Reparameterizing the Birkhoff Polytope for Variational Permutation Inference

Anonymous Authors
Anonymous Institutions

Abstract

How can we efficiently perform posterior inference over the space of permutations when there are $N!$ permutations of a set of N elements? Combinatorial optimization algorithms may enable efficient point estimation, but fully Bayesian inference poses a severe challenge in this high-dimensional, discrete space. We begin with a common maneuver: we relax the discrete set of permutation matrices—the vertices of the Birkhoff polytope—to the continuous set of doubly-stochastic matrices—the interior of the polytope. Our primary contribution is a pair of invertible and differentiable maps from densities on unconstrained space to densities on or near the Birkhoff polytope. These transformations are parameterized by a "temperature" that controls how concentrated the resulting density is at the extrema of the Birkhoff polytope; i.e. at permutation matrices. With these transformations, we perform variational inference over distributions on doubly stochastic matrices (and in the zero-temperature limit, on permutation matrices), leveraging reparameterization gradients to guide our optimization.

1 Introduction

Permutation inference is central to many modern machine learning problems. Identity management [Guibas, 2008] and multiple-object tracking [Shin et al., 2005, Kondor et al., 2007] are fundamentally concerned with finding a permutation that maps an observed set of items to a set of canonical labels. Ranking problems, critical to search and recommender systems, require inference over the space of item orderings [Meilă et al., 2007, Lebanon and Mao, 2008, Adams and Zemel, 2011]. Moreover, many probabilistic models, like preferential attachment network models [Bloem-Reddy and Orbanz, 2016] and repulsive point process models [Rao et al., 2016], incorporate a latent permutation into their generative processes; inference over model parameters requires integrating over the set of permutations that could have given rise to the observed data. In many of these settings, permutation inference is just one component of a larger estimation problem involving unknown model parameters and hierarchical structure.

The task of computing optimal point estimates of permutations under various loss functions has been well studied in the combinatorial optimization literature [Kuhn, 1955, Munkres, 1957, Lawler, 1963]. However, many probabilistic tasks require reasoning about uncertainty regarding permutation matrices. A variety of Bayesian permutation inference algorithms have been proposed, leveraging sampling methods [Diaconis, 1988, Miller et al., 2013, Harrison and Miller, 2013], Fourier representations [Kondor et al., 2007, Huang et al., 2009], as well as convex [Lim and Wright, 2014] and continuous [Plis et al., 2011] relaxations for approximating the posterior distribution. Here, we address this problem from an alternative direction, leveraging stochastic variational inference [Hoffman et al., 2013] and reparameterization gradients [Rezende et al., 2014, Kingma and Welling, 2014] to derive a scalable and efficient permutation inference algorithm.

Section 2 lays the necessary groundwork, introducing definitions, prior work on permutation inference, variational inference, and continuous relaxations. Section 3 presents our primary contribution: a pair of transformations that enable variational inference over doubly-stochastic matrices, and, in the zero-temperature limit, permutations, via stochastic variational inference. In the process, we show how these transformations connect to recent work on discrete variational inference [Maddison et al., 2016, Jang et al., 2016]. Sections 4 and 5 presents a variety of experiments that illustrate the benefits of the proposed variational approach.

Preliminary work. Under review by AISTATS 2018. Do not distribute.

2 Background

2.1 Definitions and notation.

A permutation is a bijective mapping of a set $\mathcal{X}$ onto itself. When $\mathcal{X} = \{x_1, \ldots, x_N\}$, this mapping is conveniently represented as a binary matrix $X \in \{0,1\}^{N \times N}$ where $X_{m,n} = 1$ implies that x_m is mapped to x_n. Since permutations are bijections, both the rows and columns of X must sum to one. From a geometric perspective, the Birkhoff-von Neumann theorem states that permutation matrices are vertices of the convex hull of doubly stochastic matrices; i.e. non-negative square matrices whose rows and columns sum to one. The set of doubly stochastic matrices is known as the *Birkhoff polytope*, and it is defined by,

$$\begin{aligned}\mathcal{B}_N = \Big\{ X : \quad & X_{m,n} \geq 0 \quad \forall\, m, n \in 1, \ldots, N; \\ & \sum_{n=1}^{N} X_{m,n} = 1 \quad \forall\, m \in 1, \ldots, N; \\ & \sum_{m=1}^{N} X_{m,n} = 1 \quad \forall\, n \in 1, \ldots, N \Big\}.\end{aligned}$$

These linear row- and column-normalization constraints restrict $\mathcal{B}_N$ to a $(N-1)^2$ dimensional subset of $\mathbb{R}^{N \times N}$. Despite these constraints, we have a number of efficient algorithms for working with these objects. The *Sinkhorn-Knopp algorithm* [Sinkhorn and Knopp, 1967] projects the positive orthant onto $\mathcal{B}_N$ by iteratively normalizing the rows and columns, and the *Hungarian algorithm* [Kuhn, 1955, Munkres, 1957] solves the minimum weight bipartite matching problem—optimizing a linear objective over the set of permutation matrices—in cubic time.

2.2 Variational inference and the reparameterization trick

Variational Bayesian inference algorithms aim to approximate the posterior distribution $p(x \,|\, y)$ with a more tractable distribution $q(x; \theta)$, where "tractable" means that, at a minimum, we can sample q and evaluate it pointwise (including its normalization constant). We find this approximate distribution by searching for the parameters θ that minimize the Kullback-Leibler (KL) divergence between q and the true posterior, or equivalently, maximize the evidence lower bound (ELBO),

$$\mathcal{L}(\theta) \triangleq \mathbb{E}_q \left[\log p(x, y) - \log q(x; \theta) \right].$$

Perhaps the simplest method of optimizing the ELBO is stochastic gradient ascent. However, computing $\nabla_\theta \mathcal{L}(\theta)$ requires some care since the ELBO contains an expectation with respect to a distribution that depends on these parameters.

When x is a continuous random variable, we can often go one step further and leverage the *reparameterization trick* [Salimans and Knowles, 2013, Kingma and Welling, 2014]. Specifically, in some cases we can simulate from q via the following equivalence,

$$x \sim q(x; \theta) \qquad \Longleftrightarrow \qquad \xi \sim r(\xi), \quad x = g(\xi; \theta),$$

where r is a distribution on the "noise" ξ and where $g(\xi; \theta)$ is a deterministic and differentiable function. For example, if $q(x; \theta) = \mathcal{N}(x \,|\, \theta, 1)$, we can reparameterize by setting the noise distribution to $r(\xi) = \mathcal{N}(\xi \,|\, 0, 1)$ and using the transformation $g(\xi; \theta) = \xi + \theta$. The reparameterization trick effectively "factors out" the randomness of q. With this transformation, we can bring the gradient inside the expectation as follows,

$$\begin{aligned}\nabla_\theta \mathcal{L}(\theta) = \mathbb{E}_{r(\xi)} \Big[& \nabla_\theta \log p(g(\xi; \theta) \,|\, y) \\ & - \nabla_\theta \log q(g(\xi; \theta); \theta) \Big]. \quad (1)\end{aligned}$$

This gradient can be estimated with Monte Carlo, and, in practice, this leads to lower variance estimates of the gradient than, for example, the score function estimator [Williams, 1992, Glynn, 1990].

2.3 Related Work

A number of previous works have considered approximate methods of posterior inference over the space of permutations.

