{nl}^T \Big(\sum_{k : l \neq k} b_{nk} \phi_{nk} \mu_k \Big) \right]}_{\stackrel{\text{def}}{=} \vartheta}
$$

$$
b_{nl} = \frac{1}{1 + e^{-\vartheta}}
$$

Next, for the scaling weights w_n, we want to solve the following equation for mean ϕ_n and covariance Φ_n:

$$
\begin{aligned}
\log q(w_n; \phi_n, \Phi_n) &= \mathbb{E}_{q(z_n, A | o_{\leq n}; \theta_n)}[\log p(z_n, w_n, o_n, A)] \\
-\frac{1}{2}(w_n^T \Phi_n^{-1} w_n - 2 \phi_n^T \Phi_n^{-1} w_n) &= \mathbb{E}_{q(z_n, A | o_{\leq n}; \theta_n)}[\log p(w_n) + \log p(o_n | z_n, w_n, A)] \\
&= -\frac{1}{2} w_n^T \Sigma_w^{-1} w_n - \frac{1}{2\sigma_o^2} \mathbb{E}_q \Big[(o_n^T o_n - 2 o_n^T A^T (z_n \circ w_n) + (z_n \circ w_n)^T A A^T (z_n \circ w_n)) \Big] \\
&= -\frac{1}{2} w_n^T \Sigma_w^{-1} w_n - \frac{1}{2\sigma_o^2} \Big(- 2 o_n^T \mu_n^T \operatorname{diag}(b_n) w_n + \mathbb{E}_q[(z_n \circ w_n)^T A A^T (z_n \circ w_n)] \Big) \\
&= -\frac{1}{2} w_n^T \Sigma_w^{-1} w_n - \frac{1}{2\sigma_o^2} \Big(- 2 o_n^T \mu_n^T \operatorname{diag}(b_n) w_n + w_n^T \mathbb{E}_q[\operatorname{diag}(z_n)^T A A^T \operatorname{diag}(z_n)] w_n \Big)
\end{aligned}
$$

The term $\mathbb{E}_q[\operatorname{diag}(z_n)^T A A^T \operatorname{diag}(z_n)]$ is slightly trickier. We take the expectation with respect to A, then z:

$$
\begin{aligned}
\mathbb{E}_{q(A)}[AA^T]_{ij} &= \mathbb{E}_q[A_i^T A_j] \\
&= \begin{cases} \mathrm{Tr}[\mu_{ni}\mu_{nj}^T] & i \neq j \\ \mathrm{Tr}[\mu_{ni}\mu_{ni}^T + \Sigma_{ni}] & i = j \end{cases} \\
&\stackrel{\text{def}}{=} M_{ij} \\
\mathbb{E}_{q(z_n,A)}[\mathrm{diag}(z_n)^T AA^T \,\mathrm{diag}(z_n)]_{ij} &= \mathbb{E}_{q(z_n)}[\mathrm{diag}(z_n)^T M \,\mathrm{diag}(z_n)]_{ij} \\
&= \mathbb{E}_{q(z_n)}[z_{ni} M_{ij} z_{nj}] \\
&= \begin{cases} b_{ni} M_{ij} b_{nj} & i \neq j \\ b_{ni} M_{ii} & i = j \end{cases} \\
&\stackrel{\text{def}}{=} S_{ij} \\
-\frac{1}{2} w_n^T \Phi_n^{-1} w_n &= -\frac{1}{2} w_n^T \Sigma_w^{-1} w_n - \frac{1}{2\sigma_o^2}\left(w_n^T S w_n\right)
\end{aligned}
$$

We can then solve for Φ_n:

$$
\Phi_n = \left(\Sigma_w^{-1} + \frac{1}{\sigma_o^2} S\right)^{-1} \tag{12}
$$

Solving for ϕ_n similarly gives:

$$
\begin{aligned}
-\frac{1}{2}\left(-2\phi_n^T \Phi_n^{-1}\right) w_n &= -\frac{1}{2\sigma_o^2}\left(-2 o_n^T \mu_n^T \,\mathrm{diag}(b_n)\right) w_n \\
\phi_n &= \frac{1}{\sigma_o^2} \Phi_n^T \,\mathrm{diag}(b_n) \mu_n o_n
\end{aligned}
$$

