Pitfalls in the use of Parallel Inference for the Dirichlet Process

Yarin Gal, Zoubin Ghahramani

University of Cambridge, UK

In Short:

- How can we do non-approximate parallel inference in the Dirichlet process?
- **Recent work** by Lovell, Adams, and Mansingka [2012] and Williamson, Dubey, and Xing [2013] **suggested a re-parametrisation** of the process to derive such inference.
- **We show that the approach suggested is impractical** due to an extremely unbalanced distribution of the data.
- We show that the suggested approach fails most requirements of parallel inference – **the load balance is independent of the size of the dataset and the number of nodes**.
- We end with suggestions of alternative paths of research.

Requirements of Distributed Samplers

Given a network with many nodes (computers in a network or cores in a cluster), we would like to have inference that:

- distributes the computational load evenly across the nodes,
- scales favourably with the number of nodes,
- has low overhead in the global steps,
- and converges to the true posterior distribution.

Parallel Inference in the DP

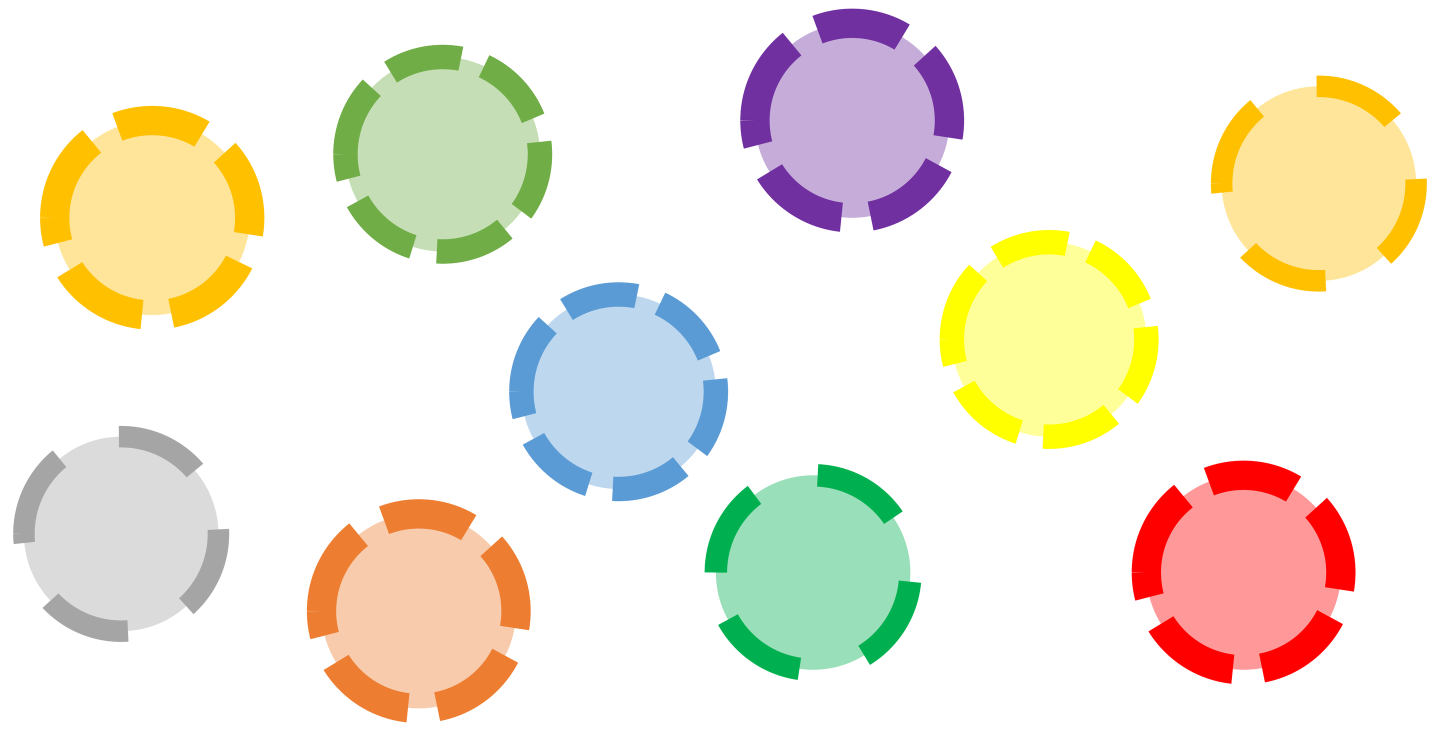
- **Two-staged Chinese restaurant process** was introduced in Lovell, Adams, and Mansingka [2012].
- Each data point (customer) chooses one of the K nodes (tables) according to its popularity:

