UNIVERSITY OF CAMBRIDGE

Accelerated Gibbs Sampling for the Indian Buffet Process (and more!)

Finale Doshi-Velez and Zoubin Ghahramani, University of Cambridge

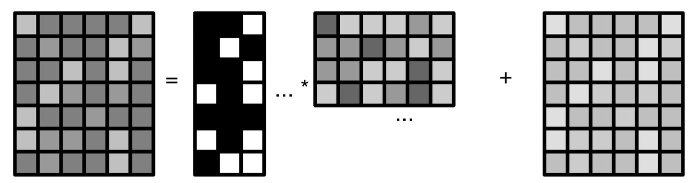
Abstract

We often seek to identify co-occurring hidden features in a set of observations. The Indian Buffet Process (IBP) provides a non-parametric prior on the features present in each observation, but current inference techniques for the IBP often scale poorly. The collapsed Gibbs sampler for the IBP has a running time cubic in the number of observations, and the uncollapsed Gibbs sampler, while linear, is often slow to mix. We present a new linear-time collapsed Gibbs sampler for conjugate likelihood models and demonstrate its efficacy on large real-world datasets.

More generally, our method, which maintains a posterior within the sampler to increase efficiency, is applicable to any bilinear model with a Gaussian likelihood (or other conjugate likelihood).

Bilinear Models

Bilinear models are common in machine learning.



X = UV + E

data = matrix product + error

Examples:

Factor Analysis	Probabilistic PCA	Probabilistic Matrix Factorization	Indian Buffet Process with linear likelihood
Y = LX + E	T = WX + E	X = UV + E	X = ZA + E

Suppose

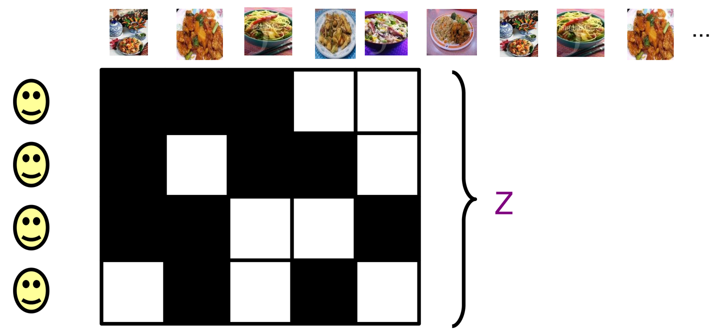
- We can compute P(X|Z) , but it's expensive
- We can compute P(A|X,Z)
- We cannot compute P(Z,A|X)

We develop a fast sampler for inference in these models.

The Indian Buffet Process

The Indian Buffet Process (IBP) is a non-parametric prior on binary matrices—useful as a general tool in latent feature models. The generative process proceeds as follows: Customers 1...N enter an "infinite buffet" one at a time. Customer n

- Samples a previously sampled dish based on its popularity.
- Samples Poisson(alpha / n) new dishes.

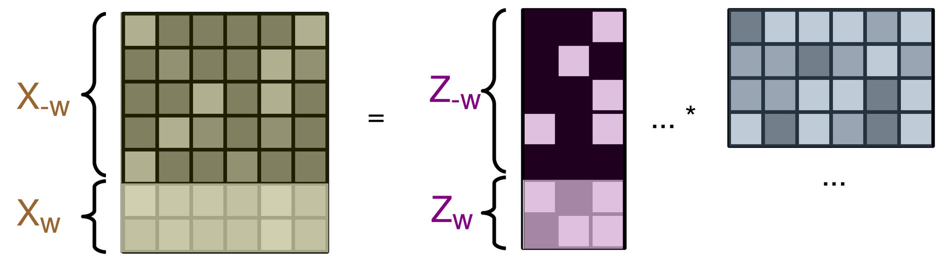


It has some nice properties:

- Observations are exchangeable.
- Infinite features, but finite datasets contain a finite number of features.