Lastly, for the feature values A_k, we solve the following equation to obtain the mean μ_{nk} and covariance Σ_{nk}:

$$
\begin{aligned}
\log q(A_l | o_{\leq n}; \mu_{nl}, \Sigma_{nl}) &= \mathbb{E}_{q(z_n, w_n, A_{-l} | o_{\leq n}; \theta_n)}[\log p(z_n, w_n, o_n, A)] \\
&= \mathbb{E}_{q(z_n, w_n, A_{-l} | o_{\leq n}; \theta_n)}[\log q(A_l | o < n)] + \mathbb{E}_{q(z_n, w_n, A_{-l} | o_{\leq n}; \theta_n)}[\log p(o_n | z_n, w_n, A)]
\end{aligned}
$$

As before, taking only the terms depending on A_l gives:

$$
\log q(A_l | o_{\leq n}; \mu_{nl}, \Sigma_{nl}) = -\frac{1}{2}\left(A_l^T \Sigma_{nl}^{-1} A_l - 2 A_l^T \Sigma_{nl}^{-1} \mu_{nl}\right)
$$

$$
\mathbb{E}_q\left[\log q(A_l | o < n)\right] = -\frac{1}{2}\left(A_l^T \Sigma_{n-1,l}^{-1} A_l - 2 A_l^T \Sigma_{n-1,l}^{-1} \mu_{n-1,l}\right)
$$

$$
\begin{aligned}
\mathbb{E}_q\left[\log p(o_n | z_n, w_n, A)\right] &= -\frac{1}{2\sigma_o^2} \mathbb{E}_q\left[\left(o_n - A^T(z_n \circ w_n)\right)^T \left(o_n - A^T(z_n \circ w_n)\right)\right] \\
&= -\frac{1}{2\sigma_o^2} \mathbb{E}_q\left[-2 o_n^T \sum_k z_{nk} w_{nk} A_k + \sum_k z_{nk}^2 w_{nk}^2 A_k A_k^T + \sum_{k,k':k\neq k'} z_{nk} z_{nk'} w_{nk} w_{nk'} A_k A_{k'}^T\right] \\
&= -\frac{1}{2\sigma_o^2}\left[-2 o_n^T b_{nl} \phi_{nl} A_l + b_{nl}\left[\Phi_{nll} + \phi_{nl}^2\right] A_l^T A_l + 2 b_{nl} \phi_{nl}\left(\sum_{k:k\neq l} b_{nk} \mu_k \phi_{nk}\right)^T A_l\right]
\end{aligned}
$$

Setting the two sides equal, removing the $-\frac{1}{2}$ prefactor, and considering terms of the form $A_l^T(\cdot)A_l$ allows us to solve for the covariance Σ_{nl}:

$$A_l^T\Sigma_{nl}^{-1}A_l = A_l^T\left(\Sigma_{n-1,l}^{-1} + \frac{b_{nl}}{\sigma_o^2}\left[\Phi_{nll} + \phi_{nl}^2\right]I\right)A_l$$

which yields the final expression:

$$\Sigma_{nl} = \left(\Sigma_{n-1,l}^{-1} + \frac{b_{nl}}{\sigma_o^2}\left[\Phi_{nll} + \phi_{nl}^2\right]I\right)^{-1}$$

To obtain the mean μ_{nl}, we consider terms of the form $A_l(\cdot)\mu_{nl}$:

$$-2A_l^T\Sigma_{nl}^{-1}\mu_{nl} = -2A_l^T\Sigma_{n-1,l}^{-1}\mu_{n-1,l} + 2A_l^T\left(\frac{b_{nl}\phi_{nl}}{\sigma_o^2}\sum_{k:k\neq l}b_{nk}\mu_k\phi_{nk} - o_n\right)$$

which gives the final expression:

$$\mu_{nl} = \Sigma_{nl}\left(\Sigma_{n-1,l}^{-1}\mu_{n-1,l} + \frac{b_{nl}\phi_{nl}}{\sigma_o^2}\left(o_n - \sum_{k:k\neq l}b_{nk}\mu_k\phi_{nk}\right)\right)$$

