Efficient Streaming Inference for Infinite Latent Feature Models

Anonymous Authors

Abstract
Bayesian nonparametrics are a classic Bayesian approach to continual learning, in which the non-parametric model grows in representational capacity as more data are observed. One Bayesian nonparametric model, the Indian Buffet Process (IBP), is used when the stream of data is assumed to be generated by an unknown number of binary latent features. However, use of the IBP has been limited by the lack of inference algorithms for the online setting. Here we propose a Bayesian recursion called the Recursive IBP (R-IBP) that efficiently filters a posterior distribution over latent features from a sequence of observations in a streaming manner. R-IBP can be combined with a variety of other models whenever the number of components is unknown, including independent component analysis (ICA), non-negative matrix factorization (NMF), and factor analysis (FA). Our recursive filter has quasilinear average time complexity and logarithmic average space complexity in the number of observations.

1. Introduction
A central challenge of continual learning is defining a model capable of growing in representational capacity as more data are observed. One approach is to define a model with an infinite number of parameters, under the constraint that only a finite number of parameters may be used for a finite amount of data; then, as more data are observed, more parameters become available for use and the model can more flexibly model the data. This approach, called Bayesian nonparametrics (BNPs), offers a principled and powerful solution to continual learning.

However, despite the seemingly clear relevance of BNPs, these models are not widely used (although see (Mehta et al., 2021; Lee et al., 2020; Kessler et al., 2019)). One reason may be that most, if not all, inference algorithms (both sampling and variational) for BNP models are not applicable in the streaming setting. Recent work sought to rectify this shortcoming by providing an efficient streaming inference algorithm for an “infinite” clustering model called the Chinese Restaurant Process (Schaeffer et al., 2021). In this paper, we propose an efficient streaming inference algorithm for an “infinite” feature model called the Indian Buffet Process (IBP), which has been used in the continual learning literature (Kessler et al., 2019; Mehta et al., 2021). We start by introducing our precise inference problem, then define the IBP and its utility, and finally detail our proposed inference algorithm.

2. Background
2.1. Generative Model
We consider a latent variable time series model with \( K \)-dimensional binary latent variables \( z_{1:T} \) (i.e. \( z_t \in \{0, 1\}^K \)) and observable variables \( o_{1:T} \), where \( K \) is unknown and \( 1:T \) denotes the sequence \( \{1, 2, ..., T\} \). Our generative process assumes an Indian Buffet Process (IBP) prior over the sequence of latent states:

\[
  z_{1:T} \sim IBP(\alpha, \beta)
\]

\[
o_t | z_t \sim p(o | z)
\]

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 finite rows and unbounded number of columns. The term IBP arises from an analogy of a sequence of customers (rows) arriving at an Indian buffet and choosing an unbounded number of dishes (columns). The \( t \)th customer samples an integer number of new dishes \( \lambda_t \sim Poisson(\alpha \beta / (\beta + t - 1)) \) and then samples previous dishes with probability proportional to the number of previous customers who sampled said dishes. Denoting the total number of dishes after the first \( t \) customers as \( \Lambda_t = \sum_{t'=1}^{t} \lambda_{t'} \), the IBP defines a conditional distribution for the \( t \)th row and column given the previous \( (t' \leq t) \) rows and columns.

\(^1\)The original IBP paper (Griffiths & Ghahramani, 2005) defined a single parameter model that (Ghahramani et al., 2007) extended to two parameters and that (Teh & Görür, 2009) extended to three. This paper applies equally to all, but since our focus is on efficient streaming inference and not particular properties of a specific model, we chose the 2 parameter IBP to balance expositional simplicity and model complexity.
3.1. Objective

Our goal is to infer a posterior over the current observation’s binary latent variables \( z_t \), given the entire history of observations i.e. filter \( p(z_t | o_{\leq t}) \), subject to two constraints:

1. Inference must be performed online
2. Inference must be efficient in the large observation (i.e. time) limit

In a time series context, inferring \( p(z_t | o_{\leq t}) \) is often called filtering (e.g. Kalman filter, particle filter).

