Reparameterizing the Birkhoff Polytope for Variational Permutation Inference

Anonymous Authors
Anonymous Institutions

Abstract

Many matching, tracking, sorting, and ranking problems require probabilistic reasoning about possible permutations, a set that grows factorially with dimension. 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 the usual step of relaxing a discrete set (here, of permutation matrices) to its convex hull, which here is the Birkhoff polytope: the set of all double-stochastic matrices. We then introduce two novel transformations: first, an invertible and differentiable map from unconstrained space to the Birkhoff polytope, and second, a similar map to a ball around the polytope. Both transformations include a temperature parameter that, in the limit, concentrates the densities on permutation matrices. We then exploit these transformations and reparameterization gradients to introduce variational inference over permutation matrices, and we show via a series of simulated and real experiments the value of this approach.

1 Introduction

Permutation inference is central to many modern machine learning problems. Identity management [**?**] and multiple-object tracking [**??**] 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 [**???**]. Furthermore, many probabilistic models, like preferential attachment network models [**?**] and repulsive point process models [**?**], 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 neuroscience, experimentalists now measure whole-brain recordings in *C. Elegans* [**??**], a model organism with a known synaptic network [**?**]; a current challenge is matching the observed neurons to corresponding nodes in the reference network. In Section 5, we address this problem from a Bayesian perspective in which permutation inference is a central component of a larger inference 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 [**???**]. However, many probabilistic tasks require reasoning about the posterior distribution over permutation matrices. A variety of Bayesian permutation inference algorithms have been proposed, leveraging sampling methods [**???**], Fourier representations [**??**], as well as convex [**?**] and continuous [**?**] relaxations for approximating the posterior distribution. Here, we address this problem from an alternative direction, leveraging stochastic variational inference [**?**] and reparameterization gradients [**??**] 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 [**???**]. 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 onto itself. When this set is finite, the mapping is conveniently represented as a binary matrix $X \in \{0,1\}^{N\times N}$ where $X_{m,n} = 1$ implies that element m is mapped to element 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 the convex hull of the set of permutation matrices are the set 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* [**?**] maps the positive orthant onto $\mathcal{B}_N$ by iteratively normalizing the rows and columns, and the *Hungarian algorithm* [**??**] solves the minimum weight bipartite matching problem—optimizing a linear objective over the set of permutation matrices—in cubic time.

2.2 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 [**?**]. **?** 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 [**?**]. Another line of work considers inference in the spectral domain, approximating distributions over permutations with the low frequency Fourier components [**??**]. Perhaps most relevant to this work, **?** 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. We will relax permutations to points in the Birkhoff polytope and derive temperature-controlled densities such that as the temperature goes to zero, the distribution converges to an atomic density on permutation matrices. This will enable efficient variational inference with the reparameterization trick, which we describe next.

2.3 Variational inference and the reparameterization trick

Given an intractable model with data y, likelihood $p(y \,|\, x)$, and prior $p(x)$, 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 sometimes leverage the *reparameterization trick* [**??**]. 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. 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 [**??**].

Critically, the gradients in (1) can only be computed if x is continuous. Recently, [**?**] and [**?**] proposed the "Gumbel-softmax" method for discrete variational