C. Theory

C.1. Summary of (Broderick et al., 2013b)

Broderick, Kulis & Jordan ICML's 2013 paper "MAD-Bayes: MAP-based Asymptotic Derivations from Bayes" shows that Kulis & Jordan's 2012 DP-Means can be derived in a different manner, as the zero-noise limit of the MAP estimator of a Dirichlet Process Gaussian mixture model. With this view, they also consider a zero-noise limit of the MAP estimator of a Beta-Bernoulli Process Linear-Gaussian feature model. Letting K^+ denote the inferred number of dishes, the high level idea is that the MAP estimator is the solution to the following optimization problem:

$$\arg\max_{Z,A,K^+} p(Z,A,K^+|O) = \arg\max_{Z,A,K^+} p(Z,A,K^+,O) = \arg\max_{Z,A,K^+} p(X|Z,A,K^+)p(Z,K^+)p(A)$$

If A has a matrix normal prior, Z a Beta-Bernoulli prior with concentration parameter α, $X|Z,A$ a matrix normal likelihood with covariance $\sigma_o^2 I$, then under the zero noise limit (i.e. $\sigma_o^2 \to 0$) and reparameterizing $\alpha = \exp(-\lambda^2/2\sigma_o^2)$, the objective function can be written:

$$\arg\min_{Z,A,K^+} \mathrm{Tr}[(X-ZA)^T(X-ZA)] + K^+\lambda^2 \tag{13}$$

Broderick et al. then define an algorithm BP-Means and show it converges to a local optimum. My goal is to define a similar optimization problem for R-IBP and show that R-IBP monotonically improves.

C.2. Summary of R-IBP for Linear Gaussian Data

We consider the Linear-Gaussian generative model:

$$X = ZA + \epsilon$$

We posit the following variational family, which is a fancy way of saying (a) each z_{nk} is a Bernoulli with parameter b_{nk} and (b) each feature A_k is a Normal with mean μ_k and covariance Σ_k:

$$q(z_n, A|o_{\leq n};\theta_n) \stackrel{\text{def}}{=} \prod_{k=1}^{k=\Lambda_n} q(z_{nk}|o_{\leq n};b_{nk})q(A_k|o_{\leq n};\mu_{nk},\Sigma_{nk})$$

$$q(z_{nk}|o_{\leq n};b_{nk}) \stackrel{\text{def}}{=} \text{Bern}(b_{nk})$$

$$q(A_k|o_{\leq n};\mu_{nk},\Sigma_{nk}) \stackrel{\text{def}}{=} \mathcal{N}(\mu_{nk},\Sigma_{nk})$$

On each time step n, we perform coordinate ascent, changing the variational parameters. For the Bernoulli parameters, the updates are given by

$$\log \frac{b_{nl}}{1-b_{nl}} = \underbrace{\log \frac{q(z_{nl}|o_{<n})}{1-q(z_{nl}|o_{<n})} - \frac{1}{2\sigma_o^2}\Big[-2\mu_l^T o_n + \mathrm{Tr}[\Sigma_{nl} + \mu_l \mu_l^T] + 2\mu_l^T \Big(\sum_{k:k\neq l} b_{nk}\mu_k \Big) \Big]}_{\stackrel{\text{def}}{=} \vartheta}$$

$$b_{nl} = \frac{1}{1+e^{-\vartheta}}$$

For the Normal parameters, the updates are given by:

$$\Sigma_{nl} = \Big(\Sigma_{n-1,l}^{-1} + \frac{b_{nl}}{\sigma_o^2} I \Big)^{-1}$$

$$\mu_{nl} = \Sigma_{nl} \Big(\Sigma_{n-1,l}^{-1} \mu_{n-1,l} + \frac{b_{nl}}{\sigma_o^2} \big(o_n - \sum_{k:k\neq l} b_{nk}\mu_{nk} \big) \Big)$$

C.3. Zero Noise Limit of R-IBP

We repeat Thm. 4.1 for ease of reading. The proof follows.