3.2. Bayesian Recursion

The challenge with using the IBP in a streaming setting is that its conditional form renders the current latent variables \( z_t \) dependent on all previous latent variables \( z_{\leq t} \). Our approach is to break the dependence by converting the conditional into a marginal, which can be expressed as a running sum. For brevity, we refer to the prior on the current latent variables \( p(z_t | o_{\leq t}) \) as the “latent prior” and the posterior on the current variables \( p(z_t | o_{\leq t}) \) as the “latent posterior”.

Bayes’ rule relates the latent prior to the latent posterior:

\[
p(z_t = 1 | o_{\leq t}) = \frac{p(o_t | z_t = 1) p(z_t = 1)}{p(o_t | z_t = 1) p(z_t = 1) - p(z_t = 1 | o_{\leq t})}
\]

(3)

The latent prior \( p(z_t = 1 | o_{\leq t}) \) can be rewritten as the expectation of an indicator random variable that we can expand using the Law of Total expectation. Suppressing \( \alpha, \beta \) for brevity, the IBP’s marginal distribution is the expectation of the IBP’s now-random conditional distribution:

\[
p(z_t = 1 | o_{\leq t}) = \mathbb{E}_t[p(z_t = 1 | o_{\leq t})]
\]

Substituting Eqn. 2, taking the expectations and simplifying yields

\[
p(z_t = 1 | o_{\leq t}) = \frac{1}{\beta + t - 1} \sum_{t' < t} p(z_{t'} = 1 | o_{\leq t}) + p(\Lambda_{t-1} \leq k - 1 | o_{\leq t}) - p(\Lambda_{t-1} + \lambda_t \leq k - 1 | o_{\leq t})
\]

(4)

where \( k \leq \Lambda_{t-1} \) disappears because the probability of a customer eating a non-existent dish is 0 (i.e. \( p(z_{t'k} = 1, k > \Lambda_{t-1} | o_{\leq t}) = 0 \)) and \( p(z_{t'k} = 1 | o_{\leq t}) = p(z_{t'k} = 1, k > \Lambda_{t-1} | o_{\leq t}) \) and where the other two terms are a difference of two cumulative distribution functions (CDFs)

\[
p(\Lambda_{t} = k < \Lambda_{t-1} + \lambda_t | o_{\leq t}) = 1 - p(\Lambda_{t-1} < k \leq \Lambda_{t-1} + \lambda_t | o_{\leq t})
\]

where \( (\Lambda_{t-1} + \lambda_t | o_{\leq t}) \) is a cumulative distribution function.
not possible in the streaming context. We therefore assume latent posteriors depend only on data seen up to that point i.e. p(z_{t|k}|o_{\leq t}) = p(z_{t|k}|o_{\leq t'}) for t' \leq t, precluding retroactive revision. This yields our key expression for the latent prior:

\[
p(z_{tk} = 1|o_{\leq t}) \approx \frac{1}{\beta + t - 1} \sum_{t' < t} p(z_{t'k} = 1|o_{\leq t'})
\]

\[+ p(\Lambda_{t-1} \leq k - 1|o_{\leq t}) - p(\Lambda_{t-1} + \lambda_t \leq k - 1|o_{\leq t})
\]

(5)

We call this streaming inference algorithm the **Recursive Indian Buffet Process (R-IBP)**. Intuitively, Eqn. 5 tells us that the prior probability the tth observation was generated by the kth feature is approximately given by the running sum of the previous customers’ posterior probabilities of also being generated by the kth feature, plus one other term. This running sum of probability masses of the latent binary variables is a “soft” version of the IBP’s natural behavior, which would ordinarily add the realizations of the binary variables if the variables were observed. The other term can be shown to be a difference of two Poisson CDFs, which we discuss below; its role is to incentivize the creation of new features. As new observations arrive, these two terms compete to explain the observations: the first term advises the observation can be explained using previous features, proportional to how common those features are, while the second term advises the new observation might be better explained by creating new, additional features. This recursion is visually displayed in Fig 1 for \(\alpha = 30.91, \beta = 13.82\).