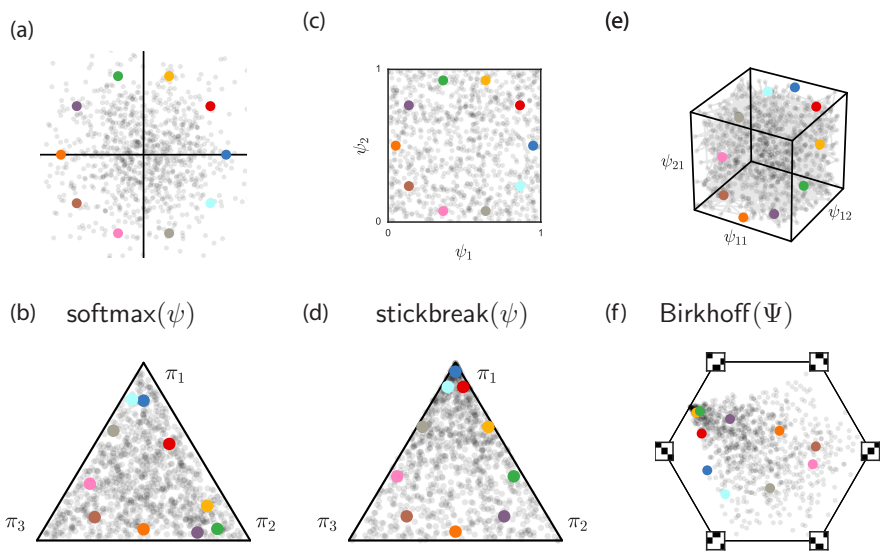

Figure 1: Reparameterizations of discrete polytopes. (a,b) The Gumbel-softmax, or "Concrete" transformation maps points $\psi \in \mathbb{R}^N$ to points $x \in \Delta_N$ by adding noise and applying the softmax. Here we show a slice for $N = 3$ with $\psi_3 = 0$. Colored points are aids to visualize 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).

inference. It is based on the following observation: one-hot vectors $x \in \{0,1\}^N$ can be viewed as vertices of the simplex Δ_N; likewise, discrete probability mass functions $q(x;\theta)$ can be seen as atomic densities on the vertices of the simplex. This motivates a natural relaxation: let $q(x;\theta)$ be a density on the interior of the simplex instead and anneal this density such that it converges to an atomic density on the vertices. Figure 1a and b illustrate this idea. Real-valued random variables, e.g. Gumbel random variates, are mapped through a temperature-controlled softmax function, $g(\psi;\tau) = \left[e^{\psi_1/\tau}/Z, \ldots, e^{\psi_K/\tau}/Z\right]$, where $Z = \sum_{k=1}^{K} e^{\psi_k/\tau}$, to obtain points in the simplex. As τ goes to zero, the density concentrates on one-hot vectors. We extend this idea to the Birkhoff polytope for variational permutation inference.

3 Variational permutation inference via reparameterization

The Gumbel-softmax method scales linearly with the support of the discrete distribution, rendering it prohibitively expensive for direct use on the set of $N!$ permutations. Instead, we develop two transformations to map $O(N^2)$-dimensional random variates to points in or near the Birkhoff polytope. Like the Gumbel-softmax method, these transformations will be controlled by a temperature that concentrates the resulting density near permutation matrices. The first method is a novel "stick-breaking" construction; the second rounds points toward permutations with the Hungarian algorithm. We present these in turn and then discuss their relative merits.

3.1 Stick-breaking transformations of the Birkhoff polytope

Let Ψ be a 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, analogously to the stick-breaking construction of the Dirichlet process [**?**]. 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. The first row follows the standard stick-breaking construction,

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

$$x_{1N} = 1 - \sum_{n=1}^{N-1} x_{1n}.$$

This is illustrated in Figure 1c and d, where points in

the unit square map to points in the simplex.

Subsequent rows are more interesting, requiring a novel advance on the typical uses of stick breaking. Here we the need to conform to row and column sums (which introduces an upper bound), and a lower bound induced by stick remainders that must allow completion of subsequent sum constraints. Specifically, 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 $\theta = \{\mu_{mn}, \eta^2_{mn}\}_{m,n=1}^{N}$ 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. This map is illustrated in Figure 1e and f for permutations of $N = 3$ elements.

To our knowledge, this map is a novel transformation to the Birkhoff polytope with the essential properties for gradient-based variational permutation inference.