Windowing the Model

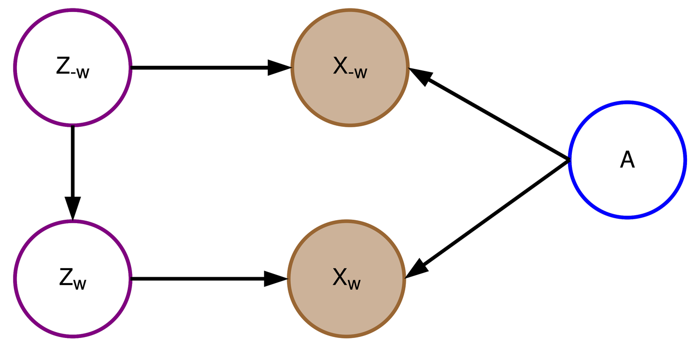
For large datasets, we do not want to look at all of the data at once. We consider doing (principled) inference on only a subset of the data. Note: this is not blocked sampling—we still only consider one element of Z at a time!



Gibbs Sampling

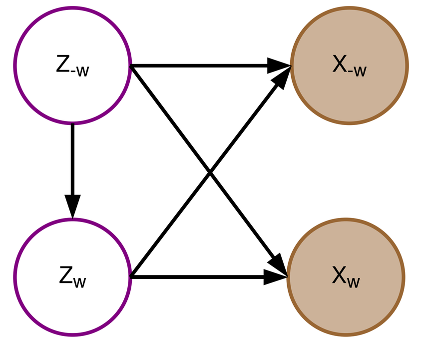
Uncollapsed Gibbs Sampling explicitly samples both Z and A (we experiment with a 'semi-collapsed' sampler which samples Z and A but integrates out new rows of A when considering whether to add a new feature).

- Advantage: Each iteration is fast to compute.
- Disadvantage: Often slow to mix.



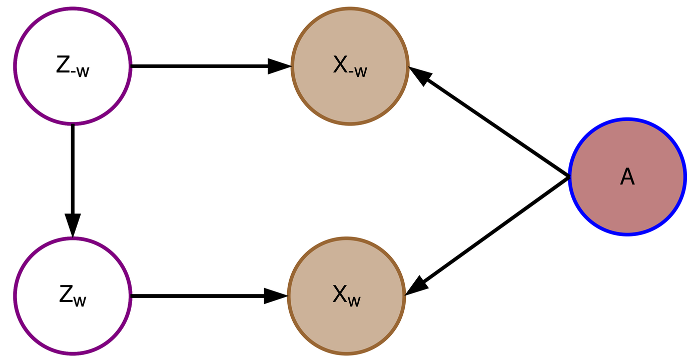
Collapsed Gibbs Sampling integrates out A, so only Z must be sampled.

- Advantage: Faster to mix.
- Disadvantage: Inference no longer scales!



Accelerated Sampling keeps a posterior on A, $P(A|Z_{-W},X_{-W})$ so that we may sample values in Z_W without knowing the values of X_{-W}. Once we have finished sampling within Z_W, the posterior is updated for sampling on a new window of observations.

- Mixes like the collapsed sampler.
- Runtime like an uncollapsed sampler.



Formal Derivation

Given a posterior $P(A | Z_{-n}, X_{-n})$, we can sample Z_n without looking at the data X_{-n}:

$$
\begin{aligned}
P(Z_{nk}=1|Z_{-nk},X) \propto\; & P(Z_{nk}=1|Z_{-nk})P(X|Z) \\
& P(Z_{nk}=1|Z_{-nk})\int_A P(X|Z,A)P(A)dA \\
& P(Z_{nk}=1|Z_{-nk})\int_A P(X_n|Z_n,A)P(X_{-n}|Z_{-n},A)P(A)dA \\
& P(Z_{nk}=1|Z_{-nk})\int_A P(X_n|Z_n,A)P(A|Z_{-n},X_{-n})dA
\end{aligned}
$$