When a point estimate will not suffice, sampling methods like Markov chain Monte Carlo (MCMC) algorithms may yield a reasonable approximate posterior for simple problems [Diaconis, 1988]. Harrison and Miller [2013] developed an importance sampling algorithm that fills in count matrices one row at a time, showing promising results for matrices with $O(100)$ rows and columns. It may also be possible to turn the Hungarian algorithm into an efficient sampling algorithms using Perturb-and-MAP [Li et al., 2013]. Another line of work considers inference in the spectral domain, approximating distributions over permutations with the low frequency Fourier components [Kondor et al., 2007, Huang et al., 2009]. Perhaps most relevant to this work, Plis et al. [2011] propose a continuous relaxation from permutation matrices to points on a hypersphere, and then use the von Mises-Fisher (vMF) distribution to model distributions on the sphere's surface. While the vMF distribution does have a concentration parameter, as the concentration goes to infinity, the distribution converges to a point on the sphere. By contrast, we will derive temperature-controlled densities over points inside or near the Birkhoff polytope such that as the temperature goes to zero, the distribution converges to an atomic density on permutation matrices.

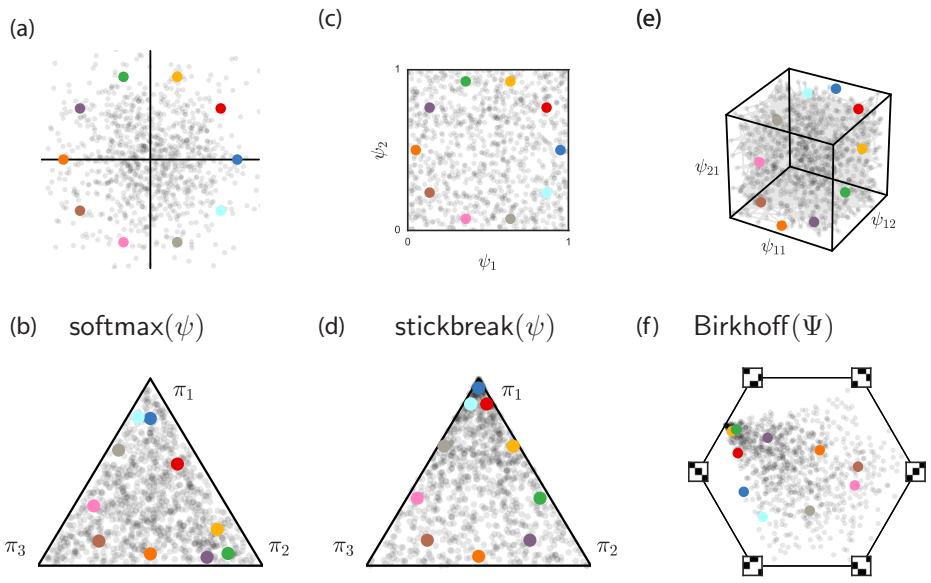

Figure 1: Reparameterizations of discrete polytopes. (a,b) The Gumbel-softmax, or "Concrete" transformation maps points $\psi \in \mathbb{R}^N$ to points $x \in \Delta_N$ by adding noise and applying the softmax. Here we show a slice for $N = 3$ with $\psi_3 = 0$. Colored points are aids to visualize the transformation. (c,d) Stick-breaking offers and alternative transformation, here from points $\psi \in [0, 1]^{N-1}$ to Δ_N. The ordering of the stick-breaking induces an asymmetry in the transformation. (e,f) We extend this stick-breaking transformation to reparameterize the Birkhoff polytope, i.e. the set of doubly stochastic matrices. Here, $\mathcal{B}_3$ is reparameterized in terms of matrices $\Psi \in [0, 1]^{2\times 2}$, of which three coordinates are shown in (e). These points are mapped to doubly stochastic matrices, which we have projected onto $\mathbb{R}^2$ in panel (f).

3 Variational permutation inference via reparameterization

Unfortunately, reparameterization-based variational inference 2.2 is not readily available for permutations: as samples (permutations) are discrete, the function g cannot be continuous. However, recently, there have been a number of proposals to extend the reparameterization trick to discrete inference problems via continuous relaxation [Maddison et al., 2016, Jang et al., 2016, Kusner and Hernández-Lobato, 2016]. Essentially, the method is based on replacing the original ELBO objective (1) by a surrogate one, arising from replacing the original samples x by a relaxation; that is, now assuming they can belong to a larger, continuous set where derivatives are well defined. This surrogate construction is controlled by a "temperature" knob that dictates the degree of relaxation; how "far" is the relaxation from the original discrete problem.

These relaxations, however, are not suitable for permutations, as the representation of a permutation as a category requires $N!$ slots. In this section, we develop two relaxations to enable variational permutation inference. In both relaxations the Birkhoff polytope will play a fundamental role, although clear differences exist. Also, in either case we start with a noise distribution Ξ (here, gaussian) and transform it to a "relaxed" permutation X using a differentiable and invertible mapping G, and parameters θ; i.e. $X = G(X_i; \theta)$. With a suitable choice of priors this is enough to compute all terms of equation (1). We refer the reader to the supplemental material for details on choice of the (continuous) priors, and on how our assumptions lead to explicit expressions for the ELBO, for both relaxations introduced in the following.

3.1 Stick-breaking transformations of the Birkhoff polytope

Let Ψ be an arbitrary matrix in $[0, 1]^{(N-1)\times(N-1)}$; we will transform it into a doubly stochastic matrix, $X \in [0, 1]^{N\times N}$ by filling in entry by entry, starting in the top left and raster scanning left to right then top to bottom. Denote the (m, n)-th entries of Ψ and X by ψ_{mn} and x_{mn}, respectively.

Each row and column has an associated unit-length "stick" that we allot to its entries. The first entry in the matrix is given by, $x_{11} = \psi_{11}$. As we work left to right in the first row, the remaining stick length decreases as we add new entries. This reflects the row normalization constraints. Formally, the stick-breaking

transformation for the first row is given by,

$$x_{1n} = \psi_{1n}\left(1 - \sum_{k=1}^{n-1} x_{1k}\right) \quad \text{for } n = 2, \ldots, N-1$$
$$x_{1N} = 1 - \sum_{n=1}^{N-1} x_{1n}.$$

However, the remaining rows must now conform to both row- and column-constraints. That is,

$$x_{mn} \leq 1 - \sum_{k=1}^{n-1} x_{mk} \qquad \text{(row sum)}$$
$$x_{mn} \leq 1 - \sum_{k=1}^{m-1} x_{kn} \qquad \text{(column sum)}.$$

Moreover, there is also a lower bound on x_{mn}. This entry must claim enough of the stick such that what is leftover fits within the confines imposed by subsequent column sums. That is, each column sum places an upper bound on the amount that may be attributed to any subsequent entry. If the remaining stick exceeds the sum of these upper bounds, the matrix will not be doubly stochastic. Thus,

$$\underbrace{1 - \sum_{k=1}^{n} x_{mk}}_{\text{remaining stick}} \leq \underbrace{\sum_{j=n+1}^{N} \left(1 - \sum_{k=1}^{m-1} x_{kj}\right)}_{\text{remaining upper bounds}}.$$

Rearranging terms, we have,

$$x_{mn} \geq 1 - N + n - \sum_{k=1}^{n-1} x_{mk} + \sum_{k=1}^{m-1} \sum_{j=n+1}^{N} x_{kj}.$$

Of course, this bound is only relevant if the right hand side is greater than zero. Taken together, we have $\ell_{mn} \leq x_{mn} \leq u_{mn}$, where,

$$\ell_{mn} \triangleq \max\left\{0,\, 1 - N + n - \sum_{k=1}^{n-1} x_{mk} + \sum_{k=1}^{m-1} \sum_{j=n+1}^{N} x_{kj}\right\}$$
$$u_{mn} \triangleq \min\left\{1 - \sum_{k=1}^{n-1} x_{mk},\, 1 - \sum_{k=1}^{m-1} x_{kn}\right\}.$$

Accordingly, we define, $x_{mn} = \ell_{mn} + \psi_{mn}(u_{mn} - \ell_{mn})$. The inverse transformation from X to Ψ is analogous. We start by computing ψ_{11} and then progressively compute upper and lower bounds and set $\psi_{mn} = (x_{mn} - \ell_{mn})/(u_{mn} - \ell_{mn})$.