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 $\Xi \in \mathbb{R}^{N\times N}$, $M \in \mathbb{R}_+^{N\times N}$, and $H \in \mathbb{R}_+^{N\times N}$;

(ii) Map $M \to \mathsf{sink}(M)$, a point near the Birkhoff polytope, using the Sinkhorn-Knopp algorithm;

(iii) Set $\Psi = \mathsf{sink}(M) + H \odot \Xi$ where $\odot$ denotes elementwise multiplication;

(iv) Find $\mathsf{round}(\Psi)$, the nearest permutation matrix to Ψ, using the Hungarian algorithm;

(v) Output $X = \tau\Psi + (1 - \tau)\mathsf{round}(\Psi)$.

This procedure is a mapping $X = g(\Xi; M, H)$, and when the elements of Ξ are independent sampled from a standard normal distribution, it implicitly defines a distribution over matrices X parameterized by M and H. Furthermore, as τ goes to zero, the density concentrates on permutation matrices. We use this procedure to define a variational distribution with density $q(X; M, H)$.

To compute the ELBO and its gradient (1), we need to evaluate $q(X; M, H)$. By construction, steps (i) and (ii) involve differentiable transformations of parameter M to set the mean close to the Birkhoff polytope, but since these do not influence the distribution of Ξ, the non-invertibility of the sink function poses no problems. Had we applied sink directly to Ξ, this would not be true. The challenge in computing the density stems from the rounding in steps (iv) and (v).

To compute $q(X; M, H)$, we need the inverse $g^{-1}(X; M, H)$ and its Jacobian. The inverse is straightforward: when $\tau \in [0, 1)$, $\mathsf{round}(\Psi)$

outputs a point strictly closer to the nearest permutation, implying $\mathsf{round}(\Psi) \equiv \mathsf{round}(X)$. Thus, the inverse is $\Psi = \frac{1}{\tau}X - \frac{1-\tau}{\tau}\mathsf{round}(X)$. A slight wrinkle arises from the fact that step (v) maps to a subset $\mathcal{X}_\tau \subset \mathbb{R}^{N\times N}$, but this inverse is valid for all $X \in \mathcal{X}_\tau$.[1]

The Jacobian is more challenging due to the non-differentiability of round. However, since the nearest permutation output only changes at points that are equidistant from two or more permutation matrices, round is a piecewise constant function with discontinuities only at a set of points with zero measure. In practice, we find that we can safely ignore these discontinuities.

With the inverse and its Jacobian, we have

$$q(X; M, \Sigma) = \frac{1}{\tau}\mathcal{N}\left(\frac{1}{\tau}X - \frac{1-\tau}{\tau}\mathsf{round}(X); \mathsf{sink}(M), \Sigma\right),$$

for $X \in \mathcal{X}_\tau$. In the zero-temperature limit we recover a discrete distribution on permutation matrices; otherwise the density concentrates near the vertices as $\tau \to 0$. This transformation leverages computationally efficient algorithms like Sinkhorn-Knopp and the Hungarian algorithm to define a temperature-controlled variational distribution near the Birkhoff polytope, and it enjoys many theoretical and practical benefits.

3.3 Theoretical considerations

Stick-breaking and rounding each have their strengths and weaknesses. Here we list some of their conceptual differences. While these considerations aid in understanding the differences between the two transformations, the ultimate test is in their empirical performance, which we study in Section 4.

- Stick-breaking relaxes to $\mathcal{B}_N$ whereas rounding relaxes to $\mathbb{R}^{N\times N}$. The Birkhoff polytope is intuitively appealing, but as long as the likelihood, $p(y \mid X)$, accepts real-valued matrices, either may suffice.
- Rounding uses the $O(N^3)$ Hungarian algorithm in its sampling process, whereas stick-breaking has $O(N^2)$ complexity. In practice, the stick-breaking computations are slightly more efficient.
- Rounding can easily incorporate constraints. If certain mappings are invalid, i.e. $X_{mn} \equiv 0$, they are given an infinite cost in the Hungarian algorithm.[2] 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.

In summary, stick-breaking offers an intuitive advantage—an exact relaxation to the Birkhoff polytope—but it suffers from its sensitivity to ordering and its inability to easily incorporate constraints. As we show next, these concerns ultimately lead us to favor the rounding based methods in practice.