We now return to the difference of CDFs to show why the posterior over the total number of dishes remains Poisson and to give its exact rate. As before, let \(\Lambda_t\) denote the total number of dishes after the \(t\)th customer:

\[
\Lambda_t \overset{def}{=} \sum_{k=1}^{k=\infty} \min \left( 1, \sum_{t' = 1}^{t' = t} z_{t'k} \right)
\]

Each term in the sum represents whether the \(k\)th feature was present in the first \(t\) customers. Consider one term in the sum, \(M_{tk} \overset{def}{=} \min(1, \sum_{t' \leq t} z_{t'k})\). We can use the following proposition to determine the distribution of \(M_{tk}\):

**Proposition 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 \(M_{tk}\) for \(Y\), \(\sum_{t' \leq t} z_{t'k}\) for \(X\) and 1 for \(c\), it follows that

\[
F_{M_{tk}|o_{\leq t}}(0) = F_{\sum_{t' \leq t} z_{t'k}|o_{\leq t}}(0)
\]

and

\[
F_{M_{tk}|o_{\leq t}}(1) = 1.
\]

We can now determine the probability mass function (PMF) of \(M_{tk}\):

\[
p(M_{tk} = 0|o_{\leq t}) = p(M_{tk} \leq 0|o_{\leq t})
\]

\[= F_{M_{tk}|o_{\leq t}}(0)
\]

\[= F_{\sum_{t' \leq t} z_{t'k}|o_{\leq t}}(0)
\]

\[= p\left( \sum_{t' \leq t} z_{t'k} \leq 0 | o_{\leq t} \right)
\]

\[= p\left( \sum_{t' \leq t} z_{t'k} = 0 | o_{\leq t} \right)
\]

where the first and last steps follow because \(M_{tk}\) and \(\sum_{t' \leq t} z_{t'k}\) can only take values in \(\{0, 1, 2, ..., t\}\). Each \(z_{t'k}\) is a Bernoulli random variable with distribution given by \(p(z_{t'k}|o_{\leq t'})\). The sum can only be 0 if all \(z_{t'k} = 0\), which occurs with probability \(\prod_{t' \leq t} p(z_{t'k} = 0|o_{\leq t'})\). The PMF of \(M_{tk}\) is therefore

\[
p(M_{tk} = 0|o_{\leq t}) = \prod_{t' \leq t} p(z_{t'k} = 0|o_{\leq t'})
\]

\[
p(M_{tk} = 1|o_{\leq t}) = F_{M_{tk}|o_{\leq t}}(1) - F_{M_{tk}|o_{\leq t}}(0)
\]

\[= 1 - \prod_{t' \leq t} p(z_{t'k} = 0|o_{\leq t'})
\]

and \(p(M_k = n) = 0\) for \(n = 2, 3, ..., t\). This tells us that \(M_k \sim \text{Bernoulli}(1 - \prod_{t' \leq t} p(z_{t'k} = 0|o_{\leq t'})\), which is sensible as \(M_k\) describes the presence of the \(k\)th feature. Then, because \(\Lambda_t\) is the sum of independent but non-identically distributed Bernoullis, Le Cam’s Theorem
According to the derivation, the recursion should hold exactly for the IBP prior in the absence of observations. We test this by checking whether the recursion correctly captures the sequence of marginal distributions by comparing the recursion’s analytical expressions to 5000 Monte Carlo samples, each of 50 customers, drawn using the IBP’s conditional distribution, for \( \alpha \in \{2.64, 30.91\} \times \beta \in \{3.11, 13.82\} \). First, we visually compared the analytical expressions versus the Monte Carlo estimates and found excellent agreement (Fig. 2).

Second, we computed the mean squared error between the analytical expression and Monte Carlo estimates as a function of the number of Monte Carlo samples. For all \((\alpha, \beta)\) values, the squared error falls as a power law with the number of samples (Fig 3), supporting our claim that the recursion is exact for the IBP prior.

For all settings of \((\alpha, \beta)\), the mean-squared error of the Monte Carlo estimate and our analytical expression falls as a power law with the number of samples.

![Figure 2](image2.png)

**Figure 2.** Two examples (top: \(\alpha = 2.64, \beta = 3.11\); bottom: \(\alpha = 30.91, \beta = 13.82\)) comparing Monte Carlo estimates of \(p(z_{tk} = 1)\) against our analytical expression.

![Figure 3](image3.png)

**Figure 3.** For all settings of \((\alpha, \beta)\), the mean-squared error of the Monte Carlo estimate and our analytical expression falls as a power law with the number of samples.
References