To complete the reparameterization, we define a parametric, temperature-controlled density for Ψ. Let $\Xi \in \mathbb{R}^{(N-1)\times(N-1)}$ be a matrix of standard Gaussian random variables. We define,

$$\psi_{mn} = \sigma\left(\frac{\mu_{mn} + \eta_{mn}\Xi_{mn}}{\tau}\right),$$

where $\theta = \{\mu_{mn}, \eta_{mn}^2\}_{m,n=1}^{N}$ are the mean and variance parameters of the mapping, $\sigma(u) = (1 + e^{-u})^{-1}$ is the logistic function, and τ is a temperature parameter. As $\tau \to 0$, the values of ψ_{mn} are pushed to either zero or one, depending on whether the input to the logistic function is negative or positive, respectively. As a result, the doubly-stochastic output matrix X is pushed toward the extreme points of the Birkhoff polytope, the permutation matrices.

3.2 Rounding toward permutation matrices

While relaxing permutations to the Birkhoff polytope is intuitively appealing, it is not strictly required. For example, consider the following procedure for sampling a point *near* the Birkhoff polytope:

(i) Input a point $M \in \mathbb{R}_{+}^{N\times N}$;

(ii) Project M onto the Birkhoff polytope (approximately) using the Sinkhorn-Knopp algorithm;

(iii) Sample a Gaussian random variable Ψ with mean $\text{proj}(M)$ and variance Σ;

(iv) Round Ψ to the nearest permutation matrix, $P^*(\Psi)$, using the Hungarian algorithm; and

(v) Return $X = \tau\Psi + (1-\tau)P^*(\Psi)$.

This procedure implicitly defines a distribution over matrices X parameterized by M and Σ, which we will optimize. Steps (i) and (ii) involve differentiable transformations of parameter M to set the mean close to the Birkhoff polytope; the challenge in computing the density $p(X; M, \Sigma)$ stems from step (iv), since the rounding operation is not differentiable. However, the rounding output only changes at points that are equidistant from two or more permutation matrices. In other words, rounding is a piecewise constant function of Ψ with discontinuities only at a set of points with zero measure. In practice, we find that we can safely ignore these discontinuities. Furthermore, note that $P^*(\Psi) \equiv P^*(X)$ so that the inverse transformation is $\Psi = \frac{1}{\tau}X - \frac{1-\tau}{\tau}P^*(X)$. Taken together, X is a linear function of a Gaussian random variable and its density is,

$$p(X; M, \Sigma) = \frac{1}{\tau}\mathcal{N}\left(\frac{1}{\tau}X - \frac{1-\tau}{\tau}P^*(X);\, \text{proj}(M), \Sigma\right),$$

which is valid for X in the image of the rounding operation. In the zero-temperature limit we recover a discrete distribution on permutation matrices; otherwise the density concentrates near the vertices as $\tau \to 0$.

Why is step (iii) necessary? Naïvely, we could sample a matrix and project it with the Sinkhorn-Knopp algorithm; the problem is that we could not calculate the

resulting density since the Sinkhorn-Knopp algorithm is non-invertible. With the algorithm above, Sinkhorn-Knopp is employed *prior* to sampling, allowing us to compute the resulting density.

3.3 Theoretical considerations

Stick-breaking and rounding each have their strengths and weaknesses. Here we list some of their conceptual differences, and in Section 4 we evaluate the two approaches empirically.

- Stick-breaking relaxes to $\mathcal{B}_N$ whereas rounding relaxes to $\mathbb{R}^{N\times N}$. The Birkhoff polytope is perhaps intuitively nicer, but as long as the log probability accepts real-valued matrices, either may suffice.
- Rounding uses the $O(N^3)$ Hungarian algorithm in its sampling process, whereas stick-breaking is a feed-forward, $O(N^2)$, process.

cobian $O(N^4)$ in

- Stick-breaking is a bijective mapping from $\mathbb{R}^{(N-1)\times(N-1)}$ to $\mathcal{B}_N$, whereas rounding maps $\mathbb{R}^{N\times N}$ to a subset of $\mathbb{R}^{N\times N}$. Consider a simple example of rounding in the one-dimensional simplex, that is, the unit interval. If $\tau = 0.5$, the rounding operation maps $[0, 1]$ to $[0, 0.25) \cup [0.75, 1]$ (the midpoint is arbitrarily rounded up); the resulting density has zero measure in the interval $[0.25, 0.75)$. The same is true of rounding toward permutations: the inverse mapping is only defined for points within τ of a permutation, and the resulting density is discontinuous at the boundaries.
- Rounding can easily incorporate constraints. If certain mappings are invalid, e.g. $X_{mn} \equiv 0$, they are given an infinite cost in the Hungarian algorithm.[1] This is hard to do this with stick breaking as it would change the computation of the upper and lower bounds.
- Stick-breaking introduces a dependence on ordering. While the mapping is bijective, a desired distribution on the Birkhoff polytope may require a complex distribution for Ψ. Rounding, by contrast, is more "symmetric" in this regard.
- Unfortunately, neither admits a simple probability mass function on permutations, in contrast to the Gumbel-softmax trick [Maddison et al., 2016, Jang et al., 2016]. This complicates the computation of the ELBO, as we discuss next.

[1] Constraints of the form $X_{m,n} \equiv 1$ simply reduce the dimension of the inference problem.

Table 1: Summary of results in VAE

Method	$-\log p(x)$
Gumbel-Softmax	106.7
Concrete	111.5
Rounding	121.1
Stick-breaking	119. 8

4 Synthetic Experiments

We are interested in two principal questions: (i) how well can the stick-breaking and rounding reparameterizations of the Birkhoff polytope approximate the true posterior distribution over permutations in tractable, low-dimensional cases? and (ii) when, if ever, do our proposed continuous relaxations offer advantages over alternative Bayesian permutation inference algorithms?

Before addressing those questions we start by comparing how the categorical counterparts [2] of our proposed distributions over permutations perform on a simple VAE task. Results of this task may shed light on the usefulness of our proposed relaxations.

4.1 Variational Autoencoders (VAE) with categorical latent variables

We considered the density estimation task on MNIST digits, as in Maddison et al. [2016], Jang et al. [2016], where observed digits are reconstructed from a latent discrete code. We used the continuous ELBO for training, and evaluated performance based on the marginal likelihood, estimated through the multi-sample variational objective of the discretized model. We compared against the methods of Jang et al. [2016], Maddison et al. [2016], finding similar (although slightly worse) results (Table 1). This difference may be intepreted as the price to be paid in order to enable an extension of a relaxed distribution over categories, to permutations. In the supplement more results on this task are available.

4.2 Synthetic matching experiments

To assess the quality of our approximations for distributions over permutations, we considered a toy matching problem in which we are given the locations of N cluster centers and a corresponding set of N observations, one for each cluster, corrupted by Gaussian noise. Moreover, the observations are permuted so there is no correspondence between the order of observations and the order of the cluster centers. The goal is to

[2] That is, simple stick breaking and rounding in the probability simplex.