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 do our proposed continuous relaxations offer advantages over alternative Bayesian permutation inference algorithms?

We first studied how stick-breaking and rounding perform in simple categorical inference tasks, where they offer an alternative to the Gumbel-softmax method. We found that our methods were comparable, though slightly inferior; this is the price paid for techniques that extend to more complicated discrete inference problems. Since our main interest lies in permutation inference, we defer these results to the supplement.

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 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 **?** 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, $d(\phi, \phi_0) = \sum_{i=1}^{N} |\phi(i) - \phi_0(i)|$ is a distance between permutations, and θ controls the spread around ϕ_0.

[1] 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 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.

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

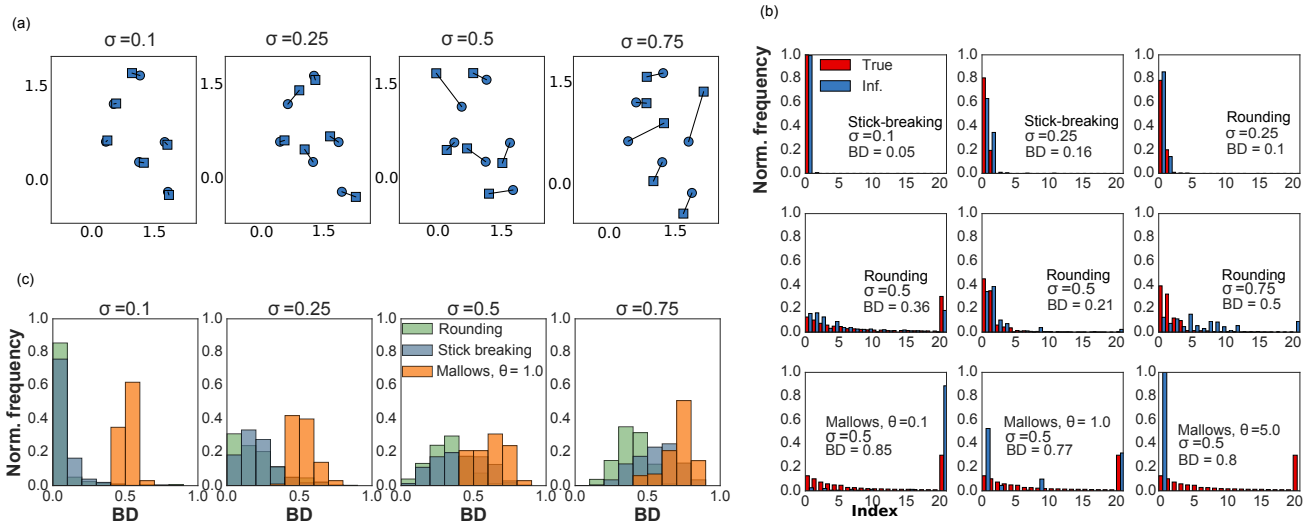

Figure 2: Synthetic matching experiment results. The goal is to infer the lines that match squares to circles. (a) Examples of center locations (circles) and noisy samples (squares), at different noise variances. (b) For illustration, histograms of the true and inferred posterior distribution of identities 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.

This is the most popular exponential family model for permutations, but since it is necessarily unimodal, it can fail to capture complex permutation distributions.

Table 1: 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

We measured the discrepancy between true posterior and an empirical estimate of the inferred posteriors using using the Battacharya distance (BD). We fit $q(X;\theta)$ for both stick-breaking and rounding transformations, sampled the variational posterior, and rounded the samples to the nearest permutation matrix with the Hungarian algorithm. For the Mallows distribution, we set ϕ_0 to the MAP estimate, also found with the Hungarian algorithm, and sampled using MCMC.