$$P(\text{data point } n \text{ chooses node } k \mid \alpha) = \frac{\alpha\mu_k + \sum_{i=1}^{n-1}\mathbb{I}(s_{z_i} = k)}{\alpha + n - 1},$$

  for some vector of weights (μ_k) where s_{z_i} is the node allocation of point i.
- In each node k the data points follow the usual Chinese restaurant process (CRP) with parameter $\alpha\mu_k$.
- The resulting random partition has the same distribution as the CRP with parameter α as proved in [Williamson et al., 2013].
- Given many tables, the asymptotic number of tables (nodes) and their configuration drawn from the first stage of the process converges to that of a sample from a CRP with the same parameter.

The Pathology

Actual samples from a Dirichlet process with 50 data points **don't look like this**:



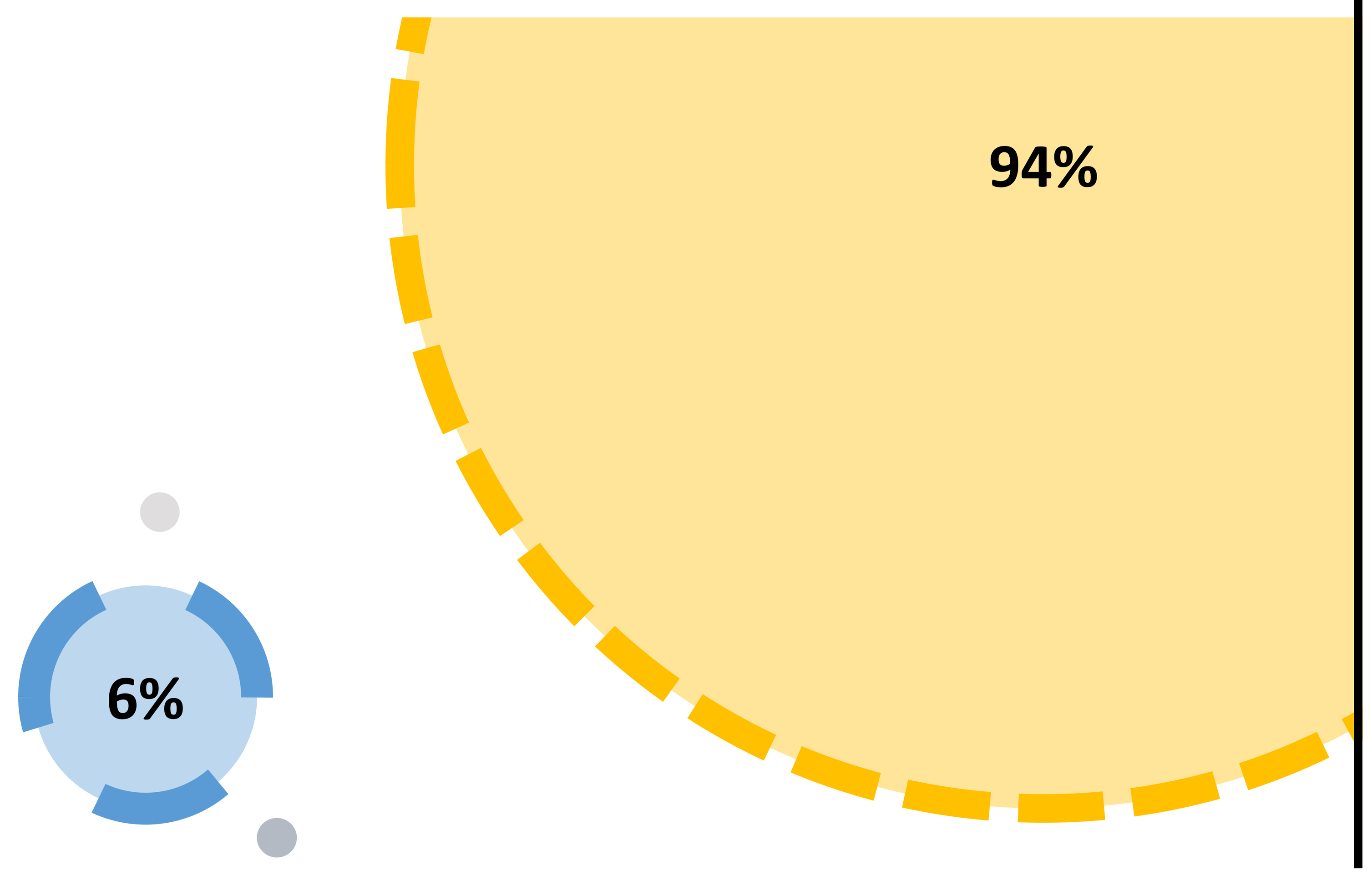
The expected number of tables in a restaurant with n customers is given by