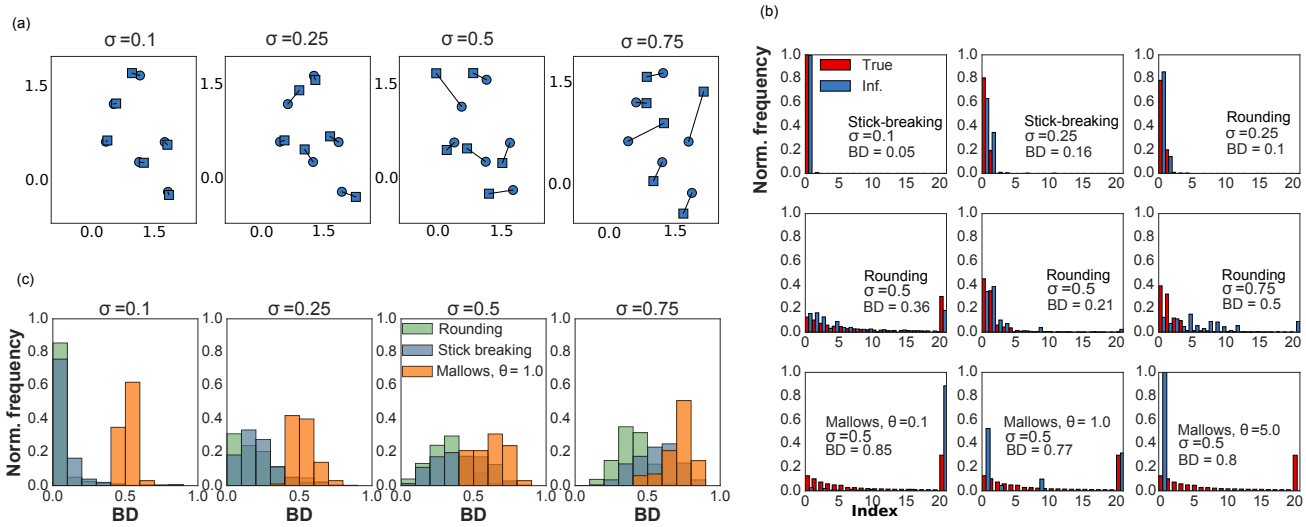

Figure 2: Synthetic matching experiment results. The goal is to infer the lines that match squares to circles. (a) Examples of center locations (circles) and noisy samples (squares), at different noise variances. (b) For illustration, histograms of the true and inferred posterior distribution of identities along the corresponding BD, for selected cases. Histogram indexes are sorted from the highest to lowest actual posterior probability. Only the 20 most likely configurations are shown, and the 21st bar collapses the mass of all remaining configurations. (c) Population results (histograms) across 200 experiment repetitions of each parameter configuration.

recover the posterior distribution over permutations. For $N = 6$, we can explicitly enumerate the $N! = 720$ permutations and compute the posterior exactly.

As a baseline, we consider the Mallows distribution Mallows [1957] with density over a permutations ϕ given by $p_{\theta,\phi_0}(\phi) \propto \exp(-\theta d(\phi, \phi_0))$, where ϕ_0 is a central permutation, d is a distance between permutations [3] and θ controls the spread around ϕ_0. This is perhaps the most popular exponential family model for permutations; however, it is too simple and might fail to capture complex features of distributions.

Table 2: Mean BDs in the synthetic matching experiment for various methods and observation variances.

	Variance σ^2			
Method	$.1^2$	$.25^2$	$.5^2$	$.75^2$
Stick-breaking	.09	.23	.41	.55
Rounding	**.06**	**.21**	**.32**	**.38**
Mallows ($\theta = 0.1$)	.93	.92	.89	.85
Mallows ($\theta = 0.5$)	.51	.53	.61	.71
Mallows ($\theta = 2$)	.23	.33	.53	.69
Mallows ($\theta = 5$)	.08	.27	.54	.72
Mallows ($\theta = 10$)	.08	.27	.54	.72

Using the Battacharya distance (BD) we measured the discrepancy between true posterior and an empirical estimate of the inferred posteriors: in our relaxations, by sampling from $q(X; \theta)$ and 'rounding' to the nearest permutation using the Hungarian algorithm. Likewise, for the Mallows distribution, we set ϕ_0 to the MAP estimate, also through the Hungarian algorithm, and sampled using MCMC.

We found our method outperforms the simple Mallows distribution, suggesting it might reasonably approximate non-trivial distributions over permutations. Fig 2 illustrates our findings by showing sample experiment configurations (a), examples of inferred posteriors (b) and distribution of BD's (c). These histograms are summarized by Table 2.

5 Inferring neuron identities in *C. elegans*

Finally, we consider an application motivated by the study of the neural dynamics in the *Caenorhabditis elegans* (C.elegans) Kato et al. [2015], a nematode (worm) of interest for neuroscience, as its neural network changes little from animal to animal. Recent efforts have focused on establishing an accurate and complete neural wiring diagram from anatomical data [Varshney et al., 2011] — a connectome — that we represent as graph whose nodes are neurons (there are 278 somatic neurons for the hermaphrodyte C. elegans) and whose edges are synapses. Fig 3a shows the corresponding adjacency matrix, that we name $\mathcal{C}$.

The C. elegans, then, is particularly suited from inves-

[3] Here, $d(\phi, \phi_0) = \sum_{i=1}^{N} |\phi(i) - \phi_0(i)|$.