We found our method outperforms the simple Mallows distribution and reasonably approximates non-trivial distributions over permutations. Fig 2 illustrates our findings, showing (a) sample experiment configurations; (b) examples of inferred, discrete, posteriors for stick breaking (top), rounding (middle), and Mallows (bottom); and (c) histogram of Battacharya distance. These latter are summarized in Table 1.

5 Inferring neuron identities in *C. elegans*

Finally, we consider an application motivated by the study of the neural dynamics in *C. elegans*. This worm is a model organism in neuroscience as its neural network is stereotyped from animal to animal and its complete neural wiring diagram is known [**?**]. We represent this network, or connectome, as a binary adjacency matrix $A \in \{0,1\}^{N\times N}$, shown in Fig. 3a. The hermaphrodyte has $N = 278$ somatic neurons, and (undirected) synaptic connections between neurons m and n are denoted by $A_{mn} = 1$.

Modern recording technology enables simultaneous measurements of hundreds of these neurons simultaneously [**??**]. However, matching the observed neurons to nodes in the reference connectome is still a manual task. Experimenters consider the location of the neuron along with its pattern of activity to perform this matching, but the process is laborious and the results prone to error. We prototype an alternative solution, leveraging the location of neurons and their activity in a probabilistic model. We resolve neural identity by integrating different sources of information from the connectome, some covariates (e.g. position) and neural dynamics. Moreover, we combine infor-