Theorem C.1. *For all k, initialize A_k's variational parameters $\mu_{0k} = 0$ and $\Sigma_{0k} \sim O(1)$ with respect to σ_o^2. On each n and for all k, initialize z_{nk}'s variational parameters $b_{nk} \sim O(1)$ with respect to σ_o^2. Reparameterize $\alpha \stackrel{def}{=} \exp(-\gamma^2/2\sigma_o^2)$. Then in the limit $\sigma_o^2 \to 0$, R-IBP minimizes Eqn. (13).*

Lemma C.2. *Under the above assumptions, R-IBP and BP-Means populate the $Z \in \{0,1\}^K$ and $A \in \mathbb{R}^{K\times D}$ matrices with the same values after a single pass through the data.*

Proof. We prove Lemma C.2 via induction.

Base Case: Consider the first observation ($n = 1$) and first feature ($k = 1$). We initialize b_{11} at $q(z_{11} = 1|o_{<1}) = q(z_{11} = 1)$, which is the sum of the probabilities that $k \geq 1$ features are added:

$$q(z_{11} = 1; \alpha) = \sum_{k=1}^{\infty} \frac{\alpha^k}{k!} e^{-\alpha} \quad \stackrel{\sigma_o^2 \to 0}{=} \quad O(\alpha e^{-\alpha})$$

The update for this first feature's covariance is given by:

$$\begin{aligned}
\Sigma_{11} &= \Big(\Sigma_{01}^{-1} + \frac{q(z_{11}=1;\alpha)}{\sigma_o^2} I \Big)^{-1} \\
&= \frac{\sigma_o^2}{q(z_{11}=1;\alpha)} \Big(\frac{\sigma_o^2}{q(z_{11}=1;\alpha)} \Sigma_{01}^{-1} + I \Big)^{-1} \\
&= \frac{\sigma_o^2}{q(z_{11}=1;\alpha)} \sum_{i=0}^{i=\infty} \Big(\frac{\sigma_o^2}{q(z_{11}=1;\alpha)} \Sigma_{01}^{-1} \Big)^i (-1)^i \\
&\stackrel{\sigma_o^2 \to 0}{=} 0(I) \\
&= 0
\end{aligned}$$

where the second to last step is the Neumann series of the matrix $(\frac{\sigma_o^2}{q(z_{11}=1;\alpha)} \Sigma_{01}^{-1} + I)^{-1}$, which is applicable because the matrix $\frac{\sigma_o^2}{q(z_{11}=1;\alpha)} \Sigma_{01}^{-1}$ has spectral radius < 1. Intuitively, this makes sense: when the noise vanishes, we should be more

confident with where our features are. Now we turn to the updates for the mean:

$$\begin{aligned}\mu_{11} &= \Sigma_{11}\Big(\Sigma_{01}^{-1}\mu_{01} + \frac{b_{01}}{\sigma_o^2}\big(o_n - \sum_{k>1} b_{1k}\mu_{1k}\big)\Big) \\ &= \frac{\sigma_o^2}{q(z_{11}=1;\alpha)}\Big(\frac{\sigma_o^2}{q(z_{11}=1;\alpha)}\Sigma_{01}^{-1} + I\Big)^{-1}\Big(\frac{q(z_{11}=1;\alpha)}{\sigma_o^2}o_1\Big) \\ &\overset{\sigma_o^2\to 0}{=} o_1\end{aligned}$$