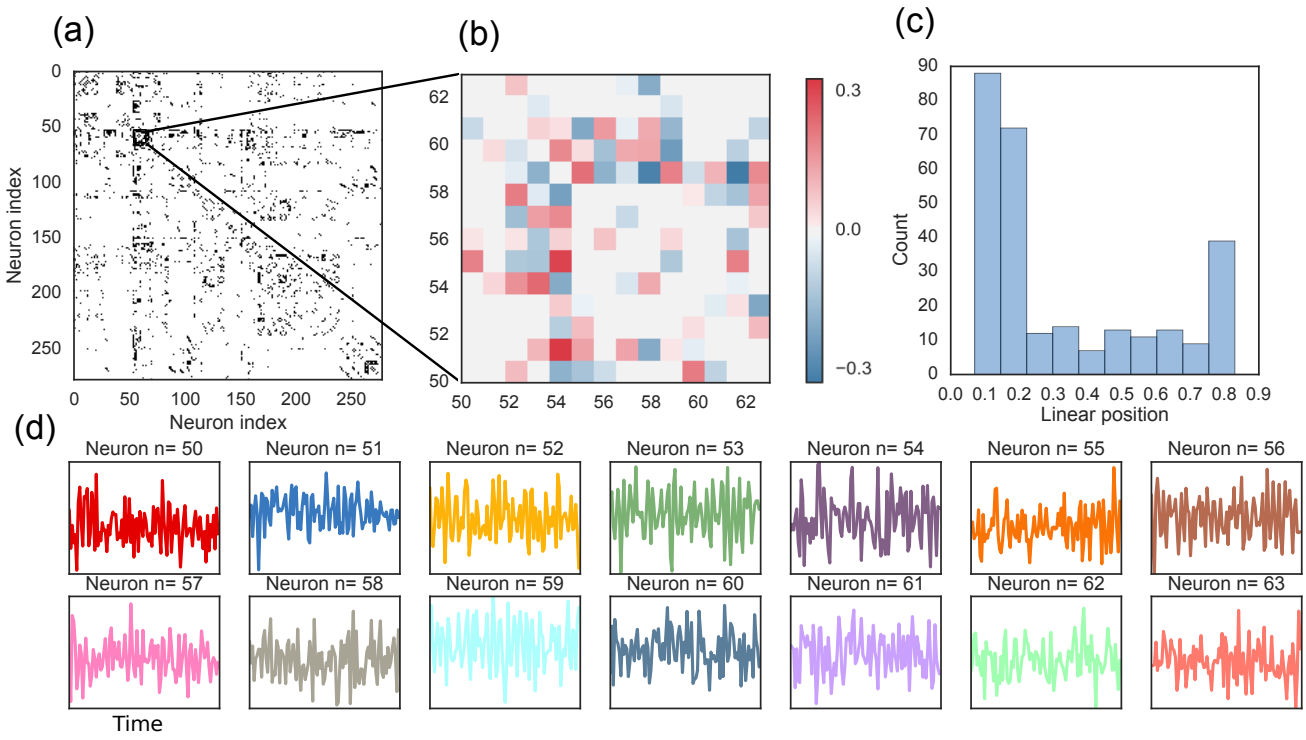

Figure 3: Problem setup. (a) Hermaphrodite C.elegans reference connectome (from Varshney et al. [2011], Lints et al. [2005]) consisting of 278 somatic neurons, merging two distinct types of synapses: chemical and electrical (gap junctions). (b) Example of matrix W consistent with the connectome information (only 14 neurons for visibility), (c) Distribution of neuron position in the body, zero means head and one means tail. From White et al. [1986], Lints et al. [2005] (d). Examples of the dynamical system sampled from matrix W.

tigating how neural activity gives rise to behaviour, a question that has been recently rigorously addressed Kato et al. [2015]. However, there, intensive manual data curation was needed to match neural recordings from calcium imaging techniques to actual neurons. This manual analysis was based on the study of joint patterns of neural activity, and the comparison of observed linear position of recorded neurons to a reference worm. Unfortunately, in some cases, identity could not be exactly resolved, and only putative candidates were inferred.

This difficulty offers fertile ground for the development of new methods. Recently, promising approaches Aoki et al. [2017] have illustrated the plausibility of using the Brainbow technology Livet et al. [2007] for such purposes, by genetically engineering worms to express fluorescent proteins.

We prototype an alternative solution that bypasses the need for such sophisticated genetic engineering. Our method embodies the criteria of manual data curation into an algorithm: we resolve neural identity by integrating different sources of information from the connectome, some covariates (e.g. position) and neural dynamics. Moreover, we combine information from many individuals to facilitate identity resolution in hard cases.

5.1 Probabilistic Model

We consider $n = 1, \dots, M$ linear (for simplicity) dynamical systems recorded during $t = 1, \dots, T$ time-steps $Y_t^m = P_m W P_m^\top Y_{t-1}^m + \varepsilon_t$, $\varepsilon_t \sim \mathcal{N}(0, I)$ (Fig 3d). Each of the Y_t^m is a $N = 278$ dimensional vector representing the recorded activity of the entire nervous system. These recordings are a permutation (represented by P_n) of the dynamics in a canonical order. Entries of W[4] are chosen consistently with the connectome: i.e., $W_{i,j} = 0$ if $\mathcal{C}_{i,j} = 0$. The remaining non-zero entries are then independently sampled from a normal distribution, and scaled by a factor of the spectral radius to ensure stability (see Fig 3b for an example of W, and see supplement for further details).

We perform variational inference on this model for the joint estimation of the posterior probability of P_m and W given Y_m [5]. For W we use a gaussian

[4] Alternatively, one could have chosen a hierarchical model of $W_m \sim p(W)$, a direction that we avoided here for the sake of simplicity.

[5] ε is assumed known for simplicity, but could otherwise

prior $p(W) \sim \mathcal{N}(0, I)$. Also, for each of the P_m use the machinery developed in section 3, and we focus on the rounding approximation. Then, we approximate the true posterior $p(W, P_m|Y) \propto p(Y|W, P_m) \times p(W) \prod_{m=1}^{M} p(P_m)$ by a variational family q of the form $q(W, P_m) \equiv q(W) \prod_m^M q(P_m)$, where $q(W)$ is also gaussian and $q(P_m)$ has the distribution described in 3.2.

Finally, we use neural position along the worm's body to constrain the number of possible neural identities for a given neuron: specifically, we utilize previously documented positions of each neuron as numbers between zero and one White et al. [1986], Lints et al. [2005] (under the abstraction that a worm can be represented as one-dimensional object, see Fig 3c). Then, given reported positions of all (or some) neurons, we can conceive a binary *confusion* matrix D^m so that $D^m_{i,j} = 1$ if (observed) neuron i is close enough to (canonical) neuron j; i.e., if their distance is smaller than a tolerance ν. We can enforce this constrain during inference, by zeroing corresponding entries in the parameter matrix M described in 3.2. This modeling choice greatly reduces the number parameters of the model, and facilitates inference. Also, we allow for a certain number of neural identities to be known beforehand, easily encoded in D^m as well.

5.2 Results

We compared against three methods: i) naive variational inference, where we don't enforce the constraint that P is a permutation but allow many neurons to be mapped to the same one, ii) MCMC, where one alternates between sampling from the conditionals of W (gaussian) and P_m, from which one can sample by proposing local swipes, as described in Diaconis [2009], and iii) MAP estimator, which can be understood as a 'hard' version of ii); instead of iteratively sampling, we alternate between the MAP estimate of W (a ridge regression-like expression) and the MAP of the P_m's. For the P_m's we notice the objective is a quadratic assignment problem (QAP) in P_m, that is, it can be expressed as $Trace(APBP^\top)$ for some matrices A, B. We used the QAP solver proposed in Vogelstein et al. [2015].

As shown in Fig 4, our method outperforms each baseline: specifically, first, Fig 4a illustrates convergence to a better solution for a certain parameter configuration. More conclusively, Fig 4b and Fig 4c shows that our method outperforms alternatives when there is much uncertainty on neural position; i.e, when there are many possible candidates (large ν), and where only a small proportion of neurons are known with certitude. Fig 4c also shows that we indeed obtain benefits from combining information of many worms (although the same applies to MCMC).

Altogether, these results indicate our method enables a more efficient use of information than MCMC. We interpret this in terms of the observation that in practice variational inference converges faster than MCMC Blei et al. [2017]: we expect MCMC would eventually converge, but in practice it does so slow that asymptotic values are not achieved in reasonable time, as changes in P are very local and proposals (local swipes) are usually rejected. On the other hand, our parameterization allows to more freely sample the parameter space. This is observed in Fig 4d: variability of permutation samples is high during iterations (see variability in Fig 4e), but eventually decays to asymptotic values after more certainty of the true permutation has been accumulated.

6 Discussion

Our results provide evidence that permutation variational inference might provide a helpful tool for the inference of neural identity, as it allows to properly represent shared information across animals, and different degrees of certainty based on covariates. In order to apply it to real data it is necessary to consider more realistic models of neural dynamics, which are nonlinear but might be well characterized, for example, by a set of atomic low-dimensional linear dynamical systems, each of one corresponding to a certain behavioral state Kato et al. [2015]. The methodology developed in Linderman et al. [2016] seems particularly suitable to harness that increased level of complexity. 2.3 3.5 11

be included in the posterior, or be directly estimated from data.