$$\alpha \log(n)$$

and the sizes of the different tables follow an exponential decay, so the the number of customers sitting next to each table would actually be

$$C, \quad Cq, \quad Cq^2, \quad Cq^3, \quad \cdots$$

for $q = \dfrac{\alpha}{1+\alpha} < 1$ and $C = \dfrac{1}{1+\alpha}$. So for $n = 50$ data points and $\alpha = 0.1$ **the parallel inference would send 94% of the data to a single machine**, and an actual sample would be...



What Should We Do Instead?

Does there exist a setting of the inference which would give better load balance? What alternative approaches exist?

- **Optimal number of nodes.** Use the inference when *a small number of nodes is available*. $K = \lceil\alpha\rceil$ nodes in the network would result in uniform γ for example.
- **Optimal initialisation.** Initialise the sampler *close to the posterior* to have many evenly balanced clusters, to get a less distorted distribution of the load.
- **Metropolis–Hastings corrections.**
  - *Split the cluster representation among different nodes.*
  - A recent attempt is presented in Chang and Fisher III [2013].
  - Suitable for the case *when the posterior is known in advance and the initialisation can reflect that.*
  - However we suspect that by introducing additional random moves that depend on α in an inverse way this limitation might be overcome.

Future Research

- **Better approximate parallel inference.**
  - Current approach uses Gibbs sampling after distributing the data evenly across the nodes [Asuncion et al., 2008]. We synchronise their state only in the global step, leading the distribution to diverge from the true posterior.
  - Williamson et al. [2013] reported this to have slow convergence in practice.
  - *Can this approximate parallel inference be adjusted to have better mixing?*
- **Use distributions alternative to the Dirichlet process for clustering.**
  - Miller and Harrison [2013] showed that the Dirichlet process posterior is inconsistent in the number of cluster.
  - Suggested an alternative distribution for clustering: a Poisson mixture of Dirichlet distributions.
  - This might open the door for more efficient parallel inference.

References

Arthur U Asuncion, Padhraic Smyth, and Max Welling. Asynchronous distributed learning of topic models. In *Advances in Neural Information Processing Systems*, pages 81–88, 2008.

Jason Chang and John W Fisher III. Parallel sampling of DP mixture models using sub-cluster splits. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 620–628. 2013.

Dan Lovell, Ryan P Adams, and VK Mansingka. Parallel Markov chain Monte Carlo for Dirichlet process mixtures. In *Workshop on Big Learning, NIPS*, 2012.

Jeffrey W Miller and Matthew T Harrison. A simple example of Dirichlet process mixture inconsistency for the number of components. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 199–206. 2013.

Sinead Williamson, Avinava Dubey, and Eric P Xing. Parallel Markov Chain Monte Carlo for nonparametric mixture models. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 98–106, 2013.