We next update the Bernoulli variational parameter, recalling that $b_{11} = 1/(1+e^{-\vartheta_{nk}})$ where:

$$\begin{aligned}\vartheta_{nk} &\overset{\text{def}}{=} \log\frac{\alpha e^{-\alpha}}{1-\alpha e^{-\alpha}} - \frac{1}{2\sigma_o^2}\Big(-2\mu_{11}^T(o_n - \sum_{k>1} b_{nk}\mu_{nk}) + \mathrm{Tr}[\Sigma_{11} + \mu_{11}\mu_{11}^T]\Big) \\ &= \log\alpha - \alpha - \log(1-\alpha e^{-\alpha}) - \frac{1}{2\sigma_o^2}\mathrm{Tr}[\Sigma_{11}] + \frac{1}{2\sigma_o^2}(o_n^T o_n) \\ &\overset{\sigma_o^2\to 0}{=} -\frac{\gamma^2}{2\sigma_o^2} - 0 - 0 + \frac{1}{2\sigma_o^2}(o_n^T o_n) \\ &\overset{\sigma_o^2\to 0}{=} \begin{cases} -\infty & \text{if } \gamma^2 > o_1^T o_1 \\ 0 & \text{if } \gamma^2 = o_1^T o_1 \\ \infty & \text{if } \gamma^2 < o_1^T o_1 \end{cases} \quad\Rightarrow\quad b_{11} = \begin{cases} 0 & \text{if } \gamma^2 > o_1^T o_1 \\ 0.5 & \text{if } \gamma^2 = o_1^T o_1 \\ 1 & \text{if } \gamma^2 < o_1^T o_1 \end{cases}\end{aligned}$$

Next, consider the first observation ($n = 1$) and any feature beyond the first ($k > 1$). Because $\mu_{11} = o_1$ and $b_{11} = 1$, the observation is fully explained and so $o_n - b_{11}\mu_{11} = 0$, so all $b_{1k} = 0$ and no further features will emerge. Note that this is identical to Broderick et al.'s BP-Means on the first pass.

Inductive Step. Assume that by the $(n-1)$th observation, inclusive, R-IBP has filled the first $n-1$ rows in Z and A with the same values as BP-Means. We show that for the nth observation, R-IBP and BP-Means fill the nth row with the same values. First, note that the total number of features Λ_{n-1} is known exactly because $\forall n' \le n-1, \forall k$, we have that $b_{n'k} \in \{0,1\}$. We need to consider what each algorithm will do in 1 of two cases:

1. Columns corresponding to existing dishes/features i.e. $k \in [1, \Lambda_{n-1}]$. In this case, Eqn. 5 dictates that

$$q(z_{nk}=1|o_{<n}) = \frac{\alpha}{\beta+n-1}\sum_{n'<n} b_{n'k} + \underbrace{q(\Lambda_{n-1} \le k-1|o_{<n})}_{=0} - \underbrace{q(\Lambda_{n-1}+\lambda_n \le k-1|o_{<n})}_{=0}$$

   which is $\sim O(1)$ with respect to σ_o^2. Consequently, the update for the Bernoulli variational parameter becomes:

$$\vartheta \overset{\sigma_o^2\to 0}{=} \frac{1}{2\sigma_o^2}2\mu_{nk}^T\big(o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'}\big) - \frac{1}{2\sigma_o^2}\mu_{nk}^T\mu_{nk}$$

   If the inner product of μ_{nk} with the unexplained remainder $o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'}$ is more than half the inner product of μ_{nk} with itself, R-IBP will set $b_{nk} = 1$ and if not, R-IBP will set $b_{nk} = 0$. This is precisely what BP-Means does. This is because in BP-Means, b_{nk} is set to 1 if

$$\begin{aligned}&(o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'})^T(o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'}) - 2\mu_{nk}^T(o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'}) + \mu_{nk}^T\mu_{nk} \\ &\quad< (o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'})^T(o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'})\end{aligned}$$

   and 0 otherwise. Simplifying, we see that the BP-Means criterion is identical to the R-IBP criterion:

$$\frac{1}{2}\mu_{nk}^T\mu_{nk} < \mu_{nk}^T(o_n - \sum_{k'\neq k} b_{nk'}\mu_{nk'})$$

2. Columns corresponding to new dishes/features i.e. $k \in (\Lambda_{n-1}, \Lambda_{n-1} + \lambda_n]$

□