References

R. P. Adams and R. S. Zemel. Ranking via sinkhorn propagation. *arXiv preprint arXiv:1106.1925*, 2011.

W. Aoki, H. Matsukura, Y. Yamauchi, H. Yokoyama, K. Hasegawa, R. Shinya, and M. Ueda. Cellomics approach for high-throughput functional annotation of caenorhabditis elegans neural network. *bioRxiv*, 2017. doi: 10.1101/182923.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, (just-accepted), 2017.

B. Bloem-Reddy and P. Orbanz. Random walk models of network formation and sequential Monte

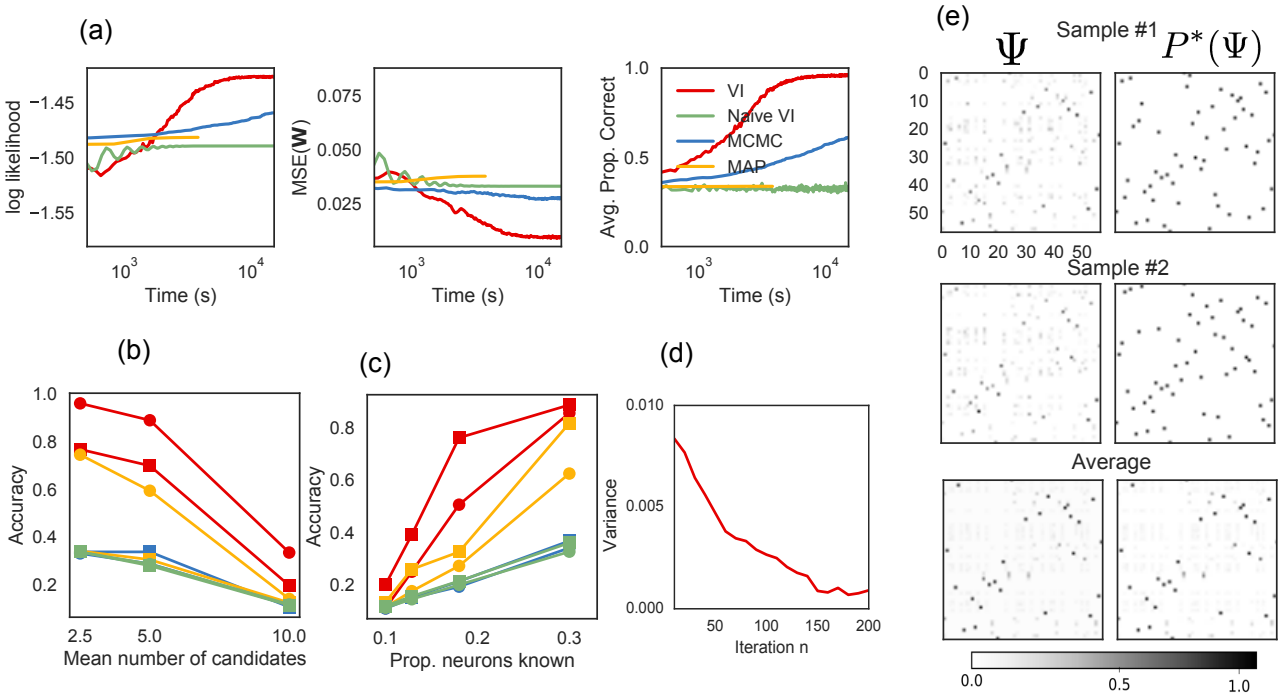

Figure 4: Results on the C.elegans inference example. (a) An example of convergence of the algorithm, and the baselines. (b) Accuracy of identity inference as a function of mean number of candidates (correlated with ν), for $M = 1$ worm (square) and combining information of $M = 5$ worms (circles). (c) Accuracy as a function of the proportion of known networks beforehand, with $\nu = 0.2$ (circles) and $\nu = 0.1$ (squares). (d)Variance of distribution over permutations (vectorized) as a function of the number of iterations. (e) Two samples of permutation matrices $P^*(\Psi)$ (right) and their noisy, non-rounded versions Ψ (left) during the execution of the algorithm. The average of many samples is also shown. Presence of grey dots indicate that the sampling procedure is not deterministic.

Carlo methods for graphs. *arXiv preprint arXiv:1612.06404*, 2016.

P. Diaconis. Group representations in probability and statistics. In S. S. Gupta, editor, *Institute of Mathematical Statistics Lecture Notes—Monograph Series*, volume 11. 1988.

P. Diaconis. The markov chain monte carlo revolution. *Bulletin of the American Mathematical Society*, 46 (2):179–205, 2009.

P. W. Glynn. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33 (10):75–84, oct 1990.

L. J. Guibas. The identity management problema short survey. In *11th International Conference on Information Fusion*, pages 1–7. IEEE, 2008.

M. T. Harrison and J. W. Miller. Importance sampling for weighted binary random matrices with specified margins. *arXiv preprint arXiv:1301.3928*, 2013.

M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

J. Huang, C. Guestrin, and L. Guibas. Fourier theoretic probabilistic inference over permutations. *Journal of machine learning research*, 10(May):997–1070, 2009.

E. Jang, S. Gu, and B. Poole. Categorical reparameterization with Gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

S. Kato, H. Kaplan, T. Schrdel, S. Skora, T. Lindsay, E. Yemini, S. Lockery, and M. Zimmer. Global brain dynamics embed the motor command sequence of caenorhabditis elegans. *Cell*, 163(3):656 – 669, 2015. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2015.09.034.

D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

R. Kondor, A. Howard, and T. Jebara. Multi-object tracking with representations of the symmetric group. In *AISTATS*, volume 1, page 5, 2007.

H. W. Kuhn. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2): 83–97, 1955.

M. J. Kusner and J. M. Hernández-Lobato. GANs for sequences of discrete elements with the Gumbel-softmax distribution. *arXiv preprint arXiv:1611.04051*, 2016.

E. L. Lawler. The quadratic assignment problem. *Management science*, 9(4):586–599, 1963.

G. Lebanon and Y. Mao. Non-parametric modeling of partially ranked data. *Journal of Machine Learning Research*, 9(Oct):2401–2429, 2008.

K. Li, K. Swersky, and R. Zemel. Efficient feature learning using Perturb-and-MAP. *Neural Information Processing Systems Workshop on Perturbations, Optimization, and Statistics*, 2013.

C. H. Lim and S. Wright. Beyond the Birkhoff polytope: Convex relaxations for vector permutation problems. In *Advances in Neural Information Processing Systems*, pages 2168–2176, 2014.

S. W. Linderman, A. C. Miller, R. P. Adams, D. M. Blei, L. Paninski, and M. J. Johnson. Recurrent switching linear dynamical systems. *arXiv preprint arXiv:1610.08466*, 2016.

R. Lints, Z. F. Altun, H. Weng, T. Stephney, G. Stephney, M. Volaski, and D. H. Hall. WormAtlas Update. 2005.

J. Livet, T. A. Weissman, H. Kang, R. W. Draft, J. Lu, R. A. Bennis, J. R. Sanes, and J. W. Lichtman. Transgenic strategies for combinatorial expression of fluorescent proteins in the nervous system. *Nature*, 450(7166):56, 2007.

C. J. Maddison, A. Mnih, and Y. W. Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

C. L. Mallows. Non-null ranking models. i. *Biometrika*, 44(1/2):114–130, 1957.

M. Meilă, K. Phadnis, A. Patterson, and J. Bilmes. Consensus ranking under the exponential model. In *In Proceedings of the 23rd Annual Conference on Uncertainty in Artificial Intelligence (UAI*, 2007.