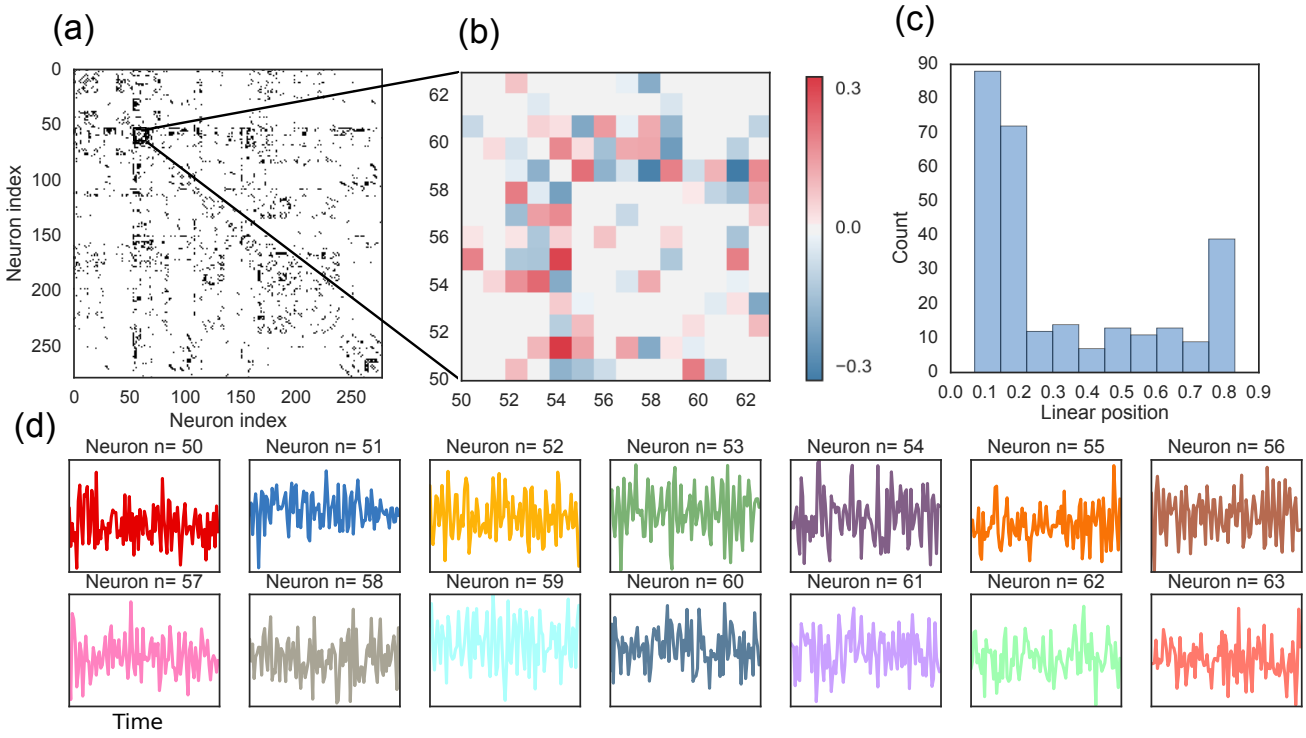

Figure 3: Problem setup. (a) Hermaphrodite C.elegans reference connectome (from **??**) consisting of 278 somatic neurons, merging two distinct types of synapses: chemical and electrical (gap junctions). (b) Example of matrix W consistent with the connectome information (only 14 neurons for visibility),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

mation from many individuals to facilitate identity resolution.

Probabilistic Model. Let J denote the number of worms and $Y^{(j)} \in \mathbb{R}^{T_j \times N}$ denote a recording of worm j with T_j time steps and N neurons. We model the neural activity with a linear dynamical system, $Y_t^{(j)} = X^{(j)} W X^{(j)\mathsf{T}} Y_{t-1}^{(j)} + \varepsilon_t^{(j)}$ where $X^{(j)}$ is a latent permutation of neurons that must be inferred, and W is a weight matrix reflecting the influence of one neuron on another (with neurons ordered as in the reference connectome A), and $\varepsilon_t^{(j)} \sim \mathcal{N}(0, I)$ is noise. The reference connectome tells us which entries of W may be non-zero; specifically, $A_{mn} = 0$ implies $W_{mn} = 0$. However, the synaptic weights must be inferred for entries where $A_{mn} = 1$. Fig. 3d shows simulated traces from a network that respects the connectivity of A and has random Gaussian weights.

Our goal is to infer W and $\{X^{(j)}\}$ given $\{Y^{(j)}\}$ using variational permutation inference. We place a standard Gaussian prior on W and a uniform prior on $X^{(j)}$, and we use the rounding transformation to approximate the posterior, $p(W, \{X^{(j)}\} \mid \{Y^{(j)}\}) \propto p(W) \prod_m p(Y^{(j)} \mid W, X^{(j)})\, p(X^{(j)})$.

Finally, we use neural position along the worm's body to constrain the possible neural identities for a given neuron. We use the known positions of each neuron [**?**], approximating the worm as a one-dimensional object with neurons locations distributed as in Fig. 3c. Then, given reported positions of the neurons, we can conceive a binary *constraint* matrix $C^{(j)}$ so that $C_{mn}^{(j)} = 1$ if (observed) neuron m is close enough to (canonical) neuron n; i.e., if their distance is smaller than a tolerance ν. We 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.

Results. We compared against three methods: (i) naive variational inference, where we do not enforce the constraint that $X^{(j)}$ be a permutation and instead treat each row of $X^{(j)}$ as a Dirichlet distributed vector; (ii) MCMC, where we alternate between sampling from the conditionals of W (gaussian) and $X^{(j)}$, from which one can sample by proposing local swaps, as described in **?**, and (iii) maximum a posteriori estimation (MAP). Our MAP algorithm alternates between the optimizing estimate of W given $\{X^{(m)}, Y^{(m)}\}$ using linear regression and finding the optimal $X^{(j)}$. The second step requires solving a quadratic assignment problem (QAP) in $X^{(j)}$; that is, it can be expressed as $\mathrm{Tr}(AXBX^{\mathsf{T}})$ for matrices A, B. We used the QAP solver proposed