Bayes Rule

Condition on and integrate out A

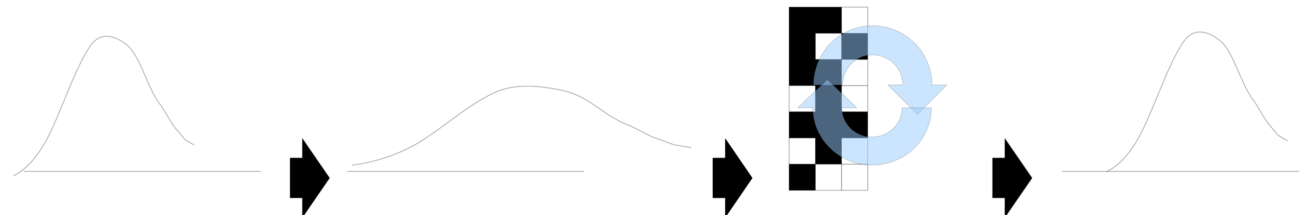
Observations independent given A , Z

Bayes Rule

We now have an **exact** method for computing $P(Z_{nk}|Z_{-nk},X)$ that depends only on X_n.

Algorithm

1. Initialise some Z, feature posterior
2. For each window of observations W



Get the feature posterior	Remove W's contribution to the posterior	Perform inference on Z_W	Reconstruct posterior with new Z_W

Key Consideration: How many observations should we consider at once?

- Depends on the cost of computing P(A|X,Z) and P(X|Z,A); for IBP with linear-Gaussian model, the optimal window is 1.
- However, considering larger groups implies fewer updates to P(A|Z,X) and slower loss of numerical precision.

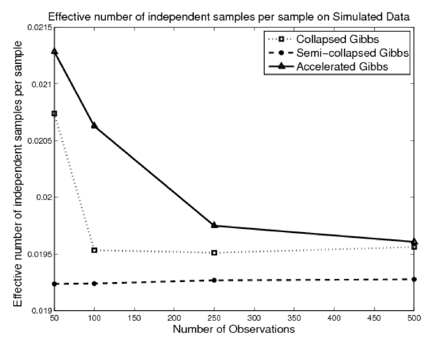
Experiments on Synthetic Data

Data was generated from the prior with

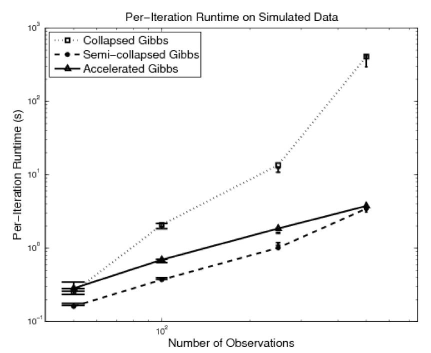
- D=10,
- N = {50,100,250, 500}.

We ran 5 chains for 1000 iterations to evaluate the mixing of each of the samplers.

Mixing was measured by the effective number of samples per sample. (Always less than one; measures how independent samples are.)



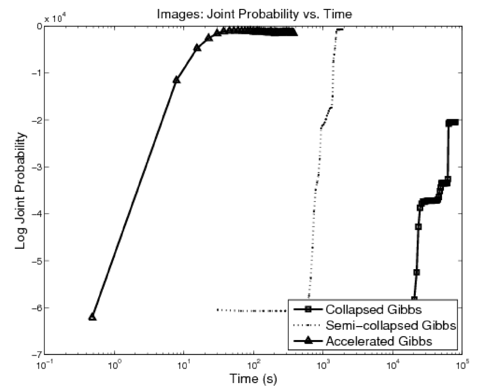
Mixing similar to collapsed sampler



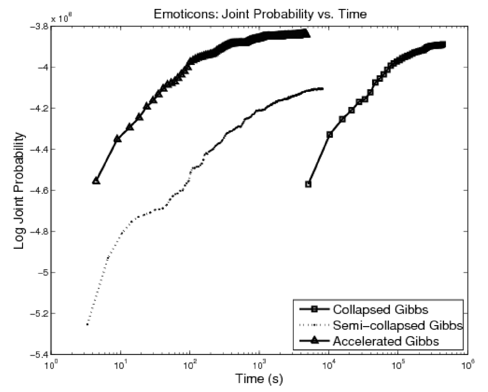
Runtime similar to semi-collapsed sampler

Experiments on Realworld Data

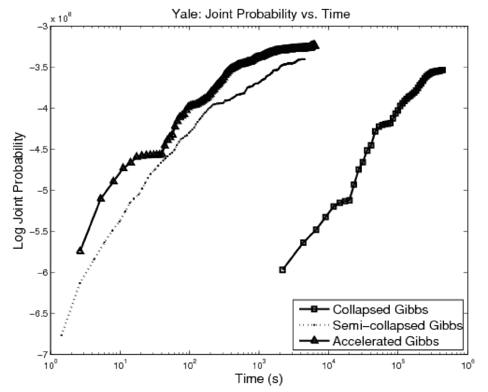
We applied the 3 samplers to several realworld data sets. The accelerated sampler achieved likelihoods similar to the collapsed sampler orders of magnitude faster.