J. W. Miller, M. T. Harrison, et al. Exact sampling and counting for fixed-margin matrices. *The Annals of Statistics*, 41(3):1569–1592, 2013.

J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957.

S. M. Plis, S. McCracken, T. Lane, and V. D. Calhoun. Directional statistics on permutations. In *AISTATS*, pages 600–608, 2011.

V. Rao, R. P. Adams, and D. D. Dunson. Bayesian inference for Matérn repulsive processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2016.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of The 31st International Conference on Machine Learning*, pages 1278–1286, 2014.

T. Salimans and D. A. Knowles. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

J. Shin, N. Lee, S. Thrun, and L. Guibas. Lazy inference on object identities in wireless sensor networks. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, page 23. IEEE Press, 2005.

R. Sinkhorn and P. Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

L. R. Varshney, B. L. Chen, E. Paniagua, D. H. Hall, and D. B. Chklovskii. Structural properties of the caenorhabditis elegans neuronal network. *PLoS computational biology*, 7(2):e1001066, 2011.

J. T. Vogelstein, J. M. Conroy, V. Lyzinski, L. J. Podrazik, S. G. Kratzer, E. T. Harley, D. E. Fishkind, R. J. Vogelstein, and C. E. Priebe. Fast approximate quadratic programming for graph matching. *PLOS one*, 10(4):e0121002, 2015.

J. G. White, E. Southgate, J. N. Thomson, and S. Brenner. The structure of the nervous system of the nematode caenorhabditis elegans: the mind of a worm. *Phil. Trans. R. Soc. Lond*, 314:1–340, 1986.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

Supplement

MNIST reconstructions

In figure 5 we show some MNIST reconstructions using Gumbel-Softmax, stick-breaking and rounding reparameterizations. In all the three cases reconstructions are reasonably accurate, and there is diversity in reconstructions.

Limit analysis for Stick-breaking

Here we state and prove that for the stick-breaking we consider here, we can arrive to either arbitrary points in the i) simplex or ii) to any categorical distribution as limiting cases (in the temperature). First, we need some lemmas.

Lemma 1. The following statements are true:

1. the degenerate case where z_k is deterministic leads to $\pi \sim \delta(\tilde{\pi})$ (i.e, single atom in the point $\tilde{\pi}$). Also, if z_k can be any in $(0, 1)$ then any deterministic π in the interior of the simplex can be realized.
2. the degenerate case where z_k are Bernoulli with parameter $p_k(\theta) \in (0, 1)$ leads to π having an atomic distribution with atoms in the vertices of Δ^{k-1}; i.e, π is categorical. We have the following expression for the probabilities of the atoms $\pi_k = 1$ (one hot vectors):

$$P(\pi_k = 1) = \prod_{i=1}^{k-1}(1 - p_i(\theta))p_k(\theta) \quad k = 2, \dots, K-1,$$

$$P(\pi_K = 1) = \prod_{i=1}^{K-1}(1 - p_i(\theta)).$$

   Moreover, if for each index k any parameter of the Bernoulli variable z_k can be realized through appropriate choice of θ, then any categorical distribution can be realized.

Proof: (a) both claims are obvious and come from the invertibility of the function $\mathcal{SB} \circ h(\cdot)$. (b) the formulae for $P(\pi_k = 1)$ comes from expressing the event $\pi_k = 1$ equivalently as $\pi_k = 1, \pi_i = 0, i < k$ and then, conditioning backwards successively. The second statement comes from the following expression, which easily follows from (2):

$$p_k(\theta) = \frac{P(\pi_k = 1)}{P(\pi_{k-1} = 1)} \frac{p_{k-1}(\theta)}{1 - p_{k-1}(\theta)}, \quad k = 1, \dots, K-1.$$

The recursive nature of the above equation gives a recipe to iteratively determine the required $p_k(\theta)$, given $P(\pi_k = 1), P(\pi_{k-1} = 1)$ and the already computed $p_{k-1}(\theta)$.

Now we can state our results:

Lemma 2. If $z = \sigma(\psi), \psi \sim \mathcal{N}(\mu, \eta^2)$, then

1. the limit $\eta \to 0$ and μ fixed leads to the deterministic $z = \sigma(\mu)$.
2. the limit $\mu \to \infty, \eta^2 = \mu/K$ with K constant leads to $z \sim$ Bernoulli$(\Phi(K))$, with $\Phi(\cdot)$ denoting the standard normal cdf.

In both cases the convergence is in distribution

Proof. The first convergence is obvious. To see the second, let's index μ_n and study the cdf F of z_n on the interval (0,1) (it evaluates zero below zero and one above one).

$$\begin{aligned} F_{z_n}(x) &= P(\sigma(\psi_n) < x) && (2)\\ &= P(\psi_n < \sigma^{-1}(x)) && (3)\\ &= P(\mu_n + \mu_n/K\xi < \sigma^{-1}(x)), && (4)\\ &= P(\xi < \sigma^{-1}(x)K/\mu_n - K) && (5)\\ &= \Phi(\sigma^{-1}(x)K/\mu_n - K) && (6) \end{aligned}$$

Therefore, by continuity of Φ we obtain $F_{\Psi_n}(x) \to \Phi(-K)$ for all points $x \in (0, 1)$. On the other hand, the cdf of a bernoulli random F variable is given by a step function that abruptly changes at zero, from zero to $1 - p$, and at one, from $1 - p$ to 1. As convergence occurs at all continuity points (the interval $(0, 1)$), we conclude (recall, $1 - p = \Phi(-K) \to \Phi(K) = p$). Notice that the above representation only allows to converge to $p > 0.5$, as K has to be positive. This can be fixed by choosing sequence with negative μ instead.

Proposition. For the stick-breaking construction, any arbitrary distribution can be realized in the low-temperature limit. Also, in the high-temperature limit convergence is to certain point(s) in the interior of the simplex.

Proof: Consider each distribution separately

We have $z_k = \sigma\left(\frac{\mu_k + \eta_k \xi}{\tau}\right)$, in the low temperature case use Lemma 2 (b) by the always available representation $K = \frac{\mu}{\eta^2}$and conclude by Lemma 1(b). In the high temperature case convergence is to the point $\pi = \mathcal{SB}(0.5, 0.5, \dots, 0.5)$.

Variational inference for Permutation details

Continuous prior distributions.

Continuous relaxations requires re-thinking of the objective. As in Maddison et al. [2016], we maximize a relaxed ELBO, for which we need to specify a new continuous prior $p(x)$ over the latent variables. Moreover,