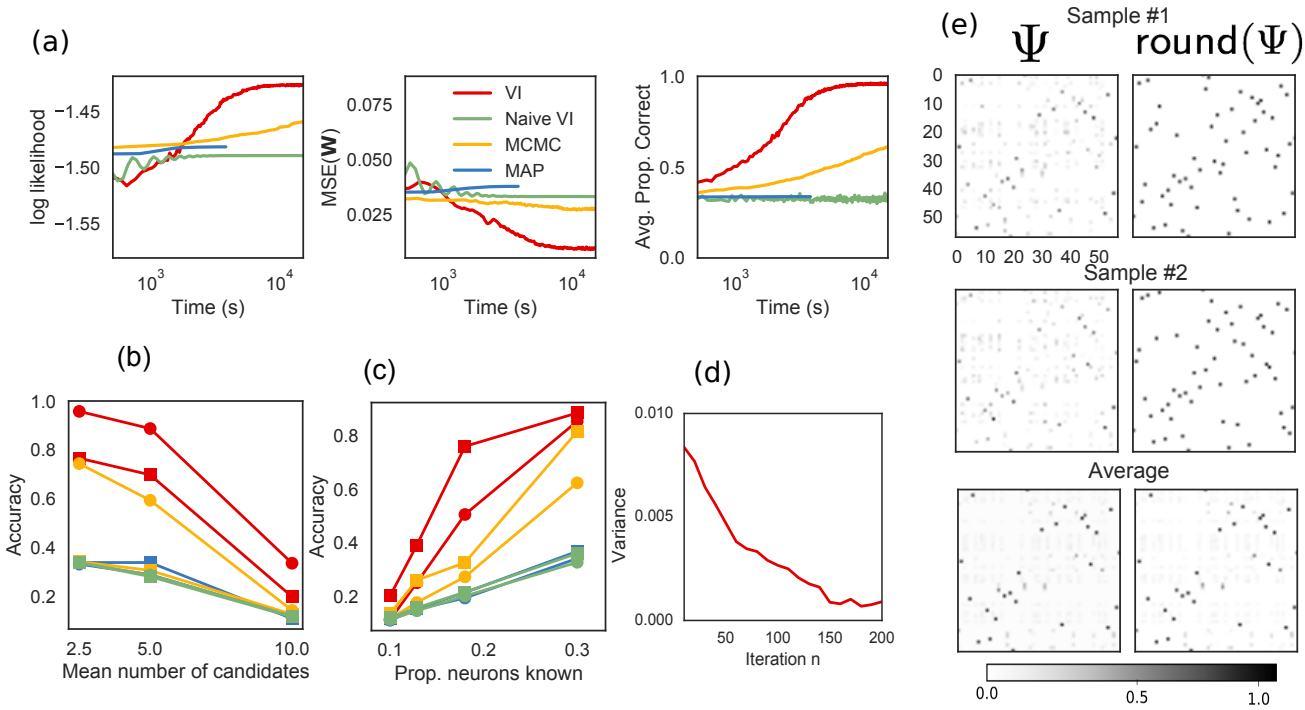

Figure 4: Results on the C.elegans inference example. (a) An example of convergence of the algorithm, and the baselines. (b) Accuracy of identity inference as a function of mean number of candidates (correlated with ν), for $M = 1$ worm (square) and combining information of $M = 5$ worms (circles). (c) 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 $\mathsf{round}(\Psi)$ (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.

by **?**.

We find that our method outperforms each baseline. Fig. 4a illustrates convergence to a better solution for a certain parameter configuration. Moreover, Fig. 4b and Fig. 4c show 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.

Altogether, these results indicate our method enables a more efficient use of information than its alternatives. This is consistent with other results showing faster convergence of variational inference over MCMC **?**, especially with simple Metropolis-Hastings proposals. In our experiments, MCMC would eventually obtain similar if not better results, but the local proposals—swapping pairs of labels—leads to slow convergence. On the other hand, our parameterization allows to more freely explore the parameter space. This is observed in Fig 4d: variability of permutation samples is high during iterations (see variability in Fig 4e), but eventually shrinks as it settles on a small number of likely permutations.