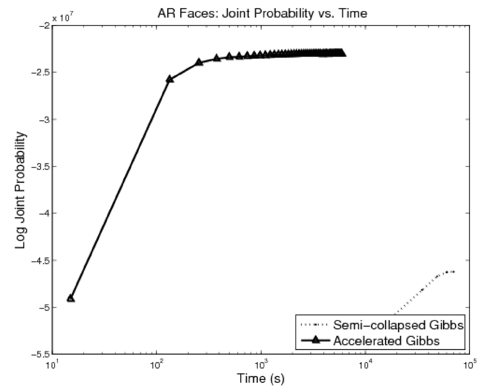
D=36, N = 1,000



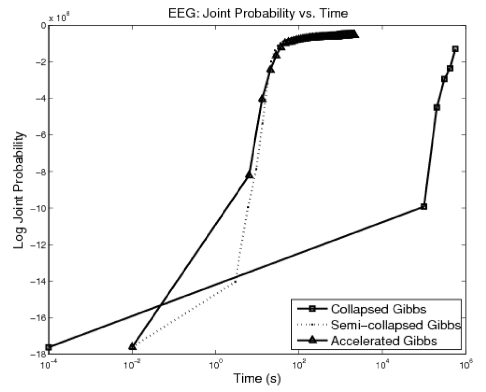
D=1,024, N = 702



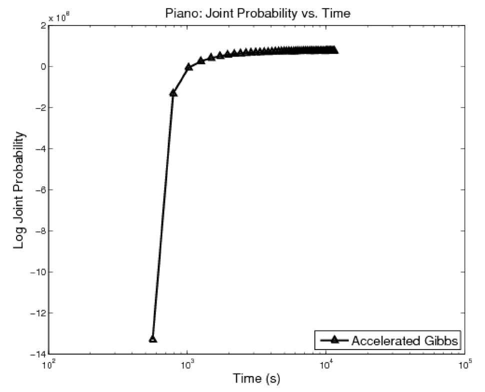
D=1,024, N = 722



D=1,598, N = 2,600



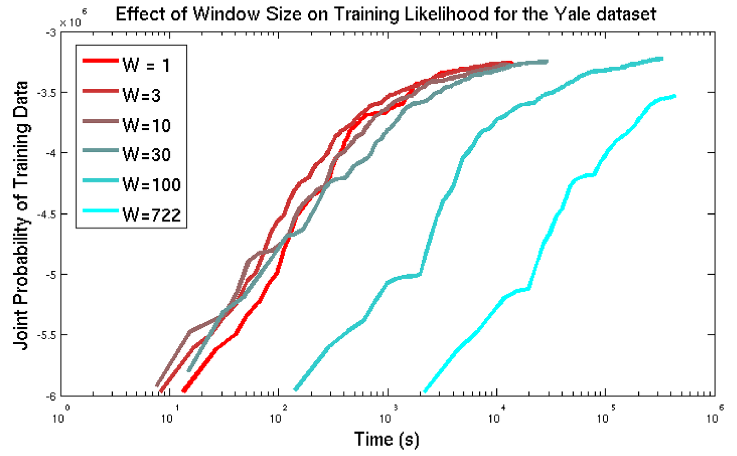
D=32, N = 4,400



D=161, N = 10,000

Effect of window size

From a series of tests on the Yale dataset, the window size has little effect on the performance. However, the larger windows take longer to process.



Conclusions

- Maintaining a posterior within a sampler allows us to perform fast inference in an important class of bilinear models
- In particular, our approach allows us to scale inference to large Indian Buffet Process models.

... code is available on my website!