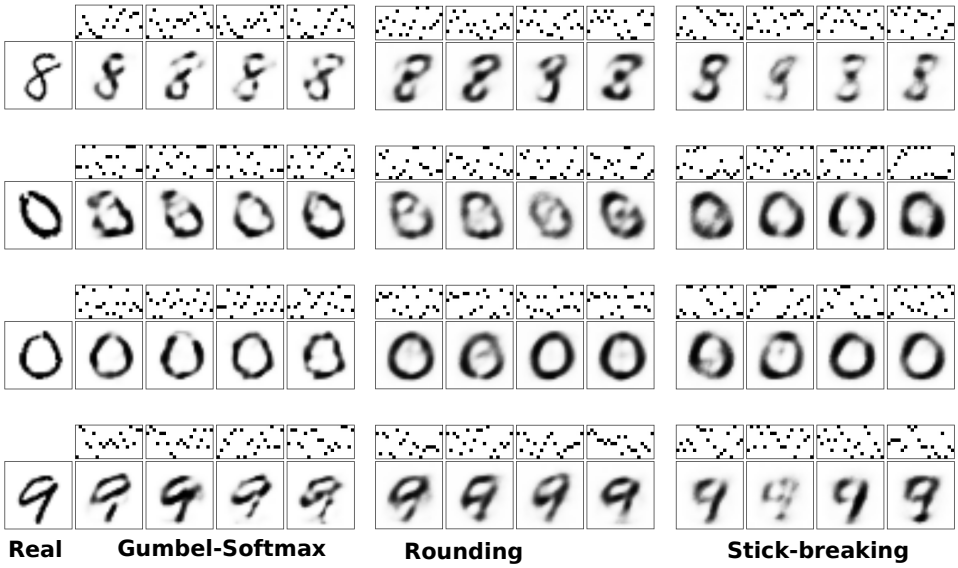

Figure 5: Examples of true and reconstructed digits from their corresponding random codes using with $N = 20$ categorical variables with $K = 10$ possible values.

it is critical to conceive sensible priors for permutations, that could serve in a variational inference routine to penalize configurations that are away from permutation matrices (i.e. close to the barycenter of the Brikhoff polytope).

For our categorical experiment on MNIST we use a mixture of Gaussians around each vertex, $p(x) = \frac{1}{N}\sum_{n=1}^{N}\mathcal{N}(x \,|\, e_k, \eta^2)$. This can be extended to permutations, where use a mixture of Gaussians for each dimension,

$$p(X) = \prod_{m=1}^{N}\prod_{n=1}^{N}\frac{1}{2}\left(\mathcal{N}(x_{mn} \,|\, 0, \eta^2) + \mathcal{N}(x_{mn} \,|\, 1, \eta^2\right).$$

Although this prior puts significant mass invalid points (e.g. $\mathbf{1}$), it penalizes X that far from $\mathcal{B}_N$.

Deriving an expression for the ELBO

Here we show that if $X = G(\Xi;\theta)$ with G differentiable one can evaluate the second term in equation (1) [6]. Moreover, here, to exploit the similarities between both methods (stick-breaking and rounding), we further factor G into two functions: $G = H \circ F$, $X = H(\Psi)$ and $\Psi = F(\Xi;\theta)$ (both H, F invertibles). This means all dependency of X in the parameters is through Ψ. Under this assumption, and denoting $\Psi \sim p(\Psi, \theta)$, the second term in equation (1) (without gradient) can be computed as: then

$$\begin{aligned}\mathbb{E}_{r(\Xi)}\left[-\log q(G(\Xi;\theta));\theta)\right] =& \mathbb{H}(\Psi;\theta) + \\ & E_{r(\Xi)}\left[\log|DH(F(\Xi,\theta))|\right].\end{aligned}$$

Proof: Indeed, first, it is obvious that

$$\mathbb{E}_{r(\Xi)}\left[-\log q(G(\Xi;\theta));\theta)\right] = \mathbb{E}_{r(\Xi)}\left[-\log q(H(F(\Xi,\theta));\theta)\right]$$

Then, by the 'Law of the Unconscious Statistician' we have:

$$\mathbb{E}_{r(\Xi)}\left[-\log q(H(F(\Xi,\theta));\theta)\right] = \mathbb{E}_{p(\Psi;\theta)}\left[-\log q(H(\psi);\theta)\right].$$

Now, by the change of variable theorem and derivative and determinant inversion rules, we obtain (D means the Jacobian, the matrix of derivatives) :

$$\begin{aligned}q(H(\Psi);\theta) &= p(H^{-1}(X);\theta)|DH^{-1}(X)| \\ &= p(\Psi;\theta)|DH(\Psi)|^{-1}.\end{aligned}$$

To conclude we use once more the Law of the Unconscious Statistician:

$$\begin{aligned}\mathbb{E}_{r(\Xi)}\left[-\log q(G(\Xi;\theta));\theta)\right] \quad &= \mathbb{E}_{p(\Psi;\theta)}\left[-\log p(\Psi;\theta)\right] + \\ &\quad \mathbb{E}_{p(\psi;\theta)}\left[\log|DH(\psi)|\right] \\ &= \mathbb{H}(\Psi;\theta) + \\ &\quad E_{r(\xi)}\left[\log|DH(F(\Xi;\theta))|\right].\end{aligned} \tag{7}$$

Estimating the ELBO

Here we describe how to compute each of terms of equation (7), needed for ELBO computations. First, as

[6] Notice that we uppercase the variables in (1) this is in consistency to our notation in section 3

Ψ is gaussian for both rounding and stick-breaking, the entropy term is straightforward and equal to $N \log(\eta^2 2\pi e)/2$ (η may depend on the temperature and depends on the method).

Notice that to state Ψ is gaussian in the stick-breaking case we slightly deviate from 3. Specifically, here we call $\Psi = \frac{\mu_{mn}+\eta_{mn}\Xi_{mn}}{\tau}$ and define $\Psi' = \sigma(\Psi)$.

The second term of equation (7) is estimated using Monte-Carlo samples, and its derivation depends on the method.

Rounding

Here H is piecewise linear: the set of discontinuities (border of the 'Voronoi cells' associated to each permutation) has Lebesgue measure zero. So we can still apply the change of variables theorem. Therefore, $\log |DH(F(\Xi;\theta))| = N \log \tau$. This means we don't even need to take samples to compute this term.

Stick-breaking

It is important to note that the transformation H that maps $\Psi' \to X$ is only piecewise continuous: the function is not differentiable at the points where the bounds change; for example, when changing Ψ' causes the active upper bound to switch from the row to the column constraint or vice versa.

Notice that these bounds only depend on values of X that have already been computed; i.e., those that are above or to the left of the (i,j)-th entry. Thus, the transformation from Ψ' to X is feed-forward according to this ordering. Consequently, the Jacobian of the inverse transformation H^{-1} , $\mathrm{d}\Psi'/\mathrm{d}X$, is lower triangular, and its determinant is the product of its diagonal,

$$\begin{aligned}
\left|\frac{\mathrm{d}\Psi'}{\mathrm{d}X}\right| &= \prod_{i=1}^{n-1}\prod_{j=1}^{n-1} \frac{\partial \psi_{ij}}{\partial \pi_{ij}} \\
&= \prod_{i=1}^{n-1}\prod_{j=1}^{n-1} \frac{\partial}{\partial \pi_{ij}} \sigma^{-1}\left(\frac{\pi_{ij}-\ell_{ij}}{u_{ij}-\ell_{ij}}\right) \\
&= \prod_{i=1}^{n-1}\prod_{j=1}^{n-1} \left(\frac{1}{u_{ij}-\ell_{ij}}\right)\left(\frac{u_{ij}-\ell_{ij}}{\pi_{ij}-\ell_{ij}}\right)\left(\frac{u_{ij}-\ell_{ij}}{u_{ij}-\pi_{ij}}\right) \\
&= \prod_{i=1}^{n-1}\prod_{j=1}^{n-1} \frac{u_{ij}-\ell_{ij}}{(\pi_{ij}-\ell_{ij})(u_{ij}-\pi_{ij})}
\end{aligned}$$

To compute the gradient of the forward transformation H one simply needs to invert the above (or put a negative sign, in the logarithm scale). Finally, to incorporate the effect of σ ($\Psi' = \sigma(\Psi)$, by the chain rule, one only needs to add a term corresponding to this derivative, $d\sigma(x)/dx = \sigma(x)\sigma(-x)$.