6 Discussion

Our results provide evidence that permutation variational inference provides a valuable tool, especially in complex problems like neural identity inference where information must be aggregated from disparate sources in a hierarchical model. As we apply this to real neural recordings, we must consider more realistic, nonlinear models of neural dynamics. Here, again, we expect variational methods to shine, leveraging automatic gradients of the relaxed ELBO to efficiently explore the space of variational posterior distributions.

Supplement

Variational Autoencoders (VAE) with categorical latent variables

We considered the density estimation task on MNIST digits, as in **??**, 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 **??**, finding similar (although slightly worse) results (Table 2). 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. Since this task is ancillary to our goal of permutation inference, we defer most details to the supplement.

Table 2: Summary of results in VAE

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

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

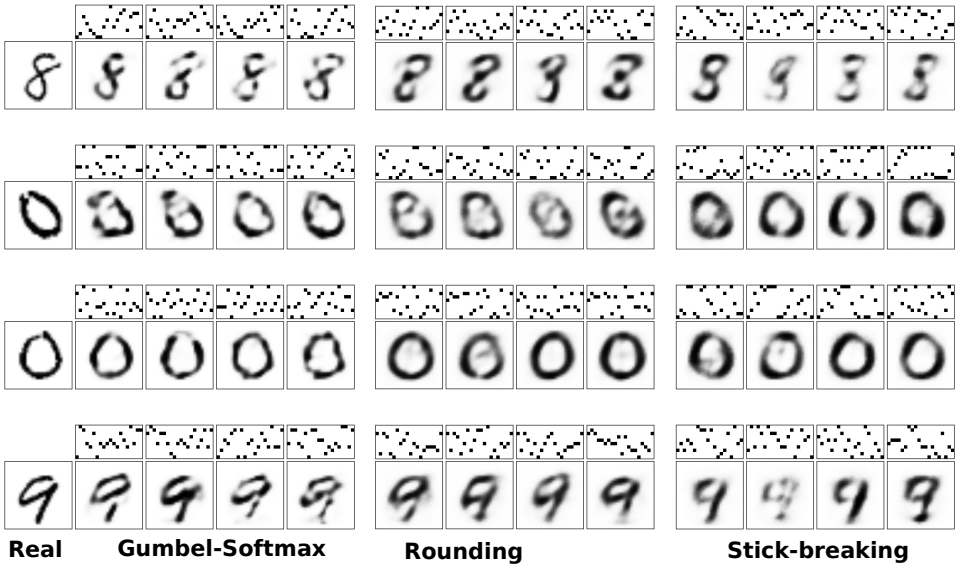

Figure 5: Examples of true and reconstructed digits from their corresponding random codes using with $N = 20$ categorical variables with $K = 10$ possible values.

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 $\pi = \mathcal{SB}(0.5, 0.5, \ldots, 0.5)$.

Variational inference for Permutation details

Continuous prior distributions.

Continuous relaxations requires re-thinking of the objective. As in **?**, we maximize a relaxed ELBO, for which we need to specify a new continuous prior $p(x)$ over the latent variables. Moreover, 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) [3]. 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

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

Proof: Indeed, first, it is obvious that

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

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

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:

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

Estimating the ELBO

Here we describe how to compute each of terms of equation (7), needed for ELBO computations. First, as 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, $d\sigma(x)/dx = \sigma(x)\sigma(-x)$.

Experiment details

In all experiment we used the ADAM as optimizer, with learning rate 0.1. For rounding, the parameter vector H defined in 3.2 (iii) was constrained to lie in the interval $[0.1, 0.5]$. The temperature was calibrated using a grid search on the interval $[0, 1]$. For stick-breaking. Experiments were run on a High Performance Computer (HPC), allowing the execution of hundreds of processes in parallel to efficiently determine best hyperparameter configurations.

Variational auto-encoder experiment was done in Tensorflow [**?**]. For synthetic matching experiments and the C. elegans example we used Autograd [**?**], explicitly avoiding propagating gradients through the non-differentiable operation of solving a matching problem (the round in 3.2) Synthetic